Proceedings of the 17th Annual Conference of the European Association for Machine Translation

EAMT2014

Dubrovnik, Croatia, 16th-18th June 2014

EUROPEAN ASSOCIATION FOR MACHINE TRANSLATION

Edited by Marko Tadić, Philipp Koehn, Johann Roturier, Andy Way

EAMT 2014

Proceedings of the 17th Annual Conference of the European Association for Machine Translation

Dubrovnik, Croatia, June 15th–18th 2014

Edited by
Marko Tadić, Philipp Koehn, Johann Roturier, Andy Way

EUROPEAN
ASSOCIATION
FOR MACHINE
TRANSLATION

Croatian Language Technologies Society

Table of Contents

Invited talk

Poster Session A

Research Papers

Product/Project Papers

Oral Session 1
Research Papers

Oral Session 2
User Papers

Oral Session 3
Research Papers

Poster Session B

Product/Project Papers

User Papers

Oral Session 4

Research Papers

Oral Session 5

Research Papers

Oral Session 6

User Papers

Foreword

I am very pleased that this year's European Association for Machine Translation (EAMT) annual conference is being held in the beautiful city of Dubrovnik, Croatia. This continues the policy I started in 2009 when I became President of bringing EAMT to new regions of Europe. This began with our first visit to the Iberian peninsula (Barcelona, 2009), our first conference in France in 2010 (St. Raphaël), followed by the Benelux region in 2011 (Leuven), and continuing in 2012 with our first conference hosted in Italy (Trento). As a teaser, I can assure you that next year's conference will continue this trend, although of course you won't find out the actual conference until the closing session of this year's conference!

The EAMT organised its first Workshop/Conference back in 1996, and now we come to our Seventeenth Annual Conference in 2014. Of course, this is our first meeting since 2012, as last year we returned to the South of France as the proud hosts of the very successful MT Summit XIV, held in Nice. To me, this demonstrates very clearly that the EAMT as an organisation is continuing to grow and thrive. As I've noted before, since its inception in 1997, the EAMT has not raised its membership rates, and we will continue to hold the cost of membership for 2014. Joining us is great value, especially in years like 2014 and 2015 where more than one IAMTaffiliated event takes place: in 2014, EAMT here, and AMTA later in the year in Vancouver: http://amtaweb.org/; and in 2015, EAMT and the MT Summit , the latter in Miami. The close cooperation – including conference discounts for all IAMT members – with the other regional associations continues, despite both AMTA (Mike Dillinger) and AAMT (Hiromi Nakaiwa) having elected new Presidents. As this is my penultimate conference as EAMT President, I can only hope that this partnership continues and thrives in the future.

As ever, I would like to thank my colleagues on the EAMT Committee, who continue to provide me with invaluable support. They work tirelessly on behalf of all of us, and we are all very fortunate to have such a strong body of colleagues representing our Association. Some of those members are moving on this year, to be replaced with new blood on the Committee. I urge all of you to consider contributing to this service to the community.

As in the recent past, the strength of the programme for this 17th Conference demonstrates clearly that for many, EAMT is a 'must attend' on the conference circuit. Accordingly, I would like to thank the Programme Co-Chairs Philipp Koehn (Research track) and Johann Roturier (User track), for helping me assemble a very attractive programme, comprising of Research and User tracks, poster sessions, and an excellent Invited Speaker in Jost Zetzsche. As in recent conferences, we continue to feature a special session featuring prominent FP7 projects, which has proven very attractive in the past.

Last but not least, I would especially like to thank our local organizer, Marko Tadić, who very generously volunteered to hold the meeting in Dubrovnik. We are very grateful to Marko and his team for their excellent organization of this event.

Finally, thanks to all of you for coming. I hope you all enjoy the conference, that you benefit from the excellent programme that has been assembled, and that you go away from here having made new friends.

Andy Way
Deputy Director of the CNGL,
School of Computing,
Dublin City University.

President of the EAMT

away@computing.dcu.ie

Message from the Conference Chair

It is my privilege and great pleasure to welcome you in the Centre for Advanced Academic Studies (CAAS) for the 17th Conference of the European Association for Machine Translation. I am very proud that the EAMT conference is organized this time in Croatia and Dubrovnik was the natural choice having in mind not just its splendour and beauty, but also having the opportunity to use this convenient building that belongs to the University of Zagreb. We are indeed here on the academic ground and I am confident that the quality of papers and our discussions will confirm this statement. I will have to express my gratitute to the EAMT Board for providing me with the opportunity to host the 17th EAMT conference and to be able to contribute to a series of successful EAMT conferences.

Following the increasing popularity of the EAMT conference over last few years, the EAMT Board applied the same conference format as two years ago: two and half days of intensive oral and poster sessions, accompanied by a rich social programme. I hope you will enjoy 44 papers in three different tracks – research, user and product/project – that will give an overview of current developments and trends in Machine Translation. On top of that we also have a half-day pre-conference workshop QTLaunchPad that preceeds the main conference.

The conference will be held in the CAAS building with coffee breaks in the same venue. The lunches are organized in a nearby restaurant Mimoza which served us on many occassions and is not more than three minutes of pleasant walk away. However, don't forget that the intellectural feast should never be left alone and this is why our social programme included the guided city tour, that should unveil some of the cultural, historical and artistic secrets that Dubrovnik is hiding below its glamurous surface. The welcome reception is organized in the very CAAS, while the conference dinner is combined with the excursion to a famous town of Ston where in middle ages Dubrovnik Republic was collecting one of its most important resources – sea salt. The salt is today being collected in the very same manner and it received the attribute “ecological” in its full meaning. I hope that the conference dinner will give you the taste of the most famous shell fish breeding area in Adriatic with its unique varieties of species that you can't find anywhere else in Europe.

This conference couldn't be so successful without the hard work of team involved in its organization. I would like to thank the Program Co-Chairs, Philipp Koehn and Johann Roturier for taking care about the large number of submissions while Andy Way and Mikel Forcada were also at alert whenever needed. I wish John Hutchins could be with us, but unfortunately, his health condition prevented his from coming. I am confident that his good spirit will be with us all the time. I will certainly have to thank to our executive organizers, Ulix d.o.o. and particularly Ana Skolan, Lucija Brala and Krešimir Korda, as well as my own Zagreb team composed of Daša Berović, Danijela Merkler and Matea Srebačić. I should not forget the staff in CAAS, led by Tanja Grzilo, that helped us a lot in organizing this event.

Our sponsors should not be ommitted and this time we got sponsoring from Bloomberg (Silver sponsor), ELRA (Bronze sponsor), Springer (Best paper award sponsor) and Ulix. In the same time we also received support from the Ministry of Science, Education and Sport of the Republic of Croatia.

I wish you a pleasant stay and very successful 17th EAMT conference in Dubrovnik!

Marko Tadić
Department of Linguistics
Faculty of Humanities and Social Sciences
University of Zagreb

marko.tadic@ffzg.hr

Message from Programme Chairs

It is a great pleasure for us to welcome you to the 17th Conference of the European Association for Machine Translation (EAMT) in Dubrovnik. We have been happy to serve as programme co-chairs of a conference that has become the yearly reference conference for European machine translation developers, researchers and users, and keeps growing year by year. A sign of this growth is that the format of the previous conference (2½ days instead of 2 days) was preserved this year to keep to the single track format – which makes EAMT events very homely for regulars and newcomers alike. As in previous years, the conference has three main tracks: (i) a research track, where researchers report about significant research results in any aspect of machine translation and related areas, with a substantial evaluation component, (ii) a user track, where users report their experiences with machine translation in business, government, or NGOs, and (iii) a projects track to publicize EU and international projects and initiatives. The project track was extended to accommodate product descriptions in order to further encourage participation from developers/industry. In order to encourage submissions for the user track, the required format of the submissions was a short paper with 2-4 pages. For projects/product demonstrations, both submissions only required a 1-page abstract. We received the following number of submissions - a total of 73 papers: 40 in the research track, 19 in the user track, and 14 project/product descriptions. Most of the latter were accepted, but were reformulated by the project participants to conform to the conference style-guide. As far as research and user papers are concerned, after double-blind review by at least three leading MT reviewers, 27 of them (45%) were accepted and found their way into the proceedings: 16 research papers (40%) – 10 for oral presentation and 6 for poster presentation – and 11 user papers (58%), 6 for oral presentation and 5 for poster presentation. Two submissions that were rejected from the user track were transferred to the Project/Product track based on recommendations from reviewers. Poster presenters will also have the opportunity to showcase their work in a two-minute poster boaster oral session. As expected, submissions come mainly from Europe. We also received papers with authors from the United States, Japan, Canada, Brazil, China, Kazakhstan, India, Hong Kong, Russia, Tunisia and the Republic of Korea. We are in debt to the members of the programme committee and to the secondary reviewers they appointed for some of their papers. We especially thank them for their invaluable help, which most of them completed on time, which made our lives easier! We hope that the reviewers' comments were useful and constructive and helped all authors: for those whose papers were not accepted, by increasing their chance in a later submission somewhere else; and for those whose papers got in, to improve their manuscripts. We know we did not give them a lot of time to do so, and we thank authors for sending their camera-ready versions on time. We hope that the resulting selection of papers, which you have in your conference pack, truly represents the best of machine translation research, development and real-world usage. As an opener, we will enjoy an invited talk by Jost Zetzsche, which we hope will appeal to both our research and our user audience. During the conference, we will also have a presentation by the winner of the EAMT Best Thesis Award. We thank you all: authors, presenters, members of the programme committee, reviewers and secondary reviewers, and attendees, for helping us to make EAMT 2014 a success: we hope you enjoy the programme that we have prepared for you. As these proceedings are being finalized, our job is almost finished, and the conference is now in good hands: those of the local organizers in Dubrovnik, headed by Marko Tadić. It has been great to work with them, and we send them a special thank you!

Johann Roturier
Symantec Research Labs

Philipp Koehn
Johns Hopkins University / University of Edinburgh

EAMT 2014 co-programme chairs

Committees

Conference Chair

Marko Tadić (University of Zagreb, Faculty of Humanities and Social Sciences)

Programme Chairs

Research Track

Philipp Koehn (University of Edinburgh, UK)

User Track

Johann Roturier (Symantec, Ireland)

Programme Committee

Research Track

Eleftherios Avramidis (DFKI, Germany)
Alexandra Birch (University of Edinburgh, Scotland)
Ondřej Bojar (Charles University, Czech Republic)
Christian Buck (University of Edinburgh, Scotland)
Bill Byrne (University of Cambridge, England)
Michael Carl (Copenhagen Business School, Denmark)
Francisco Casacuberta (Polytechnic University of Valencia, Spain)
Mauro Cettolo (FBK, Italy)
David Chiang (University of Southern California, USA)
Nadir Durrani (University of Edinburgh, Scotland)
Chris Dyer (Carnegie Mellon University, USA)
Christian Federmann (Microsoft, USA)
Marcello Federico (FBK, Italy)
Mark Fishel (University of Zurich, Switzerland)
Mikel Forcada (Universitat d'Alacant, Spain)
George Foster (NRC, Canada)
Josef van Genabith (CNGL, Dublin City University, Ireland)
Ulrich Germann (University of Edinburgh, Scotland)
Barry Haddow (University of Edinburgh, Scotland)
Rejwanul Haque (Lingo24, UK)
Christian Hardmeier (University of Uppsala, Sweden)
Teresa Herrmann (Karlsruhe Institute of Technology, Germany)
Matthias Huck (University of Edinburgh, Scotland)
Gonzalo Iglesias (University of Cambridge, England)
Jie Jiang (Applied Language Solutions, UK)
Marcin Junczys-Dowmunt (Adam Mickiewicz University, Poland)
Maxim Khalilov (TAUS)
Roland Kuhn (NRC, Canada)
Qun Liu (CNGL, Dublin City University, Ireland)
Adam Lopez (Johns Hopkins University, USA)
José B. Mariño (Polytechnic University of Catalonia, Spain)
Christof Monz (University of Amsterdam, Netherlands)

Jan Niehues (Karlsruhe Institute of Technology, Germany)
Sergio Penkale (Lingo24, UK)
Maja Popović (DFKI, Germany)
Stefan Riezler (University of Heidelberg, Germany)
Felipe Sánchez Martínez (Universitat d'Alacant, Spain)
Khalil Simaan (University of Amsterdam, Netherlands)
Ankit Srivastava (CNGL, Dublin City University, Ireland)
Sara Stymne (University of Uppsala, Sweden)
Jörg Tiedemann (University of Uppsala, Sweden)
Dan Tufiş (Romanian Academy, Romania)
Francis M. Tyers (Universitat d'Alacant, Spain)
David Vilar (Pixformance)
Jörn Wübker (RWTH Aachen, Germany)
François Yvon (LIMSI, France)

User Track

Jeff Allen (SAP, France)
Nora Aranberri (University of the Basque Country, Spain)
Pratyush Banerjee (Symantec, Ireland)
Nuria Bel (UPF, Spain)
Ergun Bicici (CNGL, Dublin City University, Ireland)
Frédéric Blain (Université du Maine, France)
Bianka Buschbeck (Systran, France)
Tony Clarke (CLS Communication, Switzerland)
Béatrice Daille (University of Nantes, France)
Heidi Depraetere (Cross Language, Belgium)
Ilse Depraetere (Université de Lille III, France)
Mike Dillinger (Translation Optimization Partners, US)
Stephen Doherty (CNGL, Ireland)
Marc Dymetman (Xerox, France)
Andreas Eisele (EC, Luxemboug)
Ray Flournoy (Adobe Systems, US)
Jesús Giménez (Nuance, Spain)
Steve Götz (CNGL, Ireland)
Daniel Grasmick (Lucy Software and Services, Germany)
Declan Groves (Microsoft, Ireland)
Ana Guerberof (Universitat Autònoma de Barcelona, Spain)
Rafael Guzman (Symantec, Ireland)
Olivier Hamon (Syllabs, France)
Viggo Hansen (EAMT Executive Committee, Denmark)
Fred Hollowood (Fred Hollowood Consulting, Ireland)
Dorothy Kenny (CNGL, Dublin City University, Ireland)
Qun Liu (CNGL, Dublin City University, Ireland)
John Moran (CNGL, Ireland)
Sharon O'Brien (CNGL, Dublin City University, Ireland)
Sergio Pelino (Google, US)
Sergio Penkale (Lingo24, UK)
Mirko Plitt (Modulo Language Automation, Switzerland)
Bruno Pouliquen (WIPO, Switzerland)
Alexandros Poulis (Lionbridge, Finland)
Manny Rayner (University of Geneva, Switzerland)
Rubén Rodríguez de la Fuente (Paypal)
Raphael Rubino (Prompsit Language Engineering, Spain)

Marta Ruiz Costa-Jussà (Catalan Politechnic University, Spain)
Dag Schmidtke (Microsoft Ireland, Dublin, Ireland)
Jean Senellart (Systran, France)
Violeta Seretan (University of Geneva, Switzerland)
Svetlana Sheremetyeva (LanA Consulting, ApS, Denmark; South Ural State University, Russia)
Yanli Sun (Symantec, China)
Midori Tatsumi (Translator, Japan)
Chris Wendt (Microsoft Research, US)
Francois Yvon (University Paris Sud 11, France)
Ventsislav Zhechev (Autodesk, Switzerland)
Jost Zetsche (International Writers' Group, US)
Andy Way (CNGL, Dublin City University, Ireland)

Organizing Committee

Božo Bekavac (University of Zagreb, Croatia)
Mikel Forcada (Universitat d'Alacant, Spain)
Philipp Koehn (University of Edinburgh, UK)
Johann Roturier (Symantec, Ireland)
Marko Tadić (University of Zagreb / Croatian Language Technologies Society, Croatia)
Andy Way (CNGL, Dublin City University, Ireland)

Sponsors

Silver sponsor

Bloomberg

Bronze sponsor

EUROPEAN LANGUAGE RESOURCES ASSOCIATION

ELRA

Sponsor of the Best Paper Award

Springer

Supported by

The Ministry of Science, Education and Sports
of the Republic of Croatia

Invited talk

Encountering the Unknown, Part 2

Jost Zetzsche
International Writers' Group

At the AMTA conference four years ago in Denver, I challenged both translators and the MT community by presenting them with "task lists" of items that would help them build bridges to each other.

The tasks that the translators were "charged" with were to look back at previous responses to technology, put into perspective what MT is in relation to other technologies, differentiate between different forms of MT, employ MT where appropriate, and embrace their whole identity.

The MT community was asked to acknowledge the origin of data and linguistic expertise it uses, communicate in terms that are down to earth and truthful, engage the translation community in meaningful ways, listen to the translation community, and embrace their whole identity.

For this presentation I will attempt to evaluate how the two sides have done, what other tasks might need to be added, and whether there actually are still two sides.

I have collected feedback from the greater community of translators for this presentation.

Poster Session A
Research and Product/Project Papers

Incorporating Paraphrasing in Translation Memory Matching and Retrieval

Rohit Gupta and **Constantin Orăsan**
RGCL, Research Institute of Information and Language Processing,
University of Wolverhampton, Stafford Street,
Wolverhampton WV11LY, UK
{R.Gupta, C.Orasan}@wlv.ac.uk

Abstract

Current Translation Memory (TM) systems work at the surface level and lack semantic knowledge while matching. This paper presents an approach to incorporating semantic knowledge in the form of paraphrasing in matching and retrieval. Most of the TMs use Levenshtein edit-distance or some variation of it. Generating additional segments based on the paraphrases available in a segment results in exponential time complexity while matching. The reason is that a particular phrase can be paraphrased in several ways and there can be several possible phrases in a segment which can be paraphrased. We propose an efficient approach to incorporating paraphrasing with edit-distance. The approach is based on greedy approximation and dynamic programming. We have obtained significant improvement in both retrieval and translation of retrieved segments for TM thresholds of 100%, 95% and 90%.

1 Introduction

Translation Memories (TMs) are tools commonly used by professional translators to speed up the translation process. The concept of TM can be traced back to 1978 when Peter J. Arthern proposed the use of a translation archive (Arthern, 1978). A TM system helps translators by retrieving previously translated segments to extract the relevant match for reuse. TMs also help them in maintaining the consistency with previous work and use of appropriate terminology. Lagoudaki (2006) surveyed the use of TMs by professional translators in 2006, and 721 out of 874 (82.5%) replies confirmed the use of a TM.

Although, extensive research has been done in Natural Language Processing (NLP) with emphasis on improving the performance of automatic Machine Translation (MT), there is not much research on improving the TM systems by using NLP techniques. So far, most of the research in TM has been carried out mostly in industry with more focus on improving user interface and user experience in general rather than employing language technology to improve matching and retrieval. Recent research (Koehn and Senellart, 2010; Zhechev and Genabith, 2010) on TM focus more on improving machine translation using TMs.

The TMs currently used by translators find matches for a given segment on the basis of surface form. This means that even if a paraphrased segment is available in the TM, the TM systems have no way to retrieve such segments. In this paper we try to mitigate this problem by using existing paraphrase databases. To achieve this, we have incorporated paraphrasing in the TM matching process. A trivial approach to incorporating paraphrasing would be to generate all the possible segments based on paraphrases available. However in this approach the number of segments increases exponentially and hence can not be applied in our task. This paper proposes a greedy approximation and dynamic programming technique to incorporate paraphrasing in the matching algorithm.

2 Paraphrasing for TM

2.1 Existing Work

The idea of incorporating paraphrasing or semantic features at the conceptual level is not new. Work done by (Pekar and Mitkov, 2007) and (Mitkov, 2008) explores the issues in TM systems. Although these works present good insight into TM systems and their limitations, there is no feasible practical implementation proposed to improve them. Another work (Utiyama et al., 2011) incorporates paraphrasing into TM. This approach uses a statistical framework to integrate paraphrasing which requires corpora from the same domain with an abundance of similar segments. The downside of this approach is that it requires generation of all the additional segments based on paraphrases which is inefficient both in terms of time and space. In addition, the approach was used to get exact matches only. In SMT, Onishi et al. (2010) and Du et al. (2010) use paraphrasing lattice to improving MT by gaining more coverage.

2.2 Need for Paraphrasing

Current TM systems work on the surface level with no linguistic information. Because of this often the paraphrased segments available in the TM are either not retrieved or retrieved with a very low threshold and are ranked incorrectly among the retrieved segments. The lack of semantic knowledge in the matching process also leads to cases where, for the same similarity score shown by the system, one segment may require little effort while another requires more in terms of post editing. For example, even though segments like "the period laid down in article 4(3)" and "the duration set forth in article 4(3)" have the same meaning, the one segment may not be retrieved for another in current TM systems as having only 57% similarity based on edit-distance as implemented in OmegaT[1]. In this case we can see that one segment is a paraphrase of the another segment. To mitigate this limitation of TM, we propose an approach to incorporating paraphrasing in TM matching without compromising the beauty of edit-distance which has been trusted by translators, translation service providers and TM developers over the years.

2.3 PPDB:The Paraphrase Database

The PPDB 1.0 paraphrases database (Ganitkevitch et al., 2013) contains lexical, phrasal and syntactic paraphrases automatically extracted using a large collection of parallel corpora. This database comes in six sizes (S, M, L, XL, XXL, XXXL) where S is the smallest and XXXL is the largest. The smaller packages contain only high precision paraphrases, while the larger ones aims at more coverage. We have used lexical and phrasal paraphrases of "L" size for our approach. The reason for choosing L size was to retain the quality of segments retrieved using paraphrasing and at the same time gain some coverage.

2.4 Classification of Paraphrases

We have classified paraphrases obtained from PPDB 1.0 into four types for our implementation on the basis of the number of words in the source and target phrases. These four categories are as follows:

1. Paraphrases having one word on both the source and target sides, e.g. "period" ⇒"duration"

2. Paraphrases having multiple words on both sides but differing in one word only, e.g. "in the period" ⇒ "during the period"

3. Paraphrases having multiple words as well as same number of words on both sides, e.g. "laid down in article" ⇒ "set forth in article"

4. Paraphrases in which the number of words on the source and target sides differ, e.g. "a reasonable period of time to" ⇒ "a reasonable period to"

As we have already pointed out, a trivial approach to implementing paraphrasing along with edit-distance is to generate all the paraphrases based on the paraphrases available and store these additional segments in the TM. This approach is highly inefficient both in terms of time and space. For example, for a TM segment which has four different phrases where each phrase can be paraphrased in five more possible ways, we get 1295 (6^4 -1) additional segments (still not considering that these phrases may contain paraphrases as well) to store in the TM, which is inefficient even for small TMs. To handle this problem,

[1] OmegaT is an open source TM available form http://www.omegat.org

each class of paraphrases is processed in a different manner. In our classification, Type 1 are one-word paraphrases and Type 2 can be reduced to one-word paraphrases after considering the context when storing in the TM. For Type 1 and Type 2, we get the same accuracy as the trivial method in polynomial time complexity (see Section 3 for details). Paraphrases of Type 3 and Type 4 require additional attention because they still remain multiword paraphrases after reduction and greedy approximation is needed to implement them in polynomial time.

3 Our Approach

A general approach for TM matching and retrieval is as follows:

1. Read the Translation Memories available
2. Read the file that needs to be translated
3. Preprocess the input file, apply filter for different file formats and identify the segments
4. For each segment in the input file search for the most similar segment in TM and retrieve the most similar segment if above a predefined threshold
5. For each segment in the input file display the input segment along with the most similar segment to the translator for post-editing

There are two options for incorporating paraphrasing in this pipeline: paraphrase the input or paraphrase the TM. For our approach we have chosen to paraphrase the TM. There are many reasons for this. First, once a system is set up, the user can get the retrieved matches in real time; second, TMs can be stored in company servers and all processing can be done offline; third, the TM system need not be installed on the user computer and can be provided as a service.

For our implementation we used the open source TM tool OmegaT, which uses word-based edit-distance with cost 1 for insertion, deletion and substitution. We have employed OmegaT edit-distance as a baseline and adapted this to incorporate paraphrasing so that at a later stage we can add this feature in OmegaT without compromising the confidence users have in OmegaT fuzzy matches.

Our approach can be briefly described as the following steps:

1. Read the Translation Memories available
2. Collect all the paraphrases from the paraphrase database and classify them according to the classes presented in Section 2.4
3. Store all the paraphrases for each segment in the TM in their reduced forms according to the process presented in Section 3.1
4. Read the file that needs to be translated
5. For each segment in the input file get the potential segments for paraphrasing in the TM according to the filtering steps of Section 3.2 and search for the most similar segment based on approach described in Section 3.3 and retrieve the most similar segment if above a predefined threshold

3.1 Storing Paraphrases

The paraphrases are stored in the TM in their reduced forms as after capturing paraphrases for a particular segment we have already considered the context and there is no need for it to be considered again while calculating edit-distance. We store only the longest uncommon substring instead of the whole paraphrase. This reduced paraphrase is stored with the source word where the uncommon substring starts. We refer to this source word as “token”. Table 1 shows the TM source segment (TMS), paraphrases captured for this segment (TMP) and paraphrases stored in their reduced form (TMR). In this case, the token “period” stores the two paraphrases “duration” and “time” and the token “laid” stores the two paraphrases “referred to” and “provided for by”. For Type 3 and Type 4 the paraphrase source length (represented by ls in Table 1) is also stored along with the paraphrase (represented by tp in Table 1). In this case, length “2” for “laid down” is stored with paraphrase “referred to” and length “3” for “laid down in” is stored along with paraphrase “provided for by”.

3.2 Filtering

Before processing begins, for each input segment certain filtering steps are applied in order to speed up the process. The purpose of this preprocessing is to filter out unnecessary

TMS	the	period	laid down in article			4(3)	of	decision	468
TMP	the	period duration time	laid down referred to provided for	in in by	article article article	4(3)	of	decision	468
TMR	the	period duration time	laid ls tp 2 referred to 3 provided for by	down in article		4(3)	of	decision	468

Table 1: Representing paraphrases in TM

candidates for participating in the paraphrasing process. Because we are generally interested in candidates above a certain threshold it is obvious to filter out candidates below a certain threshold. Our filtering steps for getting potential candidates for paraphrasing are as follows:

- We first filter out the segments based on length because if segments differ considerably in length, the edit-distance will also differ. In our case, the threshold for length was 49%. So, the TM segments which are shorter than 49% of the input are filtered.
- Next, we filter out the segments based on baseline edit-distance similarity. The TM segments which are having a similarity below a certain threshold will be removed. In our case, the threshold was 49%.
- Next, after filtering the candidates with the above two steps we sort the remaining segments in decreasing order of similarity and pick the top 100 segments.
- Finally segments within a certain range of similarity with the most similar segment were selected for paraphrasing. In our case, the range is 35%. This means that if the most similar segment has 95% similarity, segments with a similarity below 60% will be discarded[2].

3.3 Matching and Retrieval

For matching, similarity is calculated with the potential segments for paraphrasing extracted as per Section 3.2. Type 1 and Type 2 paraphrases after reduction (as per Section 3.1) are single-word paraphrases and Type 3 and Type 4 paraphrases have multiple words. For Type 1 and Type 2 the edit-distance procedure can be optimised globally as this is a simple case of matching one of these "paraphrases" when calculating the cost of substitution. For the example given in Table 1, if a word from input segment matches any of the words "period", "time" or "duration", the cost of substitution will be 0.

For paraphrases of Types 3 and 4 the algorithm takes the decision locally at the point where all paraphrases finish. The basic edit-distance calculation procedure is given in Algorithm 1. The algorithm elaborating our decision-making process is given in Algorithm 2. In Algorithm 2, $Input$ is the segment that we want to translate and TMS is the TM segment. Table 2 shows the edit-distance calculation of the first five tokens of the Input and TM segment with paraphrasing. In Algorithm 2, *lines 11 to 22* executes when Type 3 and Type 4 paraphrases are not available (e.g. edit-distance calculation of the second token "period"). *Lines 24 to 57* account for the case when Type 3 and Type 4 paraphrases are available. *Line 28* calculates the edit-distance of the corresponding longest source phrase and stores it in DS matrix as shown in Algorithm 2 (e.g. calculation of the edit-distance of "laid down in" in Table 2). *Lines 33 to 46* account for the edit-distance calculation of each paraphrase (e.g. calculation of "referred to" and "provided for by" in Table 2). The edit-distance of each paraphrase is stored in DTP matrix as shown in Algorithm 2. *Lines 38 to 46* account for the selection of the minimum edit-distance paraphrase or source phrase. At *line 38*, the algorithm compares the edit-distance of paraphrase DTP (e.g. "referred to") with the edit-distance of the corresponding source phrase (e.g. "laid down") as well as with the current minimum distance. *Lines 48, 52 and 56* account for updating the value of j to reflect the current position for further calculation of edit-

[2] these thresholds were determined empirically

Algorithm 1 Basic Edit-Distance Procedure

```
procedure EDIT-DISTANCE($Input$,$TMS$)
    $M \leftarrow$ length of $TMS$                                    ▷ Initialise $M$ with length of TM segment
    $N \leftarrow$ length of $Input$                                  ▷ Initialise $N$ with length of Input segment
    $D[i, 0] \leftarrow i$ for $0 \leq i \leq N$                      ▷ initialisation
    $D[0, j] \leftarrow j$ for $0 \leq j \leq M$                      ▷ initialisation
    for $j \leftarrow 1...M$ do
        $TMToken \leftarrow TMS_j$                                    ▷ get Token of TM segment
        for $i \leftarrow 1...N$ do
            $InputToken \leftarrow InputSegment_i$                    ▷ get Token of Input segment
            if $InputToken = TMToken$ then                            ▷ match $InputToken$ with $TMToken$
                $substitutionCost \leftarrow 0$                       ▷ Substitution cost if matches
            else
                $substitutionCost \leftarrow 1$                       ▷ Substitution cost if not matches
            $D[i, j] \leftarrow minimum(D[i-1, j] + insertionCost, D[i, j-1] + deletionCost, D[i-1, j-1] + substitutionCost)$    ▷ store minimum of insertion, substitution and deletion
    Return $D[N, M]$                                                  ▷ Return minimum edit-distance
end procedure
```

	j	0	1	2				3	4				5
i		#	the	period duration time	laid	down	in	referred	to	provided	for	by	in
0	#	0	1	2	3	4	5	3	4	3	4	5	5
1	the	1	0	1	2	3	4	2	3	2	3	4	4
2	period	2	1	0	1	2	3	1	2	1	2	3	3
3	referred	3	2	1	1	2	3	0	1	1	2	3	2
4	to	4	3	2	2	**2**	3	1	**0**	2	2	3	1
5	in	5	4	3	3	3	**3**	2	1	3	3	**3**	0

Table 2: Edit-Distance Calculation using Algorithm 2

distance (e.g. $j = 5$ after selecting "referred to") and *lines 50, 54 and 57* update the matrix D as shown in Algorithm 2.

As we can see in Table 2, starting from the third token of the TM, "laid", three separate edit-distances are calculated, two for the two paraphrases "referred to" and "provided for by" and one for the corresponding longest source phrase "laid down in" and the paraphrase "referred to" is selected as it gives a minimum edit-distance of 0. The last column of Table 2 ($j = 5$) shows the edit-distance calculation of the next token "in" after selecting "referred to".

3.4 Computational Considerations

The time complexity of the basic edit-distance procedure is $O(mn)$ where m and n are lengths of source and target segments, respectively. After employing paraphrasing of Type 1 and Type 2 the complexity of calculating the substitution cost increases from $O(1)$ to $O(log(p))$ (as searching the p words takes $O(log(p))$ time) where p is the number of paraphrases of Type 1 and Type 2 per token of TM source segment, which increases the edit-distance complexity to $O(mnlog(p))$. Employing paraphrasing of Type 3 and Type 4 further increases the edit-distance complexity to $O(lmn(log(p) + q))$, where q is the number of Type 3 and Type 4 paraphrases stored per token and l is the average length of paraphrase. Assuming the source and target segment are of same length n and each token of the segment stores paraphrases of length l, the complexity will be $O((q + log(p))n^2 l)$. By limiting the number of paraphrases stored per token of the TM segment we can replace $(q + log(p))$ by a constant c. In this case complexity will be $c \times O(n^2 l)$. However, in practice it will take less time as not all tokens in the TM segment will have p and q paraphrases and the paraphrases are also stored in the reduced form.

4 Experiments and Results

For our experiments we have used English-French pairs of the 2013 release of the DGT-TM corpus (Steinberger et al., 2012). The corpus was se-

Algorithm 2 Edit-Distance with paraphrasing procedure

```
procedure EDIT-DISTANCEPP(Input,TMS)
    $M \leftarrow length(TMS)$    ▷ number of tokens in TM segment
    $N \leftarrow length(Input)$    ▷ number of tokens in Input segment
    $D[i,0] \leftarrow i$ for $0 \leq i \leq N$    ▷ initialise two dimensional matrix $D$
    $D[0,j] \leftarrow j$ for $0 \leq j \leq (M + p')$ where $p'$ accounts for increase in TM segment length because of paraphrasing
    $decisionPoint \leftarrow 0$ , $j \leftarrow 1$
    $scost \leftarrow 1, dcost \leftarrow 1, icost \leftarrow 1$    ▷ initialisation of substitution, deletion and insertion cost
    while $j \leq M$ do
        $t \leftarrow TMS_j$    ▷ getting current TM token to process, e.g. $3^{rd}$ token “laid”
        if $t$ has no paraphrases of type 3 and type 4 or $decisionPoint \geq N$ then
            $decisionPoint \leftarrow decisionPoint + 1, j \leftarrow j + 1$
            for $i \leftarrow 1...N$ do
                $InputToken \leftarrow Input_i$
                if $InputToken = t$ then
                    $scost \leftarrow 0$
                else
                    $scost \leftarrow 1$
                if $scost = 1$ then
                    $OneWordPP \leftarrow getOneWordPP(t)$    ▷ get one word paraphrases associated with TM token $t$
                    if $InputToken \in OneWordPP$ then    ▷ applying type 1 and type 2 paraphrasing
                        $scost \leftarrow 0$
                $D[i, decisionPoint] \leftarrow minimum(D[i, decisionPoint - 1] + dcost, D[i - 1, decisionPoint] + icost, D[i - 1, decisionPoint - 1] + scost)$
        else
            $tp \leftarrow$ get paraphrases stored at $t$    ▷ e.g. $tp$ for Token “laid” in Table 1
            $ls \leftarrow$ get corresponding source lengths stored at $t$    ▷ e.g. $ls$ for Token “laid” in Table 1
            $lsmax \leftarrow$ length of longest source phrase
            $DS[0, l - 1] \leftarrow D[0, decisionPoint + l]$ for $1 \leq l \leq lsmax$    ▷ initialise two dimensional matrix $DS$ to calculate edit-distance of longest source phrase
            $DS \leftarrow$ calculate edit-distance of longest source phrase with $Input$ using $D$    ▷ uses $D$ for first word, consider Type 1 and Type 2 paraphrases
            $P \leftarrow$ number of paraphrases of type 3 and type 4    ▷ E.g. 2 for “laid”
            $index \leftarrow 0, paraphraselen \leftarrow 0, isppwin \leftarrow false, curDistance \leftarrow \infty$
            $prevDistance \leftarrow D[decisionPoint, decisionPoint]$
            $DTP[k, 0, l - 1] \leftarrow D[0, decisionPoint + l]$ for $0 \leq k \leq P - 1$ for $1 \leq l \leq length(tp[k])$▷ initialise three dimensional matrix $DTP$ to calculate edit-distances of paraphrases
            for $k \leftarrow 0...P - 1$ do
                $dps[k] \leftarrow decisionPoint + ls[k]$
                $ltp \leftarrow length(tp[k])$    ▷ get paraphrase length e.g. 2 for “referred to”
                $dpt[k] \leftarrow decisionPoint + ltp$
                $DTP[k] \leftarrow$ calculate edit-distance of $tp[k]$ with $Input$ using $D$    ▷ uses $D$ for first word of $tp[k]$
                if $DTP[k, ltp - 1, dpt[k]] < DS[ls[k] - 1, dps[k]]$ and $DTP[k, ltp - 1, dpt[k]] < curDistance$ then
                    $ppwin \leftarrow true$
                    $curDistance \leftarrow DTP[k, ltp - 1, dpt[k]]$
                    $index \leftarrow k$
                    $paraphraselen \leftarrow ltp$
                else if $DS[ls[k] - 1, dps[k]] < curDistance$ then
                    $ppwin \leftarrow false$
                    $curDistance \leftarrow DS[ls[k] - 1, dps[k]]$
                    $index \leftarrow k$
            if $ppwin = true$ then    ▷ $true$ if paraphrase is better
                $j \leftarrow j + ls[index]$
                $decisionPoint \leftarrow decisionPoint + paraphraselen$
                update $D$ using $DTP[index]$
            else if $curDistance = prevDistance$ then    ▷ $true$ if source phrase is better and exactly matching
                $j \leftarrow j + ls[index]$
                $decisionPoint \leftarrow decisionPoint + ls[index]$
                update $D$ using $DS$
            else
                $j \leftarrow j + 1, decisionPoint \leftarrow decisionPoint + 1$
                update $D$ using $DS$
    Return $D[N, decisionPoint]$
end procedure
```

lected in such a way that it was not used to produce PPDB. For this reason, its language may be slightly different from the one used to produce PPDB, which may be a reason for the relatively modest results obtained in this paper. In our case English was the source language and French was the target language. From this corpus we have filtered out segments of fewer than five words and remaining pairs were used to create the TM and Test dataset. Tokenization of the English data was done using Berkeley Tokenizer (Petrov et al., 2006). Table 3 shows our corpus statistics. In our case, average number of phrases per TM segment for which paraphrases are present in PPDB is 37 (AvgPhrases) and average number of paraphrases per TM segment present in PPDB is 146 (AvgPP) as shown in the Table 3.

	TM	**Test**
Segments	319709	25000
Source words	8200796	640265
Target words	7807577	609165
Average source length	25.65	25.61
Average target length	24.42	24.36
AvgPhrases	37	
AvgPP	146	

Table 3: Corpus Statistics

TH	**100**	**95**	**90**	**85**	**80**
EDR	6352	7062	8369	9829	10730
PPR	6444	7172	8476	9938	10853
Imp	1.45	1.56	1.28	1.11	1.15
RC	13	20	43	68	88
BPP	**74.31**	**73.16**	**65.01**	63.29	60.84
BED	65.89	70.29	60.70	63.29	61.31

Table 4: Results on surface form: Using all four types of paraphrases

TH	**100**	**95**	**90**	**85**	**80**
EDR	6352	7062	8369	9829	10730
PPR	6421	7142	8450	9915	10820
Imp	1.09	1.13	0.97	0.87	0.84
RC	8	13	27	45	55
BPP	**73.18**	**73.98**	**63.08**	64.37	**63.37**
BED	60.86	71.43	61.96	65.10	63.28

Table 5: Results on surface form: Using paraphrases of Types 1 and 2 only

TH	**100**	**95**	**90**	**85**	**80**
EDR	8179	8675	9603	10456	11308
PPR	8294	8802	9735	10597	11462
IMP	1.41	1.46	1.37	1.35	1.36
RC	21	30	43	73	108
BPP	**68.61**	**78.04**	**75.40**	**69.06**	**63.93**
BED	59.89	67.88	66.32	63.57	61.92

Table 6: Results with placeholders: Using all four types of paraphrases

TH	**100**	**95**	**90**	**85**	**80**
EDR	8179	8675	9603	10456	11308
PPR	8277	8777	9706	10568	11422
IMP	1.2	1.18	1.07	1.07	1.01
RC	19	24	30	49	73
BPP	**58.28**	**67.95**	**71.03**	**68.03**	**61.02**
BED	52.00	54.81	60.09	62.13	57.42

Table 7: Results with placeholders: Using paraphrases of Types 1 and 2 only

Our evaluation has two objectives: first to see how much impact paraphrasing has in terms of retrieval and second to see the translation quality of those segments which changed their ranking and brought them up to the top because of the paraphrasing. The results of our evaluations are given in Tables 4, 5, 6, and 7 where each table shows the similarity threshold for TM (TH), the total number of segments retrieved using the baseline approach (EDR), the total number of segments retrieved using our approach (PPR), the percentage improvement in retrieval obtained over the baseline (Imp), the number of segments which changed their ranking and come up to the top because of paraphrasing (RC), the BLEU score (Papineni et al., 2002) on target side over translations retrieved by our approach for segments which changed their ranking and come up to the top because of paraphrasing (BPP) and the BLEU score on target side over corresponding translations retrieved (irrespective of similarity score) by baseline approach for these segments (BED).

As we can see in Table 4, on surface form for a threshold of 90% we got a 1.28% improvement over baseline in terms of retrieval, i.e. we have retrieved 107 more segments. We can observe an increase of more than four BLEU points for the

90% threshold and an increase of more than eight BLEU points for the 100% threshold for the segments which change their rank. There are 13 segments for threshold 100% which change their rank and 43 segments for threshold 90% which change their rank. Table 5 shows improvements we have obtained using paraphrases of Types 1 and 2 only.

To get more matches in TM, which is usually the case for real TM, we have removed punctuation and replaced numbers and dates with placeholders. For this experiment we observed significant improvement for a threshold of 80% and above as shown in Tables 6 & 7. We can observe that after removing punctuation and replacing numbers and dates with placeholders we obtained more than five BLEU points improvement over the baseline for a threshold of 85% and above for the segments which changes their rank.

Table 7 shows the improvements we have obtained using paraphrases of Type 1 and 2 only with placeholders. As we can see, improvements in retrieval is less compared to Table 6 which uses all paraphrases but the BLEU score is still improving significantly. We can observe an increase of more than 10 BLEU points over the baseline for thresholds of 95% and 90% .

5 Conclusion and Future work

We have presented an efficient approach to incorporating paraphrasing in TM. The approach is simple and fast enough to implement in practice. We have also shown that incorporating paraphrasing significantly improves TM matching and retrieval. Apart from TM, the approach can also be used for other natural language processing tasks (e.g. to incorporate paraphrasing in sentence semantic similarity measures exploiting edit-distance).

In future, we would like to consider the syntactic structure of the paraphrases when performing matching and retrieval, and also to take into account the context in which the paraphrases are used in order to have better accuracy. Alternative ways to implement using Finite State Transducers (FST) can also be considered and compared.

Acknowledgement

The research leading to these results has received funding from the People Programme (Marie Curie Actions) of the European Union's Seventh Framework Programme FP7/2007-2013/ under REA grant agreement no. 317471.

References

Arthern, Peter J. 1978. Machine Translation and Computerized Terminology Systems, A Translator's viewpoint. In *Translating and the Computer: Proceedings of a Seminar*, pages 77–108.

Du, Jinhua, Jie Jiang, and Andy Way. 2010. Facilitating Translation Using Source Language Paraphrase Lattices. In *Proceeding of EMNLP*, pages 420–429.

Ganitkevitch, Juri, Van Durme Benjamin, and Chris Callison-Burch. 2013. Ppdb: The paraphrase database. In *Proceedings of NAACL-HLT*, pages 758–764, Atlanta, Georgia. Association for Computational Linguistics.

Koehn, Philipp and Jean Senellart. 2010. Convergence of translation memory and statistical machine translation. In *Proceedings of AMTA Workshop on MT Research and the Translation Industry*, pages 21–31.

Lagoudaki, Elina. 2006. Translation Memories Survey 2006: Users' perceptions around TM use. In *Proceedings of Translating and the Computer 28*, pages 1–29, London. Aslib.

Mitkov, Ruslan. 2008. Improving Third Generation Translation Memory systems through identification of rhetorical predicates. In *Proceedings of LangTech2008.*

Onishi, Takashi, Masao Utiyama, and Eiichiro Sumita. 2010. Paraphrase Lattice for Statistical Machine Translation. In *Proceeding of the ACL*, pages 1–5.

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the ACL*, pages 311–318.

Pekar, Viktor and Ruslan Mitkov. 2007. New Generation Translation Memory: Content-Sensivite Matching. In *Proceedings of the 40th Anniversary Congress of the Swiss Association of Translators, Terminologists and Interpreters.*

Petrov, Slav, Leon Barrett, Romain Thibaux, and Dan Klein. 2006. Learning accurate, compact, and interpretable tree annotation. In *Proceedings of the COLING/ACL*, pages 433–440.

Steinberger, Ralf, Andreas Eisele, Szymon Klocek, Spyridon Pilos, and Patrick Schlüter. 2012. DGT-TM: A freely available Translation Memory in 22 languages. *LREC*, pages 454–459.

Utiyama, Masao, Graham Neubig, Takashi Onishi, and Eiichiro Sumita. 2011. Searching Translation Memories for Paraphrases. In *Machine Translation Summit XIII*, pages 325–331.

Zhechev, Ventsislav and Josef Van Genabith. 2010. Seeding statistical machine translation with translation memory output through tree-based structural alignment. In *Proceedings of ACL*, pages 43–51.

Combining Bilingual Terminology Mining and Morphological Modeling for Domain Adaptation in SMT

Marion Weller[1,2] **Alexander Fraser**[2] **Ulrich Heid**[3]

[1]Institut für Maschinelle Sprachverarbeitung
Universität Stuttgart
weller@ims.uni-stuttgart.de

[2]Centrum für Informations- und Sprachverarbeitung
LMU München
fraser@cis.uni-muenchen.de

[3]Institut f. Informationswissenschaft u. Sprachtechnologie
Universität Hildesheim
heid@uni-hildesheim.de

Abstract

Translating in technical domains is a well-known problem in SMT, as the lack of parallel documents causes significant problems of sparsity. We discuss and compare different strategies for enriching SMT systems built on general domain data with bilingual terminology mined from comparable corpora. In particular, we focus on the target-language inflection of the terminology data and present a pipeline that can generate previously unseen inflected forms.

1 Introduction

Adapting statistical machine translation (SMT) systems to a new domain is difficult when the domain lacks sufficient amounts of parallel data, as is the case in many technical or medical domains. SMT systems trained on general language (e.g. government proceedings) face data-sparsity issues when translating texts from such domains, particularly if translating into a morphologically rich language.

In this paper, we compare different strategies to adapt an EN-FR SMT system built on Europarl to a technical domain (wind energy) by making use of term-translation pairs mined from comparable domain-specific corpora. In a first series of experiments, we study two methods of integrating bilingual terminology into a phrase-based SMT system: adding term translation pairs via XML mark-up and as pseudo-parallel training data. In particular, we compare the effects of integrating translation candidates for multi-word terms vs. single-word terms and show that the use of single-word terms can be harmful. Using bilingual terminology in the form of pseudo-parallel data significantly outperforms the the baseline.

However, it also becomes evident that terminology handling requires morphological modeling: when the integrated term-translation pairs are restricted to the inflected forms seen in the (domain-specific) data, this ignores the fact that other forms might be needed when translating. Furthermore, translation-relevant morphological features (e.g. number) must be maintained during the translation process. As a way to address these problems, we present a morphology-aware translation system that treats inflection as a target-side generation problem. Combining the integration of term-translation pairs and the modeling of target-side morphology allows for the generation of unseen word forms and the preservation of translation-relevant features. In the second part of the paper, we describe and discuss a novel pipeline for morphology-aware integration of bilingual terminology. While this system's improvement over the baseline is not statistically significant, our analysis highlights the need for explicit morphological modeling, which, as far as we know, has not been addressed previously.

Issues in translating out-of-domain data. When translating texts of domains that are not well represented by the training data, there are two main problems: (i) data sparsity: many domain-specific words do not appear in the parallel data and thus cannot be translated (e.g. the English term *torque* which does not occur in Europarl), and (ii) polysemy: words can have different meanings when used in general vs. specialized language. For example, the word *boss* means either *manager* or refers to a rivet-type object. In a general language text, the meaning of *manager* is predominant, whereas in a text of a technical domain, that sense is less likely to be correct. Because a translation model trained on general language data learns that *boss* → *manager* is a good translation, this translation is likely to be used when translating data from a technical domain. In order to make previously unknown terms available and to model domain-specific preferences, we enrich the SMT system with domain-specific term-translation pairs that are not contained in the general language parallel data.

Modeling morphology. Another type of data sparsity occurs in translations to languages with rich (noun) inflection, as the parallel training data is unlikely to cover the full inflection paradigms of all words. As a result, some inflected forms are unavailable to the SMT system. This problem increases considerably when translating terms which are not well represented in the parallel training data, as is the case in the domain-adaptation scenario presented in this work. Modeling target-side morphology helps to reduce this kind of data-sparsity: we present a two-step approach, in which we separate the translation process from target-side inflection by first translating into a lemmatized representation, with a post-processing component for generating inflected forms. This simplifies the translation task, as information concerning only the target language has been removed. Also, this two-step approach allows us to generate forms which are not contained in the parallel data, which is of particular interest for domain-adaptation scenarios, where the full inflectional paradigm of term-translation pairs might not even be covered by the domain-specific data used for term mining. Furthermore, this setup allows us to specifically indicate how a term in a given context should be translated. For example, it provides the means to guarantee that a source-language term in plural is translated by the corresponding target-language term in plural, regardless of whether the required inflected form occurs in the training data. Although there are exceptions such as $\textit{furniture}_{SG}$ → $\textit{meubles}_{PL}$, we believe they play a negligible role when translating under-resourced domains.

2 Related work

There has been considerable interest in mining translations directly from comparable corpora. A few representative examples are (Daille and Morin, 2005; Haghighi et al., 2008; Daumé III and Jagarlamudi, 2011; Prochasson and Fung, 2011), all of which mine terms using distributional similarity. These approaches tend to favor recall over precision. In contrast, we use a high-precision method consisting in recognizing term candidates by means of part-of-speech patterns with an alignment method relying on dictionary entries (Weller and Heid, 2012).

A second strand of relevant work is the integration of terms into SMT decoding. Hálek et al. (2011) integrated named entity translations mined from Wikipedia using the XML mode of Moses, which creates new phrase table entries dynamically. Pinnis and Skadins (2012) also studied mining named entities, as well as using a high quality terminological database, and added these resources to the parallel training data. We compare these two options (XML vs. added parallel data) and show that adding the terms to the parallel training data leads to better results.

To deal with the issue of obtaining the proper inflection of mined terms, we implemented a morphology-aware English to French translation system that separates the translation task into two steps (translation + inflection generation), following Toutanova et al. (2008) and Fraser et al. (2012).

Formiga et al. (2012) use a component for target-side morphological generation to translate news and web-log data. In contrast to our work, they do not deal with nominal morphology, but model verb inflection: this is important for web-log data, as second-person verb forms rarely appear in Europarl-type training data. Wu et al. (2008) use dictionary entries for adapting a system trained on Europarl to news, but without applying morphological modelling to their EN-FR system. Furthermore, news and also web-log data are considerably more similar to Europarl than technical data.

Our main contribution is that we show how to combine three areas of research: bilingual term mining, using terms in SMT, and generation of inflection for SMT. We describe a novel end-to-end morphology-aware solution for using bilingual term mining in SMT decoding.

3 Bilingual terminology mining

In contrast to parallel corpora, which are difficult to obtain in larger quantities, comparable corpora of a particular domain are relatively easy to obtain. Comparable corpora are expected to have similar content and consequently similar domain-specific terms in both languages and thus constitute a suitable basis for the mining of term-translation pairs.

For both source and target-language, term candidates are extracted based on part-of-speech patterns, focusing on nominal phrases. The resulting sets of term candidates are then aligned.

We use all available domain-specific training data (cf. section 7) for monolingual term extraction on the target language. Source language terms are only extracted for the input data to the SMT system (tuning/test set) because our methods for term integration are restricted to terms contained in the sentences to be translated.

Term alignment. The task of term alignment consists in finding the equivalent of a source language term in a set of target language terms. One method is *pattern-based compositional term-alignment*: all components of a multi-word term are first translated individually using a (general language) dictionary, and then recombined according to handcrafted translation patterns such as
(EN) `noun1 noun2` ↔ (FR) `noun2` *de* `noun1`[1].
As the recombination of individual translations leads to over-generation, the generated translation candidates are filtered against the list of extracted target-language terms. A principal assumption is that the term pairs are semantically transparent and of a similar morpho-syntactic structure. The example for the term *glass fibre* illustrates the process:

(1) individual translation:
`noun1`: *glass* → *verre* (*glass*),
loupe (*magnifying glass*)
`noun2`: *fibre* → *fibre*

(2) recombination[2] of translations:
fibre de verre, *fibre de loupe*

(3) filtering against target terms:
fibre de verre, ~~*fibre de loupe*~~

Source and target terms are not necessarily of the same word class; such shifts are dealt with by simple morphological rules, as shown by the term
energy$_{\texttt{N}}$ *yield assessment*
→ *estimation du rendement* **énergétique**$_{\texttt{ADJ}}$
(*assessment of* **energetical**$_{\texttt{ADJ}}$ *yield*)
Adding the entry *energy* → *énergétique*$_{\texttt{ADJ}}$ to the dictionary allows to cover morphological variation between source and target terms.

For the alignment, terms are lemmatized and need to be mapped to the respective inflected forms before being integrated into the SMT system. The translation probabilities are computed based on the relative frequencies of the inflected forms of all translation possibilities in the domain-specific data:

		MWT	SWT
tuning set	total	440	1014
test set	total	442	1015
tuning set	not in phrase-table	156	18
test set	not in phrase-table	192	15

Table 1: Number of terms (types) for which one or more translation candidates were found.

EN	FR	freq	prob
hub height	hauteur du moyeu	14	87.5
	hauteur de moyeu	2	12.5

Table 1 gives an overview of the number of obtained translation pairs for terms extracted from the test/tuning data (cf. section 7); we differentiate between single-word terms (SWTs) and multi-word terms (MWTs). This is motivated by the fact that MWTs provide more context in step (3) and are therefore more likely to be correctly aligned. In the case of SWTs, every translation listed in the dictionary can be output as a valid alignment provided it occurred in the corpus, regardless of context. Table 1 also shows the amount of term-translation pairs not covered by the phrase-table: in the case of MWTs, a reasonable amount of term-translation pairs are new to the system, whereas the number of new SWTs is very low in comparison to the number of found SWT term-translation pairs.

The pattern-based compositional term alignment tends to favor precision over recall. This general outcome is observed in earlier work (Weller and Heid, 2012)[3]; we assume that the findings for DE-EN largely also apply to our EN-FR alignment scenario. Moreover, it is not guaranteed that the translation of a source term occurs in the target-language data when working with comparable corpora. Another problem are structural mismatches of the source term and its target-language equivalent. While the translation occurs in the target language term list, it is of a different morpho-syntactic structure in a way that is not captured by the patterns and morphological rules. Finally, lack of dictionary coverage is also responsible for not finding target-language equivalents. We focus on integrating moderate amounts of good-quality term pairs, motivated by our method for integrating term pairs: our results indicate that the SMT-system is sensitive to incorrect translations, particularly for SWTs.

[1] *de*: French preposition meaning *of*.

[2] Working with translation patterns, non-content words such as prepositions can be easily inserted in this step.

[3] We use alignment patterns adapted from this work.

SMT-output + stem-markup	**pred.** **feat.**	**gen.** **forms**	**post-** **proc.**	**gloss**
le<+DET>[ART]	M.Sg	le	**l'**	*the*
excès<**M.Sg**>[N]	M.Sg	excès	excès	*excess*
de[P]	–	de	**d'**	*of*
énergie<**F.Sg**>[N]	F.Sg	énergie	énergie	*energy*
peut[VFIN-peut]	–	peut	peut	*can*
être[VINF]	–	être	être	*be*
vendre[VPP]	M.Sg	vendu	vendu	*sold*
à[P]	–	à	**au**	*to*
le<+DET>[ART]	M.Sg	le		*the*
réseau<**M.Sg**>[N]	M.Sg	réseau	réseau	*grid*

Table 2: Processing steps for the EN input sentence *[...] excess energy can be sold back to the grid.*

4 Inflection prediction system

To build the morphology-aware system, the target-side data (parallel and language model data) is transformed into a stemmed format, based on the annotation of a morphological tagger (Schmid and Laws, 2008). This representation contains translation-relevant feature markup: nouns are marked with *gender* (considered part of the stem) and *number*. Assuming that source-side nouns are translated by nouns with the same *number* value, this feature is indirectly determined by the source-side input. The number markup is thus needed to ensure that the source-side number information is transferred to the target side. For a better generalization, we split portmanteau prepositions into article and preposition (*au* → *à+le*: *to+the*).

For predicting the morphological features of the SMT output (*number* and *gender*), we use a linear chain CRF (Lavergne et al., 2010). In the prediction step, the values specified in the stem-markup (*number* and *gender* on nouns) are propagated over the rest of the phrase, as illustrated in column 2 of table 2. Based on the stems and the morphological features, inflected forms can be generated using a morphological tool for analyzing and generating inflected forms (cf. section 7), as illustrated in column 3. In order to generate correct French surface forms, a post-processing step is required, including the re-insertion of apostrophes and portmanteau merging (*à+le* → *au*), cf. column 4.

5 Integration of term-translation pairs

In this section, we compare two methods to integrate bilingual terminology, using a standard SMT-system (to be referred to as the "inflected" system): using XML-markup and in the form of pseudo-parallel data. In section 6, we discuss the integration of terms into the "morphology-aware" system.

Using XML input to add translation options. One way to integrate term-translation pairs into an SMT system is to list translation options with their translation probabilities for a word or word sequence in the input sentence by means of XML-markup. This approach has been applied by Hálek et al. (2011) (cf. section 2) to translations of named entities mined from Wikipedia in an English-Czech SMT system. In contrast, we integrate translation pairs of nominal phrases: this requires modelling features that are dependent on the source-side (e.g. number) which is not to the same extent necessary for names. Named entities are in many cases easier to deal with than terminology, as they are usually the same on the source side, even though their inflection can vary, e.g. in the form of *case*-markers, which depend on the target-language. This means that source-side information plays a negligible role, whereas for nominal phrases, number information (as contained in the stem markup) is important for the generation of inflected forms.

For the integration of term translation pairs, potential source terms are identified in the input sentence using the same pattern-based approach as for monolingual term identification (cf. section 3). Longer terms are preferred in the case of several annotation possibilities in order to provide the system with long translations, but also to avoid that phrasal units are interrupted: *[wind$_N$ energy$_N$] site$_N$* vs. *[wind$_N$ energy$_N$ site$_N$]*.

We compare the effects of integrating multi-word and single-word terms vs. only multi-word terms. As a variant, only term-translation pairs of which the source-side term does not occur in the phrase table are integrated: assuming that the translation model already has more reliable statistics for terms in the phrase-table, only term-translation pairs that are not covered by the parallel data are used. Particularly for SWTs, this drastically reduces the amount of term-translation pairs. When restricting the integration to "new" terms, however, the problem of polysemy (e.g. *boss* → *manager* or *rivet-type object*) is not resolved. In such cases, it is even likely that the wrong sense, i.e. the general language meaning, is output by the translation system. Nevertheless, this variant leads to the best results.

As term alignment is based on lemmas, a mapping between surface forms and lemmas is needed: first, inflected EN surface forms are projected to their lemmas, which are then aligned to FR lemmas. Then, the aligned target-side lemmas are mapped

Input	`clean the <term translation=''fer au rotor||pale de rotor||pales de rotor ||pale du rotor||pales du rotor'' prob=''0.0385||0.0385||0.2692||0.1153|| 0.5384''> rotor blades </term> with a mild soap and water .`
Baseline	`nettoyage du rotor des lames de savon avec une légère et de l' eau .` *cleaning of the rotor of the blades (of a knife) of soap with a mild and water.*
With terms	`nettoyer les pales du rotor avec un savon modérée et de l' eau .` *cleaning the blades of the rotor with a moderate soap and water.*
Reference	*Nettoyez les pales du rotor au savon doux et à l'eau.*

Table 3: Adding translation options for the term *rotor blades* to the input sentence.

to the respective inflected forms observed in the domain-specific corpus. As a result, some of the inflected forms can be incorrect in terms of *number* by mapping the lemma to both singular and plural forms, regardless of the input term. Filtering for number in this step is useful only to a limited extent, as it will prevent a translation entirely if the inflected forms of the required *number* value do not occur in the domain-specific data. While a good translation in the wrong number is clearly better than no translation, it is still desirable to have the possibility to model *number*: we consider this a strong motivation for a morphology-aware integration of terminology.

Another crucial point is the language model data which needs to contain the target-language terms offered to the translation model. As all target language terms are extracted from a domain-specific corpus, this data is used in the language model.

The example in table 3 illustrates how the system benefits from the translations for the term *rotor blades* in the input sentence: while FR *pale* (blades on a wind mill) occurs once in the parallel data, there is no alignment to EN *blade*. As a result, *blades* is translated as *lames* (blades on a knife). Providing the translation options leads to the correct translation of *blades* → *pales* in the context of the term *rotor blades*. In addition, the system with terminology information produces a well-formed French sentence in contrast to the meaningless output of the baseline system, because the correct translation allows for matching a plausible word sequence with the language model.

Adding terms to parallel data. In our experiments, adding translation options via XML markup did not work as well as hoped for; this is in line with the findings of Hálek et al. (2011): adding translation pairs directly into the SMT system can be too intrusive, causing more harm than benefit. We tested a different approach: the term-translation pairs are added as a pseudo parallel corpus to the parallel training data. Adding each term-translation pair once is not likely to help if the word is ambiguous and already occurs in the parallel data with its general language translation. Instead, term translation pairs are added according to their frequency in the target-side corpus. As before, all observed inflected forms are listed as possible translations.

6 Morphology-aware integration of term-translation pairs

The setup described in the previous sections has two shortcomings: the data might not provide the full inflection paradigm of the terms, and it is not possible to model features such as *number*: integrating stemmed terms to the inflection prediction system allows us to handle these two problems as the number information of a source-term can simply be transferred as number markup to the stemmed translation candidate and specific forms not occurring in the data used for term mining can be generated using a morphological resource.

For the terminology integration into a morphology-aware translation system, we opted for the variant of adding pseudo parallel data to the training data of the SMT system as this led to the best results in the previous experiments. First, the aligned terms are transferred to the stemmed representation. For the number markup, the source-side is tagged and the *number* values are transferred to the corresponding stems based on the alignment patterns (cf. section 3). In this step, the number markup in the generated target-side text is determined by transfer from the source-side. In comparison, the number markup in the "original" parallel data (Europarl) is given by the target-side, i.e. the parse-annotation.

Generating target phrases depending on the requirements of the source-side, i.e. creating unseen forms, can lead to stem+markup combinations that do not occur in the data used to build the language model. Words not contained in the language model score very badly during decoding and are thus effectively not available to the SMT system. In order to make all stems accessible, the generated pseudo parallel data is added to the language model data.

An alternative way to avoid the generation of forms not represented in the language model consists in foregoing number markup. Instead of keeping it through the translation in form of stem markup, number information can be reinstated in the feature prediction step using source-side features. However, this creates two new problems: first, the representation without number markup loses discriminatory power[4]. For example, there is no way to guarantee subject-verb agreement without number information on nouns. The second problem is that parallel domain-specific data is needed to train the models for feature prediction. While we believe that removing number markup in the translation step is a sounder way to deal with target-side morphology in this application, we leave this extension of our model to future work due to the practical problems that arise with this.

7 Data and resources

Our experiments are carried out on an EN-FR standard phrase-based Moses[5] system which is adapted to the domain of wind energy. As a basis for terminology mining, we compiled a target-language corpus for that domain. This included documents obtained by automatic crawling (de Groc, 2011), and manually obtained data from various web-sites. In total, the corpus consists of 161.367 sentences (4.136.751 words). For the tuning/test data, we manually collected and sentence-aligned parallel texts from various internet resources, including manuals for setting up/maintaining wind energy towers, multi-lingual scientific journal articles and data about regulations and administrative aspects. The resulting 1290 parallel sentences were evenly divided into test/tuning sets.

The parallel training data for the EN-FR SMT system consists of 2.159.501 sentences (Europarl and News data from the 2013 WMT shared task). For the language model, we used a combination of the FR part of the parallel data and the wind energy corpus. As the domain-specific corpus is considerably smaller, we built individual language models for each corpus and interpolated them using weights optimized on the tuning data following the approach of Schwenk and Koehn (2008).

For the feature prediction, we used the Wapiti toolkit (Lavergne et al., 2010) to train CRFs on combinations of the wind corpus and the FR part of the parallel data. The CRF has access to the basic features *stem* and *POS tag* as well as *gender* and *number* within a window of 5 positions to each side of the current word.

The morphological analysis of the French training data is obtained using RFTagger, which is designed for annotating fine-grained morphological tags (Schmid and Laws, 2008). For generating inflected forms based on stems and morphological features, we use an extended version of the finite-state morphology FRMOR (Zhou, 2007). FRMOR is a morphology tool similar to SMOR (Schmid et al., 2004), which allows to analyze and generate inflected word forms. The term alignment requires a general language dictionary[6] from which we use the 36,963 1-to-1 entries.

8 Experiments and results

We present results for the integration of bilingual terminology into an inflected system and a morphology-aware translation system.

Integrating terminology into the inflected system. An easy way to adapt an SMT system to a new domain consists in adding language model data of that domain. This does not help with the problem of out-of-vocabulary words, but it can enhance translations with low probabilities and provide plausible contexts for the generated sentences. The systems in row 1 in table 4 show that adding domain-specific data leads to a considerable increase in BLEU; all further systems in table 4 use this enlarged language model and are compared to baseline *b*.

Moses' XML mode offers two possibilities: forcing the SMT system to use the given translations (*exclusive*) or allowing for an optional usage (*inclusive*). As preliminary experiments, as well as the findings of Hálek et al. (2011), showed that the inclusive setting leads to better results, we only report BLEU scores for this variant[7]. We compare two versions: providing only the translations of multi-word terms (MWTs) and providing the translations

[4] See also experiments on re-inflecting surface forms ("Method 1") in Toutanova et al. (2008).

[5] `http://www.statmt.org/moses`

[6] from `www.dict.cc` and `www.freelang.net`

[7] Particularly for SWTs, forcing the system to use the provided translations using the exclusive setting can very much hurt performance as it goes against Moses' tendency to use long translation units.

	system	BLEU
1	Baseline a: general LM	18.93
	Baseline b: +domain-spec. LM	21.59
2	XML-markup (MWT + SWT)	20.56
	XML-markup (MWT)	20.71
3	XML-markup-filt. (MWT + SWT)	21.68
	XML-markup-filt. (MWT)	21.57
4	Added parallel (MWT + SWT)	21.68
	Added parallel (MWT)	21.87
	Added parallel (MWT + filt. SWT)	22.03*
	Added parallel filt. (MWT + SWT)	21.96*

Table 4: Results for integration of terminology into an inflected EN-FR translation system. (*: significantly better than baseline b at a 0.05 level)

of both multi-word and single-word terms (SWTs). This is motivated by the assumption that adding translations of single words is likely to be more harmful as it is to some extent incompatible with Moses' tendency to prefer longer phrases.

The translation probabilities of term-translation pairs given in the XML markup usually are considerably higher than the ones in the phrase-table and might thus have an undue advantage, particularly when assuming that the statistics in the phrase-table are more reliable for terms that are not restricted to the domain. Furthermore, the generated translations of multi-word terms are more likely to be correct as they provide more context in the alignment step. While the system with only MWTs is slightly better, both variants are worse than the baseline (row 2 in table 4). Restricting the added term-translation pairs to those where the source-phrase does not occur in the phrase-table helps, but does not outperform the baseline (row 3 in table 4). Here, using both MWTs and SWT leads to a slightly better score, presumably because the added SWTs are unknown to the system and even a translation by a one-word phrase is beneficial.

Integrating bilingual terminology in the form of pseudo-parallel data leads to the best results (row 4 in table 4). Again, restricting the data to MWTs is slightly better than using all term-translation pairs. The score for the MWT-only system (21.87) is on the verge of being statistically significantly better than baseline *b*. Adding single-word translations which do not occur in the phrase-table leads to a statistically significant improvement (22.03), as does filtering both SWTs and MWTs (21.96).

Integrating terminology into the morphology-aware system. The score of the morphology-aware system (21.54) is comparable to that of the inflected system (21.59), as shown in table 5. The importance of in-domain training data for the CRF is illustrated by the results obtained when training the CRF on wind+news (318.112 sentences) and on wind+europarl (2.161.367 sentences): even though the second training set is considerably larger, there is basically no gain in BLEU. Considering this outcome, we assume that more in-domain training data for the CRF would lead to better overall results.

	system	CRF trained on	BLEU
1	Baseline	wind+news	21.47
		wind+europarl	21.54
2	MWT^a	wind+europarl	21.77
3	MWT + SWT^c	wind+europarl	21.11
4	MWT + filt. SWT^b	wind+europarl	21.74
5	filt. (MWT + SWT)^b	wind+europarl	21.48

Table 5: Adding pseudo parallel data to the training data for a morphology-aware system. a: LM from baseline system; b: MWT translations added to LM data; c: MWT+SWT translations added to LM data.

In order to make better use of the in-domain training data, singletons were replaced by their part-of-speech tag[8]. However, the stem feature considerably contributes to the prediction result: this is illustrated by the results in table 5, where a CRF trained on a combination of Europarl and wind energy data is only marginally better in terms of BLEU than a system trained on a much smaller amount of general language data and data of the wind energy domain.

It is important to keep in mind that the CRF is trained on fluent data whereas the SMT output is heavily disfluent. As a result, there is a mismatch between ill-formed translation output and the well-formed data used to train the CRF; the gap between training data and the text for which features are to be predicted gets larger with increasing difficulty of the translation task, as is the case here.

Effects caused by sparse data do also affect the language model data: forms which are not contained in the parallel data cannot be produced by the translation system. In order to deal with out-of-vocabulary words, stem markup+tags are stripped of all those words in the language model data that do not occur in the parallel data. This enables the SMT system to score unknown words (e.g. names) in the language model, but also leads to side-effects due to sparsity: for example, the French term *rotors* occurs once in the parallel data and is correctly stemmed as `rotor<Masc.Pl>[N]` , while all occurrences of *rotor* in the singular form

[8] Experiments with replacing out-of-vocabulary words by a special tag were also not effective in terms of BLEU.

are stripped of the markup and treated as a name and thus do not undergo the inflection process.

As the method of adding term translation pairs to the parallel data led to the best results for the inflected system, we opted for this method for the integration of terms into the morphology-aware system. While the MWT-only system (2 in table 5) gets a better score than the baseline (1 in table 5) (21.77 vs. 21.54 using the larger CRF), the difference is not statistically significant. In contrast to the results for the inflected system, adding the set of SWTs filtered against the phrase-table slightly decreases BLEU, whereas adding all SWTs leads to a considerable decrease in BLEU. We assume that this outcome is partially caused by a problem with the language model: while all generated target terms are added to the language model data, they are not embedded in the context of a sentence, or, if also adding SWTs (system 3 in table 5), not even in the context of a term.

9 Conclusion

We presented different approaches to integrate bilingual terminology of a technical domain into an SMT system. First, we compared two integrating methods (providing translation options vs. term-translation pairs as pseudo-parallel data) and studied the effects of using only multi-word terms in comparison to both single-word and multi-word terms. Then, we applied the best term integration strategy to a morphology-aware translation system.

With the inflected system, we obtained a significant improvement over the baseline when adding terms as pseudo-parallel data. Our evaluation also clearly showed that Moses' XML mode has considerable problems in dealing with single-word terms. Furthermore, we highlighted the need for explicit modeling of morphological features for the integration of bilingual terminology.

While the morphology-aware system enriched with term pairs was not able to outperform the baseline on a statistically significant level, it outlines a pipeline that tackles two central problems of adapting translation systems to under-resourced domains: (i) preservation of translation-relevant features and (ii) generation of previously unseen inflected forms.

10 Acknowledgements

This work was funded by the DFG Research Projects "Distributional Approaches to Semantic Relatedness" and "Models of Morpho-Syntax for Statistical Machine Translation".
The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under Grant Agreement n. 248005.

References

Daille, B. and E. Morin. 2005. French-English terminology extraction from comparable corpora. In *Proceedings of IJCNLP 2005*.

Daumé III, H. and J. Jagarlamudi. 2011. Domain adaptation for machine translation by mining unseen words. In *Proceedings of ACL 2011*.

de Groc, C. 2011. Babouk: Focused web crawling for corpus compilation and automatic terminology extraction. In *International Conferences on Web Intelligence and Intelligent Agent Technology*.

Formiga, L., A. Hernández, J. Mariño, and E. Monte. 2012. Improving English to Spanish out-of-domain translations by morphology generalization and generation. In *Proceedings of AMTA 2012*.

Fraser, A., M. Weller, A. Cahill, and F. Cap. 2012. Modeling Inflection and Word-Formation in SMT. In *Proceedings of EACL 2012*.

Haghighi, A., P. Liang, T. Berg-Kirkpatrick, and D. Klein. 2008. Learning bilingual lexicons from monolingual corpora. In *Proceedings of ACL 2008*.

Hálek, O., R. Rosa, A. Tamchyna, and O. Bojar. 2011. Named entities from wikipedia for machine translation. In *Proceedings of the Conference on Theory and Practice of Information Technologies*.

Lavergne, T., O. Cappé, and F. Yvon. 2010. Practical very large scale CRFs. In *Proceedings of ACL 2010*.

Pinnis, M. and R. Skadins. 2012. MT adaptation for under-resourced domains - what works and what not. In *Proceedings of HLT - the baltic Perspective*.

Prochasson, E. and P. Fung. 2011. Rare word translation extraction from aligned comparable documents. In *Proceedings of ACL 2011*.

Schmid, H. and F. Laws. 2008. Estimation of conditional probabilities with decision trees and an application to fine-grained pos tagging. In *Proceedings of COLING 2008*.

Schmid, H., A. Fitschen, and U. Heid. 2004. SMOR: a German Computational Morphology Covering Derivation, Composition, and Inflection. In *Proceedings of LREC 2004*.

Schwenk, H. and P. Koehn. 2008. Large and diverse language models for statistical machine translation. In *Proceedings of IJCNLP 2008*.

Toutanova, K., H. Suzuki, and A. Ruopp. 2008. Applying Morphology Generation Models to Machine Translation. In *Proceedings of ACL-HLT 2008*.

Weller, M. and U. Heid. 2012. Analyzing and aligning german compound nouns. In *Proceedings of LREC 2012*.

Wu, Hua, Haifeng Wang, and Chengqing Zong. 2008. Domain adaptation for statistical machine translation with domain dictionary and monolingual corpora. In *Proceedings of COLING 2008*.

Zhou, Z. 2007. Entwicklung einer französischen Finite-State-Morphologie. University of Stuttgart.

An efficient method to assist non-expert users in extending dictionaries by assigning stems and inflectional paradigms to unknown words

Miquel Esplà-Gomis mespla@dlsi.ua.es
Víctor M. Sánchez-Cartagena vmsanchez@dlsi.ua.es
Felipe Sánchez-Martínez fsanchez@dlsi.ua.es
Rafael C. Carrasco carrasco@dlsi.ua.es
Mikel L. Forcada mlf@dlsi.ua.es
Juan Antonio Pérez-Ortiz japerez@dlsi.ua.es

Dept. de Llenguatges i Sistemes Informàtics, Universitat d'Alacant, 03071 Alacant, Spain

Abstract

A method is presented to assist users with no background in linguistics in adding the unknown words in a text to monolingual dictionaries such as those used in rule-based machine translation systems. Adding a word to these dictionaries requires identifying its stem and the inflection paradigm to be used in order to generate all its word forms. Our method is based on a previous interactive approach in which non-expert users were asked to validate whether some tentative word forms were correct forms of the new word; these validations were then used to determine the most appropriate stem and paradigm. The previous approach was based on a set of intuitive heuristics designed both to obtain an estimate of the eligibility of each candidate stem/paradigm combination and to determine the word form to be validated at each step. Our new approach however uses formal models for both tasks (a hidden Markov model to estimate eligibility and a decision tree to select the word form) and achieves significantly better results.

1 Introduction

Creation of the linguistic data (such as monolingual dictionaries, bilingual dictionaries, transfer rules, etc.) required by rule-based machine translation (RBMT) systems has usually involved teams of trained linguists. However, development costs could be significantly reduced by involving a broader group of non-expert users in the extension of these resources. This may include, for instance, the very same users of the machine translation (MT) system or accidental collaborators recruited through crowdsourcing platforms (Wang et al., 2013). The scenario considered in this paper is that of non-expert *users* (in a general sense) who have to introduce into the two monolingual dictionaries[1] of a RBMT system the unknown words found in an input text so that the system is subsequently able to correctly translate them.[2] Note, however, that our method could be applied to the addition of entries into the morphological dictionaries used in many other natural language processing applications. The objective of our work is to obtain a system which can be used not only to add the particular unknown word form (for example, *wants*) to the dictionary, but also to assist in discovering an appropriate *stem* and a suitable *inflection paradigm* so that all the word forms of the unknown word and their associated morphological inflection information (such as *wants, verb, present, 3rd person* or *wanting, verb, gerund*) can be inserted as well.

Inflection paradigms are commonly introduced in RBMT systems in order to group regularities in the inflection of a set of words;[3] a paradigm is usually defined as a collection of suffixes and their corresponding morphological information; e.g., the paradigm assigned to many common English verbs indicates that by adding the suffix *-ing* to the stem,[4]

[1] One source-language dictionary used for morphological analysis and one target-language dictionary used for morphological generation.

[2] It could also happen that the word form is not completely unknown, but it is assigned to a different paradigm; for example, the word *fly* could already be included in a dictionary as a verb, but a user may need to insert it as a noun.

[3] Paradigms ease the management of dictionaries in two ways: by reducing the quantity of information that needs to be stored, and by simplifying revision and validation because of the explicit encoding of regularities in the dictionary.

[4] The stem is the part of a word that is common to all its

the gerund is obtained; by adding the suffix *-ed*, the past is obtained; etc. Adding a new entry to a monolingual dictionary therefore implies determining the stem of the new word and a suitable inflection paradigm among those defined by the MT system for the corresponding language. In this work we assume that the paradigms for all possible words in the language are already included in the dictionary.[5] We will focus on monolingual dictionaries because insertion of information in the bilingual dictionaries of RBMT systems is usually straightforward (Sánchez-Cartagena et al., 2012a).

Our approach improves a previous interactive method (Esplà-Gomis et al., 2011) that was based on a number of intuitive heuristics; the improvement presented here is twofold: on the one hand, more coherent and principled models are introduced; on the other hand, the results are significantly better.

The rest of the paper is organised as follows. Section 2 discusses other works related to our proposal. Section 3 introduces the concepts on monolingual dictionaries that will be used in the remainder of the paper. An overview of the previous method (Esplà-Gomis et al., 2011) for dictionary extension is presented in Section 4, followed by the description of our new approach in Section 5. Section 6 discusses our experimental setting in which a Spanish monolingual dictionary is used, while the results obtained are presented and discussed in Section 7. Finally, some concluding remarks are presented in Section 8.

2 Related work

In this section, related works in literature are commented and compared with the common features in our new approach and in the work by Esplà-Gomis et al. (2011).

Two of the most prominent works in literature in relation to the elicitation of knowledge to build or improve RBMT systems are those by Font-Llitjós (2007) and McShane et al. (2002). The former proposes a strategy for improving both transfer rules and dictionaries by analysing the postediting process performed by a non-expert user through a dedicated interface. McShane et al. (2002) design a framework to elicit linguistic knowledge from informants who are not trained linguists and use this information in order to build MT systems which translate into English; their system provides users with a lot of information about different linguistic phenomena to ease the elicitation task. Unlike these two approaches, our method is aimed at transfer-based MT systems in which a single translation is generated and no language model is used in order to rank a number of translation hypothesis; this kind of systems are notably more sensitive to erroneous linguistic information. We also want to relieve users from acquiring linguistic skills.

Additional tools that ease the creation of linguistic resources for MT by users with some linguistic background have also been developed. To this end, the *smart paradigms* devised by Détrez and Ranta (2012) help users to obtain the right inflection paradigm for a new word to be inserted in an MT system dictionary. A smart paradigm is a function that returns the most appropriate paradigm for a word given its lexical category, some of its word forms and, in some cases, some morphological inflection. There are two important differences with our approach: firstly, smart paradigms are created exclusively by human experts; and secondly, users of smart paradigms need to have some linguistic background. For instance, an expert could decide that in order to correctly choose the inflection paradigm of most verbs in French the infinitive and the first person plural present indicative forms are needed; dictionary developers must then provide these two forms when inserting a new verb. Bartusková and Sedlácek (2002) also present a tool for semi-automatic assignment of words to declension patterns; their system is based on a decision tree with a question in every node. Their proposal, unlike ours, works only for nouns and is aimed at experts because of the technical nature of the questions. Desai et al. (2012) focus on verbs and present a system for paradigm assignment based on the information collected from a corpus for each compatible paradigm; if the automatic method fails, users are then required to manually enter the correct paradigm.

As regards the automatic acquisition of morphological resources for MT, the work by Šnajder (2013) is of particular interest: he turns the choice of the most appropriate paradigm for a given word into a machine learning problem. Given the values of a set of features extracted from a monolingual corpus and from the orthographic properties of the lemmas, each compatible paradigm is classified as correct/incorrect by a *support vector machine* classifier. The main difference with our approach

inflected forms.

[5] This can be easily expected as most unknown words belong to regular paradigms.

lies in the fact that their method is designed to be used in a fully-automatic pipeline, while we use the inferred models in order to minimise the number of queries posed to non-expert users. Finally, the automatic identification of morphological rules to segment a word into morphemes (a problem for which paradigm identification is a potential resolution strategy) has also been recently addressed (Monson, 2009; Walther and Nicolas, 2011).

3 Preliminaries

Let $P = \{p_i\}$ be the set of paradigms in a monolingual dictionary. Each paradigm p_i defines a set of pairs (f_{ij}, m_{ij}), where f_{ij} is a suffix[6] which is appended to stems to build new *word forms*, and m_{ij} is the corresponding morphological information. Given a *stem/paradigm* pair $c = t/p_i$ composed of a stem t and a paradigm p_i, the *expansion* $I(c)$ is the set of possible word forms resulting from appending each of the suffixes in p_i to t. For instance, an English dictionary may contain the stem *want*-assigned to a paradigm with suffixes[7] p_i = {-, -*s*, -*ed*, -*ing*}; the expansion $I(\text{want}/p_i)$ consists of the set of word forms *want*, *wants*, *wanted* and *wanting*.

Given a new word form w to be added to a monolingual dictionary, our objective is to find both the stem $t \in \text{Pr}(w)$[8] and the paradigm p_i such that $I(\text{want}/p_i)$ is the set of word forms which are all the correct forms of the unknown word. To that end, a set L containing all the stem/paradigm pairs compatible with w is determined by using a *generalised suffix tree* (McCreight, 1976) containing all the possible suffixes included in the paradigms in P.

The following example illustrates the previous definitions. Consider a simple dictionary with only four paradigms: p_1 = {-, -*s*}; p_2 = {-*y*, -*ies*}; p_3 = {-*y*, -*ies*, -*ied*, -*ying*}; and p_4 = {-*a*, -*um*}. Let's assume that the new word form is w=*policies* (actually, the noun *policy*); the compatible stem/paradigm pairs which will be obtained after this stage are: c_1=*policies*/p_1; c_2=*policie*/p_1; c_3=*polic*/p_2; and c_4=*polic*/p_3.

4 Previous approach

Esplà-Gomis et al. (2011) have already proposed an interactive method for extending the dictionaries of RBMT systems with the collaboration of non-expert users. In their work, the most appropriate stem/paradigm pair is chosen by means of a sequence of simple *yes/no questions* whose answer only requires *speaker-level* understanding of the language. Basically, users are asked to validate whether some word forms resulting from tentatively assigning different compatible stem/paradigm pairs in L (see Section 3) to the new word are correct word forms of it. The specific forms that are presented to the users for validation are automatically obtained by estimating the most informative ones which allow the system to discard the greatest number of wrong candidate paradigms at each step. The results showed (Esplà-Gomis et al., 2011) that the average number of queries posed to the users for a Spanish monolingual dictionary was around 5, which is reasonably small considering that the average number of initially compatible paradigms was around 56. Furthermore, Sánchez-Cartagena et al. (2012a) have shown that when the source-language word has already been inserted, the system is able to more accurately predict the right target-language paradigm by exploiting the correlations between paradigms in both languages from the corresponding bilingual dictionary, thus reducing significantly the number of questions.

After obtaining the list of compatible stem/paradigm pairs L, the original method performs three tasks: stem/paradigm pair scoring, selection of word forms to be offered to the user for validation an discrimination between equivalent paradigms.

Paradigm scoring. A *feasibility score* is computed for each compatible stem/paradigm pair $c_n \in L$ using a large monolingual corpus C. Candidates producing a set of word forms which occur more frequently in the corpus get higher scores. Following our example, the word forms for the different candidates would be: $I(c_1)$={*policies*, *policiess*}; $I(c_2)$={*policie*, *policies*}; $I(c_3)$={*policy*, *policies*}; and $I(c_4)$={*policy*, *policies*, *policied*, *policying*}. Using a large English corpus, word forms *policies* and *policy* will be easily found, and the rest of them (*policie*, *policiess*, *policied* and *policying*) probably will not. Therefore, c_3 would probably obtain the highest feasibility score.

Selection of word forms. The best candidate is chosen from L by querying the user about a reduced

[6] Although our approach focuses on languages generating word forms by adding suffixes to stems (for example, Romance languages), it could be easily adapted to inflectional languages based on different ways of adding morphemes.

[7] We hereinafter omit the morphological information contained in p_i and show only the suffixes.

[8] $\text{Pr}(w)$ is the set of all possible prefixes of w.

set of the word forms for some of the compatible stem/paradigm pairs $c_n \in L$. To do so, the system first sorts L in descending order using the feasibility score. Then, users are asked (following the order in L) to confirm whether some of the word forms in each compatible stem/paradigm pair are correct forms of w. In this way, when a word form w' is accepted by the user, all $c_n \in L$ for which $w' \notin I(c_n)$ are removed from L; otherwise, all $c_n \in L$ for which $w' \in I(c_n)$ are removed from L. In order to attempt to maximise the number of word forms discarded and consequently minimise the amount of yes/no questions, users are iteratively asked to validate the word form from the first compatible stem/paradigm pair in L which exists in the minimum number of other compatible stem/paradigm pairs. This process is repeated until only one candidate (or a set of equivalent candidates; see next) remains in L.

Equivalent paradigms. When more than one paradigm provides exactly the same set of suffixes but with different morphological information, no additional question can be asked in order to discriminate between them.[9] For example, in the case of Spanish, many adjectives such as *alto* ('high') and nouns such as *gato* ('cat') are inflected identically. Therefore, two paradigms producing the same collection of suffixes {*-o* (masculine, singular), *-a* (feminine, singular), *-os* (masculine, plural), *-as* (feminine, plural)} but with different morphological information are defined in the monolingual dictionary, the stems *alt-* and *gat-* assigned to one of them each. This issue also affects paradigms with the same lexical category: *abeja* and *abismo* are nouns that are inflected identically; *abeja* is however feminine, whereas *abismo* is masculine. When adding unknown words such as *gato* or *abeja*, no yes/no question can consequently be asked in order to discriminate between both paradigms. Sánchez-Cartagena et al. (2012b) proposed a solution to this issue that consisted of introducing an n-gram-based model of lexical categories and inflection information which was used as a final step[10] to automatically choose the right stem/paradigm pair with success rates between 56% and 96%.

5 Method

The approach discussed in the preceding section provides a complete framework for dictionary extension, but this framework can still be improved if more rigorous and principled models rather than intuitive heuristics are used. We propose consequently to replace those heuristics with hidden Markov models (HMMs) (Rabiner, 1989) and binary decision trees as follows. For a given unknown word form, first the set L of compatible stem/paradigm pairs is determined (see Section 3). The probability of each of them is then estimated by means of a first-order HMM. After that, these probabilities are used in order to build a decision tree which is used to guide the selection of words to be offered to the non-expert user for validation. Note that, unlike in the original method in which isolated unknown words were inserted into the dictionary, the HMM in our new method explicitly considers the sentence in which the new word appears and uses this contextual information in order to better estimate the likelihood of each compatible stem/paradigm pair. The objective here is to minimise the interaction with the user so that the addition of new words is made as fast as possible.

Hidden Markov models. A first-order HMM is defined as $\lambda = (\Gamma, \Sigma, A, B, \pi)$, where Γ is the set of states, Σ is the set of observable outputs, A is the $|\Gamma| \times |\Gamma|$ matrix of state-to-state transition probabilities, B is the $|\Gamma| \times |\Sigma|$ matrix with the probability of each observable output $\sigma \in \Sigma$ being emitted from each state $\gamma \in \Gamma$, and the vector π, with dimensionality $|\Gamma|$, defines the initial probability of each state. The system produces an output each time a state is reached after a transition. In our method, Γ is made up of all the paradigms in the dictionary and Σ corresponds to the set of suffixes produced by all these paradigms.

Our HMMs are trained in a way very similar to HMMs used in unsupervised part-of-speech tagging (Cutting et al., 1992), that is, by using the Baum-Welch algorithm (Baum, 1972) with an untagged corpus. The training corpus is built from a text corpus as follows: (i) the monolingual dictionary is used in order to obtain the set F of all possible word forms; (ii) the word forms in the text corpus that belong to F are assigned all their corresponding suffix and paradigm pairs; (iii) the word forms not in F are assigned the set of suffix and paradigm pairs obtained from the set L of their compatible candidates, as described in Section 4. Once the HMM is trained, the probability $q_t(c_n)$

[9] Around 81% of the word forms in a Spanish dictionary have been reported (Sánchez-Cartagena et al., 2012b) to be assignable to more than one equivalent paradigm.

[10] Note that this model is disconnected from the models used for scoring the compatible paradigms and deciding the word forms to be shown to the user.

of assigning the word form located at position t in the sentence to the compatible candidate $c_n \in L$ can be computed by applying the following equation, which corresponds to Eq. (27) in the tutorial by Rabiner (1989):

$$q_t(c_n) = \frac{\alpha_t(c_n)\beta_t(c_n)}{\sum_{m=1}^{|L|} \alpha_t(c_m)\beta_t(c_m)} \qquad (1)$$

This equation computes the probability that the model is in state c_n when at position t. In the equation, $\alpha_t(c_n)$ accounts for the (forward) probability of the sub-sentence from the begining of the sentence to position t given state c_n at position t, whereas $\beta_t(c_n)$ corresponds to the (backward) probability of the sub-sentence from position $t+1$ to the end of the sentence, given state c_n at position t (Rabiner, 1989).

Decision trees. Decision trees are commonly used to learn classifiers: the internal nodes (decision nodes) of a decision tree are labelled with an input feature, an arc coming from an internal node exists for each possible feature value, and leaves are labelled with classes. The ID3 algorithm (Quinlan, 1986) has been proposed in order to build these trees. This algorithm follows a greedy approach (the resulting trees are therefore sub-optimal) by selecting the most appropriate attribute to split the data set on each iteration. The algorithm starts from the root of the tree with the whole data set S. At each iteration, an attribute A is picked for splitting S, being A the attribute providing the highest information gain. A child node is then created for each possible value of A, with a new test set containing only the elements matching this attribute value. The information gain measures the difference in entropy before and after S is split; for computing this entropy, the probability of each class is approximated by using the *proportion* of elements belonging to each of them.

Our method uses ID3 in order to build a binary (each node corresponds to a yes/no question) decision tree for each new word. Each class corresponds to a compatible stem/paradigm and the attribute set is made up of the set of different word forms, i.e. $\cup_{c_i \in L} I(c_i)$. The entropy in the ID3 algorithm could in principle be computed as stated before, i.e. by using the proportion of word forms derived from every stem/paradigm combination. In our approach, however, a more accurate computation of the entropy is proposed by using the class probability provided by the hidden Markov model.

A weakness that this method shares with the one described in Section 4 is that candidate paradigms producing the same collection of suffixes cannot be differentiated with yes/no questions. Therefore, at the end of the querying process, it is possible for more than one candidate to remain. In order to deal with this, the already computed HMM contextual probabilities could be used rather than the additional n-gram model of morphological information proposed by Sánchez-Cartagena et al. (2012b).[11] For this work, as in the one by (Sánchez-Cartagena et al., 2012a), we considered these paradigms producing the same word forms as equivalent and, therefore, they count as a single paradigm.

6 Experimental Setting

In order to ensure an accurate comparison between the methods described in Sections 4 and 5, our experimental framework replaces non-expert users, to which this method is eventually addressed, with an oracle so that interferences caused by human errors are avoided. The evaluation consisted of simulating the addition of a set of words to the Spanish monolingual dictionary of the Spanish–Catalan Apertium MT system (Forcada et al., 2011).

Six test sets were built consisting of sentences in Spanish containing at least an unknown word. Using an oracle, the average number of questions needed in order to obtain the correct paradigm was computed for the following three methods: the original approach by Esplà-Gomis et al. (2011) described in Section 4, a decision tree using proportions rather than probabilities,[12] and a decision tree assigning the probabilities estimated by an HMM. It is worth noting that this metric ignores the fact that, depending on the word form posed, a user could need more time to decide whether to accept or reject it. This will be evaluated in a future work. In addition to the average number of questions, the HMM probabilities and the feasibility scores of the original approach were compared by evaluating the success in detecting the correct paradigm, that is, in assigning the highest score or probability to the correct paradigm. This second metric is aimed at measuring the relation between the relative correctness in the probability/score assignment and the number of queries posed to the user.

Each of the six data sets consists of (i) a mono-

[11] Although out of the scope of this work, it could be interesting to compare both approaches to the task of choosing (or supporting a user to choose) the best correct compatible stem/paradigm combination.

[12] As in this approach there is only one element per class, this is equivalent to consider all classes as equiprobable.

lingual dictionary D; (ii) a collection of text sentences S containing each at least one word form of a word not in D; and (iii) the list of the correct stem/paradigm combination for the target word forms to be added to the dictionary, which is used as the oracle for our evaluation.

In order to measure the feasibility of these methods at different times in the development of a dictionary, the revision history of the dictionary in the Apertium project Subversion repository was used.[13] This strategy also allowed us to use the different revisions in order to build the oracle for the experiments: given a pair of dictionary revisions (R_1, R_2) with R_1 being an earlier revision than R_2, the evaluation task consisted of adding to R_1 the words in R_2 but not in R_1 (i.e., the relative complement of R_1 in R_2), which will be called, henceforth, target words. In order to ensure that all the paradigms assigned to these words were also available in R_1, we sequentially checked all the revisions of the dictionary and grouped them according to their paradigm definitions, thus obtaining ranges of *compatible revisions*. We then computed the number of words differing between the oldest and newest revisions of each range, and manually picked for the experiments six revision pairs among those with the greatest number of different words.

In order to obtain sentences containing the target words, the Spanish side of the parallel corpus News Commentary (Bojar et al., 2013) was used.[14] The corpus was split into two parts, one containing 90% of the sentences, which were used for training the HMM, and another one including the remaining 10%, which were used for testing. Sentences not containing at least one word form of one of the target words were removed from each test set. Table 1 shows the list of revision pairs, the number of words differing between them and the number of word forms included in the evaluation text. For both the training and testing corpora, the text was processed by following the strategy described in Section 5 using the revision R_1 of each test set. A different HMM was therefore trained for each test set; in all cases, the Baum–Welch algorithm was stopped after 9 iterations.

Finally, following the experimental setting proposed by Sánchez-Cartagena et al. (2012a), a word list obtained from the Spanish Wikipedia dump[15] was used as the monolingual corpus to compute the the feasibility scores in the heuristic-based approach in Section 4.

Revision pair		**Target**	**Target word**
R_1	R_2	**words**	**forms in corpus**
7217	7287	109	485
11762	12415	1802	550
17582	20212	700	362
27241	27627	1048	297
34649	35985	1194	79
36838	44118	1039	650

Table 1: Revision pairs of the Spanish monolingual dictionary in the Apertium Spanish–Catalan MT system used in the experiments, number of target words (added from R_1 to R_2), and number of target word forms appearing in the corpus.

7 Results and Discussion

Table 2 shows the average number of questions needed to determine the correct paradigm for the target words evaluated. Since the objective of our method is to reduce the interaction with the user as much as possible, lower values represent better results. Cells in bold correspond to statistically significant differences between the corresponding method and the two other approaches with $p \leq 0.05$.[16] Those values which are significantly better are marked with the symbol $\uparrow$, whereas values significantly worse are marked with $\downarrow$. As can be seen, the two decision-tree-based approaches are, in general, better than the heuristic-based approach. Contrary evidence however is seen for the sole case of the test set corresponding to revision pair $(7217, 7287)$. Furthermore, using the HMM probabilities for computing information gain in the ID3 algorithm results in a statistically significant improvement to the original ID3 method in four out of the six test sets evaluated. In order to shed some light on these results, additional experiments were performed in order to check how well the feasibility scores and the HMM-based probabilistic model ranked the candidate paradigms. Table 3 shows the average position of the correct paradigm in the sorted candidate list, as well as the percentage of times that the correct paradigm was ranked as the first one. Overall, the results in this table suggest that the quality of the ranking has a higher impact

[13] `https://svn.code.sf.net/p/apertium/svn/trunk/apertium-en-es/apertium-en-es.es.dix`

[14] This corpus was chosen because it belongs to an heterogeneous domain and it is already segmented into sentences.

[15] `http://dumps.wikimedia.org/eswiki/20110114/eswiki-20110114-pages-articles.xml.bz2`

[16] Statistical significance tests were performed with *sigf*, available at `http://www.nlpado.de/~sebastian/software/sigf.shtml`

Revision pairs		mean number of queries		
R_1	R_2	ID3+HMM	ID3	Original
7217	7287	3.26	**5.50**$^{\downarrow}$	**3.08**$^{\uparrow}$
11762	12415	5.22	5.26	**10.71**$^{\downarrow}$
17582	20212	**4.74**$^{\uparrow}$	**5.65**$^{\downarrow}$	5.18
27241	27627	**4.35**$^{\uparrow}$	5.72	5.85
34649	35985	6.22	6.32	**8.67**$^{\downarrow}$
36838	44118	**5.83**$^{\uparrow}$	6.11	**7.48**$^{\downarrow}$

Table 2: Mean number of yes/no questions needed by the tree approaches under evaluation (ID3-trained decision tree using HMM probabilities, ID3-trained decision tree using proportions, and heuristic-based approach) for each of the test sets.

in the heuristic-based original approach: in the case of the revisions pair $(7217, 7287)$, the good results in ranking end up producing a significantly smaller number of yes/no questions. However, for the remaining test sets, in which the ranking is not so good, even in the cases when it is better than the one obtained by the HMM, the mean number of questions is larger. Note that comparing both approaches in terms of ranking is difficult: while the heuristic-based approach uses a ranked list as the base for choosing the word forms to be posed to the user, the new approach uses a decision tree for this. In the case of the tree, the accumulation of probability in the correct candidate is notably more important than its position in a ranking, since this accumulation is what allows to reduce the number of questions to the user. This information nevertheless helps to understand the quality of the prediction of each strategy.

It is also important to analyse the impact of dictionary size in these results. Note that in the case of the decision-tree-based approaches, as the dictionary becomes larger, the number of yes/no questions necessary to determine the correct paradigm is also larger, although the growth rate is very slow. Similarly, the heuristic-based approach requires a larger number of questions as the dictionary size grows, although the heuristic strategy followed by the approach makes it more unstable and the differences between revisions larger. In the case of the approach using decision trees and HMM, the rising number of questions seems to be mitigated by the richer information available for disambiguating the training corpus.

Although a deeper analysis of the behaviour of the different approaches needs to be carried out, it can be concluded that decision-tree-based approaches are more stable and, in general, provide better results in terms of number of yes/no questions than the previous heuristic-based approach.

8 Conclusions and future work

In this work we have presented an approach that combines a hidden Markov model (HMM) and a binary decision tree in order to assist non-expert users in adding new words to monolingual dictionaries. This approach has been compared to the heuristic-based method proposed by Esplà-Gomis et al. (2011). The results have confirmed that the methods based on a decision tree are more stable and usually better than the original one. In addition, the comparison between the method using decision trees only and that combining decision trees and HMMs concluded that the number of queries asked in the second case is significantly lower. The Java code for the resulting system is available[17] under the free/open-source GNU General Public License.[18]

As regards future work, an extended evaluation including more pairs of languages and corpora would be necessary to confirm the results obtained here. It could be also interesting to try to improve the training corpus used, for example, by using a part-of-speech tagger to further reduce the number of compatible paradigms in L for each word form. Moreover, as pointed out in Section 5, a second part of the evaluation should still be performed to determine the feasibility of replacing the n-gram model proposed by Sánchez-Cartagena et al. (2012b) with the probabilities obtained with the HMM for choosing the correct paradigm among a set of equivalent ones.

Acknowledgements

This work has been partially funded by the Spanish Ministerio de Ciencia e Innovación through projects TIN2009-14009-C02-01 and TIN2012-32615, by Generalitat Valenciana through grant ACIF/2010/174 from VALi+d programme, and by the European Commission through project PIAP-GA-2012-324414 (Abu-MaTran).

References

Bartusková, D. and R. Sedlácek. 2002. Tools for semi-automatic assignment of Czech nouns to dec-

[17] https://apertium.svn.sourceforge.net/svnroot/apertium/branches/dictionary-enlargement

[18] http://www.gnu.org/licenses/gpl.html

Revision R_1	Revision R_2	Mean position of correct		Rate correct is first	
		HMM	Feasibility score	HMM	Feasibility score
7217	7287	1.47	**0.51**	70.31	**72.99**
11762	12415	**5.66**	10.45	**28.00**	8.36
17582	20212	1.87	1.72	**52.49**	40.88
27241	27627	7.11	**4.67**	39.73	**42.76**
34649	35985	6.66	**5.18**	45.57	45.57
36838	44118	**1.08**	3.51	**81.08**	70.52

Table 3: For the approach using decision trees and HMM and for the heuristic-based approach, mean position for each test set of the correct paradigm in the ranking of feasibility scores or probabilities and percentage of times that the correct candidate was the one with the highest score or probability.

lination patterns. In *Proceedings of the 5th International Conference on Text, Speech and Dialogue*, pages 159–164.

Baum, L. E. 1972. An inequality and associated maximization technique in statistical estimation for probabilistic functions of a Markov process. *Inequalities*, 3:1–8.

Bojar, O., C. Buck, C. Callison-Burch, C. Federmann, B. Haddow, P. Koehn, C. Monz, M. Post, R. Soricut, and L. Specia. 2013. Findings of the 2013 Workshop on Statistical Machine Translation. In *Proceedings of the Eighth Workshop on Statistical Machine Translation*, pages 1–44.

Cutting, D., J. Kupiec, J. Pedersen, and P. Sibun. 1992. A practical part-of-speech tagger. In *Proceedings of the Third Conference on Applied Natural Language Processing*, pages 133–140.

Desai, S., J. Pawar, and P. Bhattacharyya. 2012. Automated paradigm selection for FSA based Konkani verb morphological analyzer. In *COLING (Demos)*, pages 103–110.

Détrez, G. and A. Ranta. 2012. Smart paradigms and the predictability and complexity of inflectional morphology. In *Proceedings of EACL*, pages 645–653.

Esplà-Gomis, M., V.M. Sánchez-Cartagena, and J.A. Pérez-Ortiz. 2011. Enlarging monolingual dictionaries for machine translation with active learning and non-expert users. In *Proceedings of RANLP*, pages 339–346.

Font-Llitjós, A. 2007. *Automatic improvement of machine translation systems*. Ph.D. thesis, Carnegie Mellon University.

Forcada, M.L., M. Ginestí-Rosell, J. Nordfalk, J. O'Regan, S. Ortiz-Rojas, J.A. Pérez-Ortiz, F. Sánchez-Martínez, G. Ramírez-Sánchez, and F.M. Tyers. 2011. Apertium: a free/open-source platform for rule-based machine translation. *Machine Translation*, 25(2):127–144.

McCreight, E.M. 1976. A space-economical suffix tree construction algorithm. *Journal of the Association for Computing Machinery*, 23:262–272, April.

McShane, M., S. Nirenburg, J. Cowie, and R. Zacharski. 2002. Embedding knowledge elicitation and MT systems within a single architecture. *Machine Translation*, 17:271–305.

Monson, C. 2009. *ParaMor: From Paradigm Structure to Natural Language Morphology Induction*. Ph.D. thesis, Carnegie Mellon University.

Quinlan, J. R. 1986. Induction of decision trees. *Machine Learning*, 1(1):81–106.

Rabiner, L.R. 1989. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286.

Šnajder, J. 2013. Models for Predicting the Inflectional Paradigm of Croatian Words. *Slovenščina 2.0: empirical, applied and interdisciplinary research*, 1(2):1–34.

Sánchez-Cartagena, V.M., M. Esplà-Gomis, and J.A. Pérez-Ortiz. 2012a. Source-language dictionaries help non-expert users to enlarge target-language dictionaries for machine translation. In *Proceedings of LREC*, pages 3422–3429.

Sánchez-Cartagena, V.M., M. Esplà-Gomis, F. Sánchez-Martínez, and J.A. Pérez-Ortiz. 2012b. Choosing the correct paradigm for unknown words in rule-based machine translation systems. In *Proceedings of the Third International Workshop on Free/Open-Source Rule-Based Machine Translation*, pages 27–39.

Walther, G. and L. Nicolas. 2011. Enriching morphological lexica through unsupervised derivational rule acquisition. In *Proceedings of the International Workshop on Lexical Resources*, Ljubljana, Slovenia.

Wang, A., C. Hoang, and M.Y. Kan. 2013. Perspectives on crowdsourcing annotations for natural language processing. *Language Resources and Evaluation*, 47(1):9–31.

Efficient Wordgraph Pruning for Interactive Translation Prediction

Germán Sanchis-Trilles
PRHLT Centre
Univ. Politéc. de Valencia
46022 Valencia, Spain
gsanchis@dsic.upv.es

Daniel Ortiz-Martínez
PRHLT Centre
Univ. Politéc. de Valencia
46022 Valencia, Spain
dortiz@dsic.upv.es

Francisco Casacuberta
PRHLT Centre
Univ. Politéc. de Valencia
46022 Valencia, Spain
fcn@dsic.upv.es

Abstract

When applying interactive translation prediction in real-life scenarios, response time is critical for the users to accept the interactive translation prediction system as a potentially useful tool. In this paper, we report on three different strategies for reducing the computation time required by a state-of-the-art interactive translation prediction system, so that automatic completions are delivered in real time. The best possibility turns out to be to directly prune the wordgraphs derived from the search procedure, achieving real-time response rates without any degradation whatsoever in the quality of the completions provided.

1 Introduction

Despite the recent advances in machine translation (MT) technology, MT systems are not able to provide ready-to-use translations in those contexts where translation accuracy is critical, such as medical or political applications, or even in contexts where correctness is demanded, such as hardware manuals or news texts. This has given rise to increasing research in computer assisted translation (CAT), where the focus is on how to provide a human translator with the best tools available in order to improve the human's efficiency. To this purpose, several ongoing FP7 projects were approved by the European Comission, some of them still being active. These projects pursue a very similar purpose, which is to develop a next generation CAT workbench.

One of the most innovative research directions regarding CAT tools implies interactive translation prediction (ITP) (Barrachina et al., 2009). Under this paradigm, system and human translator interact more closely than in a conventional post-editing setup, and the ITP engine attempts to provide improved completions for the sentence being translated after each one of the interactions of the human translator. Ideally, a constrained decoding, forced to produce the part of the sentence which has already been validated, should be performed before providing every suggestion. However, a full decoding process gives way to an important problem in ITP: the system needs to be able to provide translation completions in real time, since only a small delay in response time could easily lead users to reject the system. For this purpose, a common approximation is to extract a wordgraph off-line, i.e., before the user is actually sitting in front of the CAT tool. Then, during the ITP procedure, suggestions are obtained by searching for the best path in such a graph.

In the present work we report on different approaches analysed for the purpose of reducing the size of the wordgraph mentioned above when using a state-of-the-art ITP system. Since response time is critical, we studied three different strategies and measured the response time in a simulated ITP setup, alongside with an analysis of the degradation of the final translation quality obtained, both in terms of automatic MT metrics and in terms of simulated user effort.

The rest of this paper is structured as follows: in the next section, we briefly review the principles of ITP as an evolution of the classical SMT formulation. Then, in Section 4, we review the theoretical grounds of the strategies studied. Next, Section 5 reports the experiments conducted to assess the quality of the pruned wordgraphs and the response time associated. Finally, Section 6 presents the conclusions of the present work.

2 Statistical Framework

2.1 Statistical Machine Translation

Given a sentence $\mathbf{s}$ in a source language, the discipline of machine translation (MT) studies techniques to obtain its corresponding translation $\mathbf{t}$ in a target language by means of computer. Statistical MT (SMT) formalises this problem as follows (Brown et al., 1993):

$$\hat{\mathbf{t}} = \arg\max_{\mathbf{t}} \Pr(\mathbf{t} \mid \mathbf{s}) \tag{1}$$

$$= \arg\max_{\mathbf{t}} \Pr(\mathbf{t}) \cdot \Pr(\mathbf{s} \mid \mathbf{t}) \tag{2}$$

The terms in the latter equation are the *language model* probability $\Pr(\mathbf{t})$ that represents the well-formedness of $\mathbf{t}$ (*n-gram* models are usually adopted), and the *translation model* $\Pr(\mathbf{s} \mid \mathbf{t})$ that represents the relationship between the source sentence and its translation.

In practice, all of these models (and possibly others) are often combined into a *log-linear model* for $\Pr(\mathbf{t} \mid \mathbf{s})$ (Och and Ney, 2002):

$$\hat{\mathbf{t}} = \arg\max_{\mathbf{t}} \left\{ \sum_{n=1}^{N} \lambda_n \cdot \log(f_n(\mathbf{t}, \mathbf{s})) \right\} \tag{3}$$

where $f_n(\mathbf{t}, \mathbf{s})$ can be any model that represents an important feature for the translation, N is the number of models (or features), and λ_n are the weights of the log-linear combination.

One of the most popular instantiations of log-linear models is that including phrase-based models (Zens et al., 2002; Koehn et al., 2003) (Zens et al., 2002; Koehn et al., 2003). Phrase-based models allow to capture contextual information to learn translations for whole phrases instead of single words. The basic idea of phrase-based translation is to segment the source sentence into phrases, then to translate each source phrase into a target phrase, and finally to reorder the translated target phrases in order to compose the target sentence. Phrase-based models were employed throughout this work.

In log-linear models, the maximisation problem stated by Equation 3 is typically solved by means of dynamic programming-based algorithms (Zens et al., 2002), where the problem of translating a source sentence is decomposed into a set of partial solutions or hypotheses that are solved separately. A given partial hypothesis aligns a certain number of source words with words of the target language, and the rest remain unaligned. These hypotheses are stored in a stack (or priority queue) and ordered by their score. Such a score is given by the log-linear combination of feature functions.

source ($\mathbf{s}$): Para ver la lista de recursos
desired translation ($\hat{\mathbf{t}}$): To view a listing of resources

IT-0	$\mathbf{p}$						
	$\mathbf{t}_s$	*To*	*view*	*the*	*resources*	*list*	
IT-1	$\mathbf{p}$	To	view				
	k			$\boxed{\text{a}}$			
	$\mathbf{t}_s$				*list*	*of*	*resources*
IT-2	$\mathbf{p}$	To	view	a	list		
	k				$\boxed{\text{i}}$		
	$\mathbf{t}_s$				*ng*	*resources*	
IT-3	$\mathbf{p}$	To	view	a	listing		
	k					$\boxed{\text{o}}$	
	$\mathbf{t}_s$					*f*	*resources*
END	$\mathbf{p}$	To	view	a	listing	of	resources

Figure 1: ITP session to translate a Spanish sentence into English. The desired translation is the translation the human user wants to obtain. At IT-0, the system suggests a translation ($\mathbf{t}_s$). At IT-1, the user moves the mouse to accept the first eight characters "To view " and presses the $\boxed{\text{a}}$ key (k), then the system suggests completing the sentence with "*list of resources*" (a new $\mathbf{t}_s$). Iterations 2 and 3 are similar. In the final iteration, the user accepts the current translation.

2.2 Interactive Translation Prediction

Unfortunately, current MT technology is not able to deliver error-free translations. This implies that, in order to achieve good translations, manual post-editing is needed. An alternative to this decoupled approach (first MT, then manual correction) is given by the ITP paradigm (Barrachina et al., 2009). Under this paradigm, translation is considered as an iterative left-to-right process where the human and the computer collaborate to generate the final translation.

Figure 1 shows an example of the ITP approach. There, a source Spanish sentence $\mathbf{s}$ ="Para ver la lista de recursos" is to be translated into a target English sentence $\hat{\mathbf{t}}$. Initially, with no user feedback, the system suggests a complete translation $\mathbf{t}_s$ ="To view the resources list". From this translation, the user marks a prefix $\mathbf{p}$ ="To view" as correct and begins to type the rest of the target sentence. Depending on the system or the user's preferences, the user might type the full next word, or only some letters of it (in our example, the user types the single next character "a"). Then, the MT system suggests a new suffix $\mathbf{t}_s$ ="list of resources" that completes the validated prefix and the input the user has just typed ($\mathbf{p}$ ="To view a"). The interaction continues with a new prefix validation followed, if necessary, by new input from the user, and so on, until the user considers the translation to be complete and satisfactory.

The crucial step of the process is the production of the suffix. Again, decision theory tells us to maximise the probability of the suffix given the available information. Formally, the best suffix of a given length will be:

$$\hat{\mathbf{t}}_s = \arg\max_{\mathbf{t}_s} \Pr(\mathbf{t}_s \mid \mathbf{s}, \mathbf{p}) \tag{4}$$

which can be straightforwardly rewritten as:

$$\hat{\mathbf{t}}_s = \arg\max_{\mathbf{t}_s} \Pr(\mathbf{p}, \mathbf{t}_s \mid \mathbf{s}) \tag{5}$$

$$= \arg\max_{\mathbf{t}_s} \Pr(\mathbf{p}, \mathbf{t}_s) \cdot \Pr(\mathbf{s} \mid \mathbf{p}, \mathbf{t}_s) \tag{6}$$

Note that, since $\mathbf{p}\,\mathbf{t}_s = \mathbf{t}$, this equation is very similar to Equation (2). The main difference is that now the search process is restricted to those target sentences $\mathbf{t}$ that contain $\mathbf{p}$ as prefix. This implies that we can use the same MT models (including the log-linear approach) if the search procedures are adequately modified (Och et al., 2003a). Finally, it should be noted that the statistical models are usually defined at word level, while the ITP process described in this section works at character level. To deal with this problem, during the search process it is necessary to verify the compatibility between $\mathbf{t}$ and $\mathbf{p}$ at character level.

2.3 ITP with Stochastic Error-Correction

A common problem in ITP arises when the user sets a prefix which cannot be explained by the statistical models. To solve this problem, ITP systems typically include ad-hoc error-correction techniques to guarantee that the suffixes can be generated (Barrachina et al., 2009). As an alternative to this heuristic approach, Ortiz-Martínez (2011) proposed a formalisation of the ITP framework that does include stochastic error-correction models in its statistical formalisation. The starting point of this alternative ITP formalisation accounts for the problem of finding the translation $\mathbf{t}$ that, at the same time, better explains the source sentence $\mathbf{s}$ and the prefix given by the user $\mathbf{p}$:

$$\hat{\mathbf{t}} = \arg\max_{\mathbf{t}} \Pr(\mathbf{t} \mid \mathbf{s}, \mathbf{p}) \tag{7}$$

$$= \arg\max_{\mathbf{t}} \Pr(\mathbf{t}) \cdot \Pr(\mathbf{s}, \mathbf{p} \mid \mathbf{t}) \tag{8}$$

The following naïve Bayes' assumption is now made: the source sentence $\mathbf{s}$ and the user prefix $\mathbf{p}$ are statistically independent variables given the translation $\mathbf{t}$, obtaining:

$$\hat{\mathbf{t}} = \arg\max_{\mathbf{t}} \Pr(\mathbf{t}) \cdot \Pr(\mathbf{s} \mid \mathbf{t}) \cdot \Pr(\mathbf{p} \mid \mathbf{t}) \tag{9}$$

where $\Pr(\mathbf{t})$ can be approximated with a language model, $\Pr(\mathbf{s} \mid \mathbf{t})$ can be approximated with a translation model, and $\Pr(\mathbf{p} \mid \mathbf{t})$ can be approximated by an error correction model that measures the compatibility between the user-defined prefix $\mathbf{p}$ and the hypothesized translation $\mathbf{t}$.

Note that the translation result, $\hat{\mathbf{t}}$, given by Equation (9) may not contain $\mathbf{p}$ as prefix because every translation is compatible with $\mathbf{p}$ with a certain probability. Thus, despite being close, Equation (9) is not equivalent to the ITP formalisation in Equation (6).

To solve this problem, we define an alignment, $\mathbf{a}$, between the user-defined prefix $\mathbf{p}$ and the hypothesised translation $\mathbf{t}$, so that the unaligned words of $\mathbf{t}$, in an appropriate order, constitute the suffix searched in ITP. This allows us to rewrite the error correction probability as follows:

$$\Pr(\mathbf{p} \mid \mathbf{t}) = \sum_{\mathbf{a}} \Pr(\mathbf{p}, \mathbf{a} \mid \mathbf{t}) \tag{10}$$

To simplify things, we assume that $\mathbf{p}$ is monotonically aligned to $\mathbf{t}$, leaving the potential word-reordering to the language and translation models. Under this assumption, $\mathbf{a}$ determines an alignment for $\mathbf{t}$, such that $\mathbf{t} = \mathbf{t}_p\mathbf{t}_s$, where $\mathbf{t}_p$ is fully-aligned to $\mathbf{p}$ and $\mathbf{t}_s$ remains unaligned. Taking all these things into consideration, and following a maximum approximation, we finally arrive to the expression:

$$(\hat{\mathbf{t}}, \hat{\mathbf{a}}) = \arg\max_{\mathbf{t},\mathbf{a}} \Pr(\mathbf{t}) \cdot \Pr(\mathbf{s} \mid \mathbf{t}) \cdot \Pr(\mathbf{p}, \mathbf{a} \mid \mathbf{t}) \tag{11}$$

where the suffix required in ITP is obtained as the portion of $\hat{\mathbf{t}}$ that is not aligned with the user prefix.

In practice, the models in Equation (11) are combined in a log-linear fashion as it is typically done in SMT (see Equation (3)).

2.4 ITP Using Wordgraphs

Common ITP implementations rely on a *wordgraph* data structure that represents possible translations of the given source sentence. A wordgraph is a weighted directed acyclic graph, in which each node represents a partial translation hypothesis and each edge is labelled with a word (or group of words) of the target sentence and is weighted according to the scores given by an SMT model (see (Ueffing et al., 2002) for more details). The use of wordgraphs in ITP has been studied in (Barrachina et al., 2009; Ortiz-Martínez, 2011; González-Rubio et al., 2013) in combination with different translation techniques.

The main advantage of wordgraph-based ITP systems is their efficiency in terms of the time cost per each interaction. This is due to the fact that the wordgraph is generated only once at the beginning of the interactive translation process of a given source sentence, and the suffixes required in ITP can be obtained by incrementally processing this wordgraph at each interaction.

All of the experiments reported in this paper always included stochastic error-correction for recovering from prefixes that cannot explained by the wordgraph for a given sentence.

3 Related Work

The use of wordgraphs in SMT was introduced in (Ueffing et al., 2002) for single word models and later extended to phrase-based models in (Zens and Ney, 2005). However, in these works, wordgraphs were applied within a fully-automatic SMT context. The first study on wordgraphs for ITP was given in (Och et al., 2003b). In that work, wordgraph pruning is used to speed-up suffix generation in an early ITP system based on the alignment template formalism. Bender et al. (Bender et al., 2005) extended the same strategy to a phrase-based ITP system with ad-hoc error correction techniques (see Section 2.3). Here, we propose efficient wordgraph pruning techniques for a state-of-the-art ITP system with stochastic error correction.

4 Efficient Suffix Generation in ITP

As it was explained in Section 2.4, common ITP formalisations, and more specifically, the one adopted in this paper, are typically based on the generation of wordgraphs.

The computational complexity of suffix generation using wordgraphs is linear in the number wordgraph states (Amengual and Vidal, 1998). Because of this, one possible way to achieve efficiency improvements would be to reduce the number of states per each wordgraph. One possible way to obtain smaller wordgraphs is to modify the pruning parameters that are applied during the decoding stage. Since wordgraphs constitute a compact representation of the search space explored by the SMT system, their size would be smaller if the search space is reduced as well. Another possibility to reduce the number of states contained in the wordgraph would be to apply pruning techniques directly over it.

4.1 Modifying Wordgraph Size in Decoding Time

To reduce the search space, regular SMT decoders based on dynamic programming have two different pruning parameters, namely, threshold pruning and histogram pruning (Och and Ney, 2002):

- **Threshold Pruning**: threshold pruning is applied for the different subsets of partial hypotheses that share the same number of already aligned source words. For a given subset, all those hypotheses whose score is below a certain percentage of the score of the best hypothesis for that subset are removed. The specific percentage used corresponds to the pruning threshold parameter.

- **Histogram Pruning**: the idea behind histogram pruning is to order those hypotheses that share the same number of already aligned source words by descending order of their scores, keeping only a certain quantity of the best of them.

4.2 Wordgraph Pruning

Threshold and histogram pruning constitute two possible techniques to reduce the wordgraph size during the decoding stage. Once the wordgraph has been generated, its size can be directly reduced using a technique that is closely related to threshold pruning. For this purpose, the probability of the best sentence hypothesis in the wordgraph is determined. After that, all those hypotheses in the graph whose probability is lower than this maximum probability multiplied by the pruning threshold are discarded. This wordgraph pruning technique was introduced in (Sixtus and Ortmanns, 1999) within the context of speech recognition.

The main difference between histogram and wordgraph pruning is that the former performs hypothesis pruning according to the score of the best partial hypothesis having a certain number of already aligned source words (i.e. pruning is locally applied) while the latter performs hypothesis pruning based on the probability of the best sentence hypothesis (a global pruning criterion is used).

If the wordgraph pruning threshold is zero, then the wordgraph is not pruned at all, and if the threshold is one, then only the sentence with maximum probability is retained.

5 Experimental Setup

In this section we detail the experimental setup designed to evaluate the different wordgraph size reduction strategies described in the previous section.

5.1 Corpora Used

The SMT system used to produce the translation models which later on were used to generate the wordgraphs were trained on the data provided for the ACL 2013 Workshop on Statistical Machine Translation (Bojar et al., 2013). Four training data sets were provided in this workshop: the Europarl corpus, the United Nations corpus, the Common Crawl corpus and the News Commentary corpus. Statistics of these data sets are provided in Table 1. As shown, these corpora together constitute a fair amount of data, and training an SMT system with all these data is computationally costly.

Additional development and test data were also considered (Table 2). For tuning the log-linear weights present in Equation 3, the test sets of the WMT 2008 to 2010 were considered, and the test set of WMT 2011 was considered as test data for the final evaluation.

5.2 System Description

For building the final ITP system, initial translation models were built by means of the open source Moses toolkit (Koehn et al., 2007)[1]. Then, the Moses decoder was also used for generating the wordgraphs. For doing this, the standard decoder configuration was used, i.e. a statistical log-linear model including a phrase-based translation model, a language model, a distortion model and word and phrase penalties. The baseline system was set up using the default threshold and histogram pruning parameters, i.e., 200 for the histogram pruning (200 maximum stack size) and 0.00001 for threshold pruning (hypothesis with a score less than 0.00001 times the best hypothesis are discarded). The weights of the log-linear combination are optimised by means of the Minimum Error Rate Training (MERT) procedure (Och, 2003).

		Es	En
Europarl	Sentences	1.9M	
	Run. words	54.0M	51.6M
	Vocabulary	181k	120.9k
United Nations	Sentences	10.8M	
	Run. words	317.6M	278.5M
	Vocabulary	612.0k	598.7k
Common Crawl	Sentences	1.8M	
	Run. words	46.7M	44.2M
	Vocabulary	763k	675.7k
News Com.	Sentences	172.8k	
	Run. words	5.0M	4.4M
	Vocabulary	88.8k	65.5k
Total	Sentences	14.7M	
	Run. words	423.3M	378.7M
	Vocabulary	1.2M	1.2M

Table 1: Statistics of the training data used in our experiments. These statistics are computed in tokenised and de-truecased conditions.

		Es	En
WMT08-10 Test	Sentences	7065	
	Run. words	186.2k	177.3k
	OoV words	1105	1073
WMT11 Test	Sentences	3003	
	Run. words	79.4k	74.8k
	OoV words	444	537

Table 2: Statistics of the WMT 2011 test data used to evaluate the system. These statistics are computed in tokenised and de-truecased conditions.

The phrase-based translation model provides direct and inverted frequency-based and lexical-based probabilities for each phrase pair in the phrase table. Phrase pairs are extracted from symmetrised word alignments generated by GIZA++ (Och and Ney, 2003). A 5-gram word-based LM is estimated on the target side of the parallel corpora using the improved Kneser-Ney smoothing (Chen and Goodman, 1999).

[1] Available from http://www.statmt.org/moses/

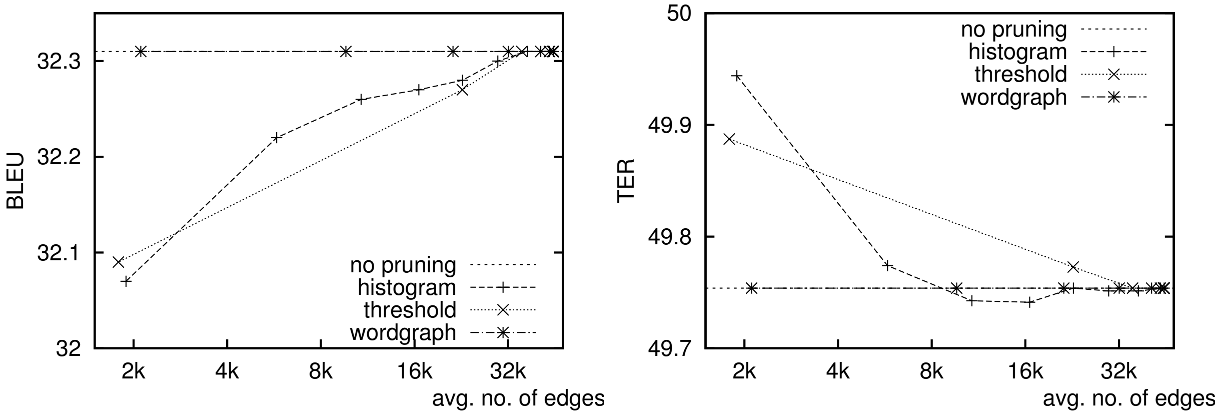


Figure 2: Translation quality of the best path in the reduced wordgraphs, measured both in terms of TER and BLEU. Note that the x-axis is in logarithmic scale for readability purposes.

For modelling word reordering, in addition to a negative-exponential on the reordering distance, a model conditioned on phrase pairs was estimated, namely the "orientation-bidirectional-fe" distortion model (Koehn et al., 2005).

Once the wordgraphs had been built, they were fed into the open-source Thot toolkit (Ortiz-Martínez and Casacuberta, 2014) [2], which implements among other things the ITP functionality used in this work. Such functionality allowed us to simulate real users by using the reference of the test data present in the corpus.

5.3 Assessment Metrics

The results produced by the ITP systems associated to the different wordgraph size reduction strategies were evaluated both in terms of conventional SMT metrics and ITP metrics. More specifically, the metrics used were:

- BLEU (Papineni et al., 2001) (Bilingual Language Evaluation Understudy) is an SMT precision metric that measures precision of unigrams, bigrams, trigrams and 4-grams, with a penalty for sentences that are too short.

- TER (Snover et al., 2006) (Translation Edit Rate) is an SMT error metric that computes the minimum number of edits required to modify the system hypotheses so that they match the references. Possible edits include insertion, deletion, substitution of single words and shifts of word sequences.

- KSMR (Barrachina et al., 2009) (Key Stroke Mouse-action Ratio) is an ITP error metric that measures the number of actions required by a human user to amend the system hypotheses so that they match the reference the user has in mind. Actions considered include key-strokes and the positioning of the mouse.

5.4 Results

We conducted experiments by testing different pruning thresholds according to the different strategies defined in Section 4. Figure 2 reports the final BLEU and TER scores achieved by the best hypothesis still present in the wordgraph after pruning has taken place. It is interesting to see that the wordgraph pruning strategy does not present any degradation as measured by TER and BLEU scores, while the other strategies do seem to correlate wordgraph size and translation quality. This is explained by the definition itself of wordgraph pruning strategy: since it only prunes the paths which fall beneath a given proportion of the probability of the best path, the best path itself is always preserved.

More interesting are the results of the ITP simulation, reported in Figure 3. Here it is shown that, just as in the case of BLEU and TER, KSMR seems to correlate quite evenly with wordgraph size in the case of histogram and threshold threshold strategies. However, when pruning the wordgraph directly, the human effort required to amend the hypotheses, as measured by KSMR, does not increase, and even presents a slight improvement for threshold values of 0.2 and 0.4 (equivalent to 21.3k and 9.6k edges on average, respectively). However, such improvement is not statistically significant and might be due to the effect of the stochastic error correction described in

[2] Available from https://github.com/daormar/thot/

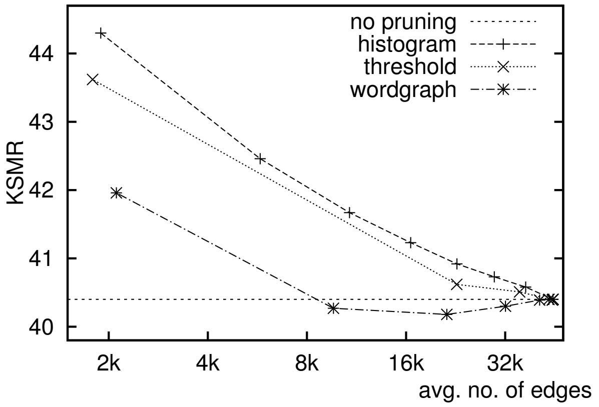


Figure 3: KSMR results when comparing different wordgraph sizes and the different pruning strategies described. Note that the x-axis is in logarithmic scale for readability purposes.

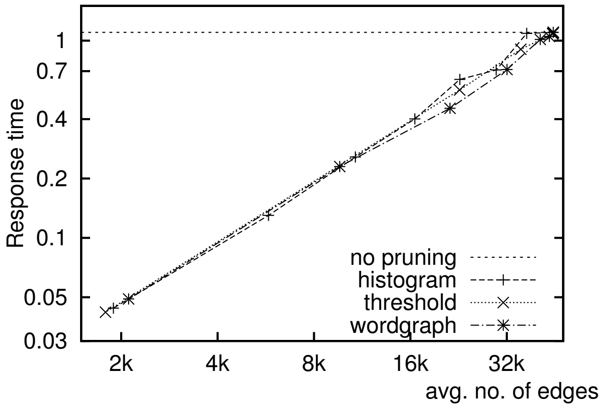


Figure 4: Average response time in seconds of the different systems when considering the different wordgraph size reduction strategies. Note that both the x-axis and the y-axis are in logarithmic scale.

Section 2.3. Nevertheless, it is important to point out that for these threshold values the amount of edges present in the wordgraph is reduced drastically, while no degradation in the performance of the system is observed until only around 8000 are left present in the wordgraph. The difference in behaviour between the BLEU and TER curves, on the one side, and the KSMR curves on the other side lies on the fact that KSMR is computed as an ITP simulation, and hence requires more information from the wordgraph than just its best path.

Finally, Figure 4 reports on the final response time required by the system. The experiments detailed here were performed on a multi-processor Intel Xeon E5-2650 @ 2.00GHz machine, with 64GBs memory, although each of the ITP simulations was not parallelised (i.e., each ITP simulation was executed sequentially). As shown, the complete wordgraph presents response times which are too high for a system set for online production. One could difficultly imagine that a potential user would wait for one second on average (much more in some cases) for the system to produce a hypothesis completion. However, by reducing the wordgraph by means of the wordgraph pruning strategy we are able to achieve real-time response times, while not having to compromise translation quality or human effort. Response time correlates empirically evenly with wordgraph size. When considering the 0.2 and 0.4 thresholds of the wordgraph pruning strategy, it was observed that the average response times were of 0.23 and 0.05 seconds, respectively, which is perfectly suitable for an ITP system set for online production. Moreover, it must be emphasised that such speed increase is achieved without any degradation of system performance measured in terms of KSMR.

6 Conclusions

In this paper we have compared three approaches for obtaining smaller-sized wordgraphs for the purpose of providing sentence completions by means of a state-of-the-art ITP engine. We have seen that regular wordgraphs, as produced by a state-of-the-art decoder, imply too much computational time for their usage within an ITP system. We have also shown that pruning the wordgraph directly by removing those paths whose probability falls below a certain proportion of the probability of the best path is able to yield completions with exactly the same quality as the un-pruned wordgraphs, but with much better response times.

We understand that the analysis performed in this work is crucial for research in ITP, since hypothesis completion times above one second can be considered unacceptable for a human translator. The pruning techniques proposed in this paper allow us to solve this issue effectively.

Acknowledgements

Work supported by the European Union 7th Framework Program (FP7/2007-2013) under the CasMaCat project (grant agreement nº 287576), by Spanish MICINN under grant TIN2012-31723, and by the Generalitat Valenciana under grant ALMPR (Prometeo/2009/014).

References

Amengual, J.C. and E. Vidal. 1998. Efficient error-correcting viterbi parsing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol.PAMI-20, No.10:1109–1116, October.

Barrachina, Sergio, Oliver Bender, Francisco Casacuberta, Jorge Civera, Elsa Cubel, Shahram Khadivi, Antonio Lagarda, Hermann Ney, Jesús Tomás, Enrique Vidal, and Juan-Miguel Vilar. 2009. Statistical approaches to computer-assisted translation. *Computational Linguistics*, 35:3–28, March.

Bender, O., S. Hasan, D. Vilar, R. Zens, and H. Ney. 2005. Comparison of generation strategies for interactive machine translation. In *Conference of the European Association for Machine Translation*, pages 33–40, Budapest, Hungary, May.

Bojar, Ondrej, Christian Buck, Chris Callison-Burch, Barry Haddow, Philipp Koehn, Christof Monz, Matt Post, Herve Saint-Amand, Radu Soricut, and Lucia Specia, editors. 2013. *Proceedings of the Eighth Workshop on Statistical Machine Translation*. Association for Computational Linguistics, Sofia, Bulgaria, August.

Brown, Peter F., Vincent J. Della Pietra, Stephen A. Della Pietra, and Robert L. Mercer. 1993. The mathematics of statistical machine translation: parameter estimation. *Computational Linguistics*, 19:263–311.

Chen, Stanley F. and Joshua Goodman. 1999. An empirical study of smoothing techniques for language modeling. *Computer Speech and Language*, 4(13):359–393.

González-Rubio, Jesús, Daniel Ortiz-Martínez, José-Miguel Benedí, and Francisco Casacuberta. 2013. Interactive machine translation using hierarchical translation models. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 244–254.

Koehn, P., F. J. Och, and D. Marcu. 2003. Statistical phrase-based translation. In *Proceedings of the Human Language Technology and North American Association for Computational Linguistics Conference*, pages 48–54, Edmonton, Canada, May.

Koehn, P., A. Axelrod, A. Birch Mayne, C. Callison-Burch, M. Osborne, and D. Talbot. 2005. Edinburgh System Description for the 2005 IWSLT Speech Translation Evaluation. In *Proc. of IWSLT*, Pittsburgh, PA.

Koehn et al., P. 2007. Moses: Open source toolkit for statistical machine translation. In *Proc. of the ACL Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic.

Och, Franz Josef and Hermann Ney. 2002. Discriminative training and maximum entropy models for statistical machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 295–302.

Och, F. J. and H. Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.

Och, Franz Josef, Richard Zens, and Hermann Ney. 2003a. Efficient search for interactive statistical machine translation. In *Proceedings of the European chapter of the Association for Computational Linguistics*, pages 387–393.

Och, Franz Josef, Richard Zens, and Hermann Ney. 2003b. Efficient search for interactive statistical machine translation. In *In EACL 03: Proceedings of the tenth conference on European chapter of the Association for Computational Linguistics*, pages 387–393.

Och, F.J. 2003. Minimum Error Rate Training in Statistical Machine Translation. In *Proc. of ACL*, pages 160–167, Sapporo, Japan.

Ortiz-Martínez, D. and F. Casacuberta. 2014. The new Thot toolkit for fully automatic and interactive statistical machine translation. In *14th Annual Meeting of the European Association for Computational Linguistics: System Demonstrations*, pages 45–48, Gothenburg, Sweden, April.

Ortiz-Martínez, Daniel. 2011. *Advances in Fully-Automatic and Interactive Phrase-Based Statistical Machine Translation*. Ph.D. thesis, Universitat Politècnica de València. Advisors: Ismael García Varea and Francisco Casacuberta.

Papineni, K., A. Kishore, S. Roukos, T. Ward, and W. Jing Zhu. 2001. Bleu: A method for automatic evaluation of machine translation. In *Technical Report RC22176 (W0109-022)*.

Sixtus, Achim and Stefan Ortmanns. 1999. High quality word graphs using forward-backward pruning. In *Proceedins of the ICASSP*, pages 593–596.

Snover, Matthew, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proc. of AMTA'06*.

Ueffing, N., F. Och, and H. Ney. 2002. Generation of word graphs in statistical machine translation. pages 156–163.

Zens, Richard and Hermann Ney. 2005. Word graphs for statistical machine translation. In *Proceedings of the ACL Workshop on Building and Using Parallel Texts*, pages 191–198, Ann Arbor, Michigan, June. Association for Computational Linguistics.

Zens, R., F. J. Och, and H. Ney. 2002. Phrase-based statistical machine translation. In *Advances in artificial intelligence. 25. Annual German Conference on AI*, volume 2479 of *LNCS*, pages 18–32. Springer Verlag, September.

Translation Model Based Weighting for Phrase Extraction

Saab Mansour and **Hermann Ney**
Human Language Technology and Pattern Recognition
Computer Science Department
RWTH Aachen University
Aachen, Germany
{mansour,ney}@cs.rwth-aachen.de

Abstract

Domain adaptation for statistical machine translation is the task of altering general models to improve performance on the test domain. In this work, we suggest several novel weighting schemes based on translation models for adapted phrase extraction. To calculate the weights, we first phrase align the general bilingual training data, then, using domain specific translation models, the aligned data is scored and weights are defined over these scores. Experiments are performed on two translation tasks, German-to-English and Arabic-to-English translation with lectures as the target domain. Different weighting schemes based on translation models are compared, and significant improvements over automatic translation quality are reported. In addition, we compare our work to previous methods for adaptation and show significant gains.

1 Introduction

In recent years, large amounts of monolingual and bilingual training corpora were collected for statistical machine translation (SMT). Early years focused on structured data translation such as newswire and parliamentary discussions. Nowadays, new domains of translation are being explored, such as talk translation in the IWSLT TED evaluation (Cettolo et al., 2012) and patents translation at the NTCIR PatentMT task (Goto et al., 2013).

The task of domain adaptation tackles the problem of utilizing existing resources mainly drawn from one domain (e.g. newswire, parliamentary discussion) to maximize the performance on the test domain (e.g. lectures, web forums).

The main component of an SMT system is the phrase table, providing the building blocks (i.e. phrase translation pairs) and corresponding translation model scores (e.g., phrase models, word lexical smoothing, etc.) to search for the best translation. In this work, we experiment with phrase model adaptation through training data weighting, where one assigns higher weights to relevant domain training instances, thus causing an increase of the corresponding probabilities. As a result, translation pairs which can be obtained from relevant training instances will have a higher chance of being utilized during search.

The main contribution of this work is designing several novel schemes for scoring sentences and assigning them appropriate weights to manifest adaptation. Our method consists of two steps: first, we find phrase alignments for the bilingual training data, then, the aligned data is scored using translation models and weights are generated.

Experiments using the suggested methods and a comparison to previous work are done on two tasks: Arabic-to-English and German-to-English TED lectures translation. The results show significant improvements over the baseline, and significant improvements over previous work are reported when combining our suggested methods with previous work.

The rest of the paper is organized as follows. Related work on adaptation and weighting is detailed in Section 2. The weighted phrase extraction training and the methods for assigning weights using translation models are described in

Section 3 and Section 4 correspondingly. Experimental setup including corpora statistics and the SMT system used in this work are described in Section 5. The results of the suggested methods are summarized in Section 6 and error analysis is given in Section 7. Last, we conclude with few suggestions for future work.

2 Related Work

A broad range of methods and techniques have been suggested in the past for domain adaptation for SMT. In recent work, language model and phrase model adaptation received most of the attention. In this work, we focus on phrase model adaptation. A prominent approach in recent work for phrase model adaptation is training samples weighting at different levels of granularity. Foster and Kuhn (2007) perform phrase model adaptation using mixture modeling at the corpus level. Each corpus in their setting gets a weight using various methods including language model (LM) perplexity and information retrieval methods. Interpolation is then done linearly or log-linearly. The weights are calculated using the development set therefore expressing adaptation to the domain being translated. A finer grained weighting is that of (Matsoukas et al., 2009), who assign each sentence in the bitexts a weight using features of meta-information and optimizing a mapping from feature vectors to weights using a translation quality measure over the development set. Foster et al. (2010) perform weighting at the phrase level, using a maximum likelihood term limited to the development set as an objective function to optimize. They compare the phrase level weighting to a "flat" model, where the weight directly models the phrase probability. In their experiments, the weighting method performs better than the flat model, therefore, they conclude that retaining the original relative frequency probabilities of the phrase model is important for good performance.

Data filtering for adaptation (Moore and Lewis, 2010; Axelrod et al., 2011) can be seen as a special case of the sample weighting method where a weight of 0 is assigned to discard unwanted samples. These methods rely on an LM based score to perform the selection, though the filtered data will affect the training of other models such as the phrase model and other translation models. LM based scoring might be more appropriate for LM adaptation but not as much for phrase model adaptation as it does not capture bilingual dependencies. We score training data instances using translation models and thus model connections between source and target sentences.

In this work, we compare several scoring schemes at the sentence level for weighted phrase extraction. Additionally, we experiment with new scoring methods based on translation models used during the decoding process. In weighting, all the phrase pairs are retained, and only their probability is altered. This allows the decoder to make the decision whether to use a phrase pair or not, a more methodological way than removing phrase pairs completely when filtering.

3 Weighted Phrase Extraction

The classical phrase model is estimated using relative frequency:

$$p(\tilde{f}|\tilde{e}) = \frac{\sum_r c_r(\tilde{f}, \tilde{e})}{\sum_{\tilde{f}'} \sum_r c_r(\tilde{f}', \tilde{e})} \quad (1)$$

Here, $\tilde{f}, \tilde{e}$ are contiguous phrases, $c_r(\tilde{f}, \tilde{e})$ denotes the count of $(\tilde{f}, \tilde{e})$ being a translation of each other in sentence pair (f_r, e_r). One method to introduce weights to eq. (1) is by weighting each sentence pair by a weight w_r. Eq. (1) will now have the extended form:

$$p(\tilde{f}|\tilde{e}) = \frac{\sum_r w_r \cdot c_r(\tilde{f}, \tilde{e})}{\sum_{\tilde{f}'} \sum_r w_r \cdot c_r(\tilde{f}', \tilde{e})} \quad (2)$$

It is easy to see that setting $\{w_r = 1\}$ will result in eq. (1) (or any non-zero equal weights). Increasing the weight w_r of the corresponding sentence pair will result in an increase of the probabilities of the phrase pairs extracted. Thus, by increasing the weight of in-domain sentence pairs, the probability of in-domain phrase translations could also increase.

We perform weighting rather than filtering for adaptation as the former was shown to achieve better results (Mansour and Ney, 2012).

Next, we discuss several methods for setting the weights in a fashion which serves adaptation.

4 Weighting Schemes

Several weighting schemes can be devised to manifest adaptation. Previous work suggested perplexity based scoring to perform adaptation (e.g. (Moore and Lewis, 2010)). The basic idea is to generate a model using an in-domain training data and measure the perplexity of the in-domain model on new events to rank their relevance to the in-domain. We recall this method in Section 4.1.

In this work, we suggest to use several phrase-based translation models to perform scoring. The basic idea of adaptation using translation models is similar to the perplexity based method. We use an in-domain training data to estimate translation model scores over new events. Further details of the method are given in Section 4.2.

4.1 LM Perplexity Weighting

LM cross-entropy scoring can be used for both monolingual and bilingual data filtering (Moore and Lewis, 2010; Axelrod et al., 2011). Next, we recall the scoring methods introduced in the above previous work and utilize it for our proposed weighted phrase extraction method.

The scores for each sentence in the general-domain corpus are based on the cross-entropy difference of the in-domain (IN) and general-domain (GD) models. Denoting $H_{LM}(x)$ as the cross entropy of sentence x according to LM, then the cross entropy difference $DH_{LM}(x)$ can be written as:

$$DH_{LM}(x) = H_{LM_{IN}}(x) - H_{LM_{GD}}(x) \qquad (3)$$

The intuition behind eq. (3) is that we are interested in sentences as close as possible to the in-domain, but also as far as possible from the general corpus. Moore and Lewis (2010) show that using eq. (3) for filtering performs better in terms of perplexity than using in-domain cross-entropy only ($H_{LM_{IN}}(x)$). For more details about the reasoning behind eq. (3) we refer the reader to (Moore and Lewis, 2010).

Axelrod et al. (2011) adapted the LM scores for bilingual data filtering for the purpose of TM training. The bilingual cross entropy difference for a sentence pair (f_r, e_r) in the GD corpus is then defined by:

$$d_r = DH_{LM_{source}}(f_r) + DH_{LM_{target}}(e_r)$$

We utilize d_r for our suggested weighted phrase extraction. d_r can be assigned negative values, and lower d_r indicates sentence pairs which are more relevant to the in-domain. Therefore, we negate the term d_r to get the notion of higher weights indicating sentences being closer to the in-domain, and use an exponent to ensure positive values. The final weight is of the form:

$$w_r = e^{-d_r} \qquad (4)$$

This term is proportional to perplexities and inverse perplexities, as the exponent of entropy is perplexity by definition.

4.2 Translation Model Weighting

In state-of-the-art SMT several models are used during decoding to find the best scoring hypothesis. The models include, phrase translation probabilities, word lexical smoothing, reordering models, etc. We utilize these translation models to perform sentence weighting for adaptation. To estimate the models' scores, a phrase alignment is required. We use the forced alignment (FA) phrase training procedure (Wuebker et al., 2010) for this purpose. The general FA procedure will be presented next followed by an explanation how we estimate scores for adaptation using FA.

4.2.1 Forced Alignment Training

The standard phrase extraction procedure in SMT consists of two phases: *(i)* word-alignment training (e.g., IBM alignment models), *(ii)* heuristic phrase extraction and relative frequency based phrase translation probability estimation.

In this work, we utilize phrase training using the FA method for the task of adaptation. Unlike heuristic phrase extraction, the FA method performs actual phrase training. In the standard FA procedure, we are given a training set, from which an initial heuristics-based phrase table p^0 is generated. FA training is then done by running a normal SMT decoder (using p^0 phrases and models) on the training data and constrain the translation to the given target instance. Forced decoding generates n-best possible phrase alignments from which we are interested in the first-best (viterbi) one. Note that we do not use FA to generate a trained phrase table but only to get phrase alignments of the bilingual training data. We explain next how to utilize FA training for adaptation.

4.2.2 Scoring

The proposed method for calculating translation model scores using FA is depicted in Figure 1. We start by training the translation models using the standard heuristic method over the in-domain portion of the training data. We then use these in-domain translation models to perform the FA pro-

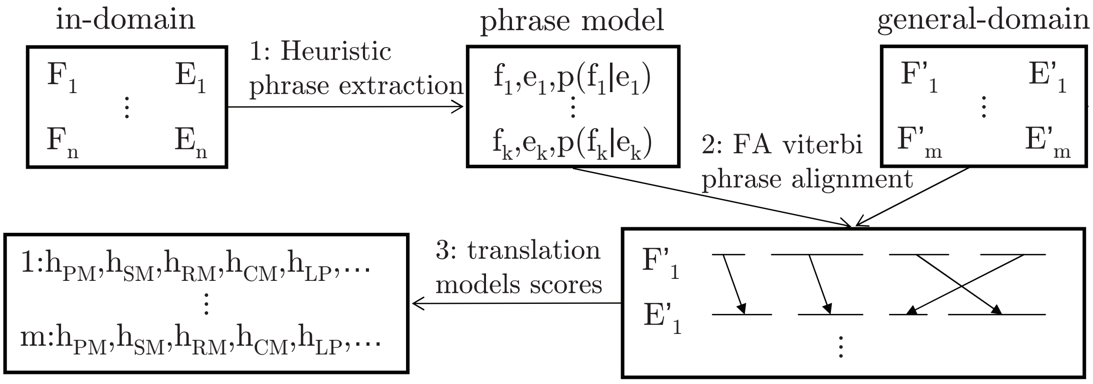

in-domain
F1
E1
Fn
En
1: Heuristic phrase extraction
phrase model
f1,e1,p(f1|e1)
fk,ek,p(fk|ek)
general-domain
F'1
E'1
F'm
E'm
2: FA viterbi phrase alignment
3: translation models scores
1:hPM,hSM,hRM,hCM,hLP,...
m:hPM,hSM,hRM,hCM,hLP,...
F'1
E'1

Figure 1: Translation model scores generation for general-domain sentence pairs using in-domain corpus and viterbi phrase alignments calculated by the FA procedure.

cedure over the general-domain (GD) data. The FA procedure provides n-best possible phrase alignments, but we are interested only in one alignment. Even though the IN data is small, we ensure that all GD sentences are phrase aligned using backoff phrases (Wuebker and Ney, 2013). Using the viterbi (first-best) phrase alignment and the in-domain models again, we generate the translation model scores for GD sentences. As the scores are calculated by IN models, they express the relatedness of the scored sentence to the in-domain. Note that the FA procedure for getting adaptation weights is different from the standard FA procedure. In the standard FA procedure, the same corpus is used to generate the initial heuristic phrase table as well as phrase training. The FA procedure to obtain adaptation weights uses an initial phrase table extracted from IN while the training is done over GD.

Next, we define the process for generating the scores with mathematical notation. Given a training sentence pair (f_1^J, e_1^I) from the GD corpus, we force decode $f_1^J = f_1...f_J$ into $e_1^I = e_1...e_I$ using the IN phrase table. The force decoding process generates a viterbi phrase alignment $s_1^K = s_1...s_K$, $s_k = (b_k, j_k; i_k)$ where (b_k, j_k) are the source phrase $\tilde{f}_k$ begin and end positions correspondingly, and i_k is the end position of translation target phrase $\tilde{e}_k$ (the start position of $\tilde{e}_k$ is $i_{k-1}+1$ by definition of phrase based translation). Using s_1^K we calculate the scores of 10 translation models which are grouped into 5 weighting schemes:

- PM: phrase translation models in both source-to-target (s2t) and target-to-source (t2s) directions

$$h_{PM_{s2t}}(f_1^J, e_1^I, s_1^K) = \sum_{k=1}^{K} \log p(\tilde{f}_k|\tilde{e}_k)$$

The t2s direction is defined analogously using the $p(\tilde{e}_k|\tilde{f}_k)$ probabilities.

- SM: word lexical smoothing models also in both translation directions

$$h_{SM_{s2t}}(f_1^J, e_1^I, s_1^K) = \sum_{k=1}^{K} \sum_{j=b_k}^{j_k} \log \sum_{i=i_{k-1}+1}^{i_k} p(f_j|e_i)$$

- RM: distance based reordering model

$$h_{RM}(f_1^J, e_1^I, s_1^K) = \sum_{k=1}^{K} |b_k - j_{k-1} + 1|$$

- CM: phrase count models

$$h_{CM_i}(f_1^J, e_1^I, s_1^K) = \sum_{k=1}^{K} \Big[c(\tilde{f}_k, \tilde{e}_k) < i\Big]$$

i is assigned the values 2,3,4 (3 count features). $c(\tilde{f}, \tilde{e})$ is the count of the bilingual phrase pair being aligned to each other (in the IN corpus).

- LP: length based word and phrase penalties

$$h_{LP_{wordPernalty}}(f_1^J, e_1^I, s_1^K) = I$$

$$h_{LP_{phrasePenalty}}(f_1^J, e_1^I, s_1^K) = K$$

We experiment with the PM scheme independently. In addition, we try using all models in a loglinear fashion for weighting (denoted by TM), and using TM and LM combined score (denoted by TM+LM). We use the decoder optimized lambdas to combine the models.

To obtain the weights for a scheme which is composed of a set of models $\{h_1^n\}$, we normalize (the sum of absolute values equals 1) the corresponding lambdas obtaining $\{\lambda_1^n\}$, and calculate:

$$w(f,e,s) = e^{-\sum_{i=1}^{n} \lambda_i \cdot h_i(f,e,s)}$$

An alternative method to perform adaptation by force aligning GD using IN would be performing phrase probability re-estimation as done in the final step of standard FA training. In this case, n-best phrase alignments are generated for each sentence in GD using the IN models and the phrase model is then reestimated using relative frequencies on the n-bests. This way we directly use the FA procedure to generate the translation models. The problem with this approach is that due to the small size of IN, some sentences in GD can not be decoded with the initial phrase table and fallback runs using backoff phrases need to be used (Wuebker and Ney, 2013). Backoff phrases of a sentence pair contain all source and target sub-strings up-to a defined maximum length. Therefore, many of these backoff phrase pairs are not a translation of each other. Using such phrases to reestimate the phrase model might generate unwanted phrase translation candidates. In the case of weighting, the backoff probabilities are used indirectly to weight the initial counts, in addition, combining with other model scores remedies the problem further.

Another way to perform adaptation using FA is by starting with a GD heuristic phrase table and utilize it to force decode IN. This way, the probabilities of the general phrase model are biased towards the in-domain distribution. This method was presented by (Mansour and Ney, 2013) and will be compared to our work.

5 Experimental Setup

5.1 Training Corpora

To evaluate the introduced methods experimentally, we use the IWSLT 2011 TED Arabic-to-English and German-to-English translation tasks. The IWSLT 2011 evaluation campaign focuses on the translation of TED talks, a collection of lectures on a variety of topics ranging from science to culture. For Arabic-to-English, the bilingual data consists of roughly 100K sentences of in-domain TED talks data and 8M sentences of "other"-domain (OD) United Nations (UN) data. For the German-to-English task, the data consists of 130K TED sentences and 2.1M sentences of "other"-domain data assembled from the news-commentary and the europarl corpora. For language model training purposes, we use an additional 1.4 billion words (supplied as part of the campaign monolingual training data).

		de	**en**	**ar**	**en**
IN	sen	130K		90K	
	tok	2.5M	3.4M	1.6M	1.7M
	voc	71K	49K	56K	34K
OD	sen	2.1M		7.9M	
	tok	55M	56M	228M	226M
	voc	191K	129K	449K	411K
dev	sen	883		934	
	tok	20K	21K	19K	20K
	oov	215 (1.1%)		184 (1.0%)	
test10	sen	1565		1664	
	tok	31K	27K	31K	32K
	oov	227 (0.7%)		228 (0.8%)	
test11	sen	1436		1450	
	tok	27K	27K	27K	27K
	oov	271 (1.0%)		163 (0.6%)	

Table 1: IWSLT 2011 TED bilingual corpora statistics: the number of sentences (sen), running words (tok) and vocabulary (voc) are given for the training data. For the test data, the number of out-of-vocabulary (oov) words relatively to using all training data (concatenating IN and OD) is given (in parentheses is the percentage).

The bilingual training and test data for the Arabic-to-English and German-to-English tasks are summarized in Table 1[1]. The English data is tokenized and lowercased while the Arabic data was tokenized and segmented using MADA v3.1 (Roth et al., 2008) with the ATB scheme (this scheme splits all clitics except the definite article and normalizes the Arabic characters alef and yaa). The German source is decompounded and part-of-speech-based long-range verb reordering rules (Popović and Ney, 2006) are applied.

From Table 1, we note that the general data is more than 20 times bigger than the in-domain data. A simple concatenation of the corpora might mask the phrase probabilities obtained from the in-domain corpus, causing a deterioration in performance. This is especially true for the Arabic-to-

[1] For a list of the IWSLT TED 2011 training corpora, see `http://www.iwslt2011.org/doku.php?id=06_evaluation`

English setup, where the UN data is 100 times bigger than the TED data and the domains are distinct. One way to avoid this contamination is by filtering the general corpus, but this discards phrase translations completely from the phrase model. A more principled way is by weighting the sentences of the corpora differently, such that sentences which are more related to the domain will have higher weights and therefore have a stronger impact on the phrase probabilities.

5.2 Translation System

The baseline system is built using the open-source SMT toolkit Jane[2], which provides state-of-the-art phrase-based SMT system (Wuebker et al., 2012). In addition to the phrase based decoder, Jane includes an implementation of the forced alignment procedure used in this work for the purpose of adaptation. We use the standard set of models with phrase translation probabilities and word lexical smoothing for source-to-target and target-to-source directions, a word and phrase penalty, distance-based reordering and an n-gram target language model. In addition, our baseline includes binary count features which fire if the count of the phrase pair in the training corpus is smaller than a threshold. We use three count features with thresholds $\{2, 3, 4\}$.

The SMT systems are tuned on the *dev* (dev2010) development set with minimum error rate training (Och, 2003) using BLEU (Papineni et al., 2002) accuracy measure as the optimization criterion. We test the performance of our system on the *test2010* and *test2011* sets using the BLEU and translation edit rate (TER) (Snover et al., 2006) measures. We use TER as an additional measure to verify the consistency of our improvements and avoid over-tuning. The Arabic-English results are case sensitive while the German-English results are case insensitive. In addition to the raw automatic results, we perform significance testing over all evaluations sets. For both BLEU and TER, we perform bootstrap resampling with bounds estimation as described by (Koehn, 2004). We use the 90% and 95% (denoted by † and ‡ correspondingly in the tables) confidence thresholds to draw significance conclusions.

6 Results

In this section we compare the suggested weighting schemes experimentally using the final translation quality. We use two TED tasks, German-to-English and Arabic-to-English translation. In addition to evaluating our suggested translation models based weighting schemes, we evaluate methods suggested in previous work, including LM based weighting and FA based adaptation.

The results for both German-to-English and Arabic-to-English TED tasks are summarized in Table 2. Each language pair section is divided into three subsections which differ by the phrase table training method. The first subsection is using state-of-the-art heuristic phrase extraction, the second is using FA adaptation and the third is using weighted phrase extraction with different weighting schemes.

To perform weighted phrase extraction, we use all data (*ALL*, a concatenation of *IN* and *OD*) as the general-domain data (in eq. 3 and Figure 1). This way, we ensure weighting for all sentences in the training data, and, data from *IN* is still used for the generation of the weighted phrase table.

6.1 German-to-English

Focusing on the German-to-English translation results, we note that using all data (ALL system) for the heuristic phrase extraction improves over the in-domain system (IN), with gains up-to +0.9% BLEU and -0.7% TER on the test2011 set. We perform significance testing in comparison to the ALL system as this is the best baseline system (among IN and ALL).

Mansour and Ney (2013) method of adaptation using the FA procedure (ALL-FA-IN) consistently outperforms the baseline system, with significant improvements on test10 TER.

Comparing the weighting schemes, weighting based on the phrase model (PM) and language model (LM) perform similarly, without a clear advantage to one method. The standalone weighting schemes do not achieve improvements over the baseline. Combining all the translation models (PM,SM,RM,CM,LP) into the TM scheme generates improvements over the standalone weighting schemes. TM also improves over the LM scheme suggested in previous work. We hypothesize that TM scoring is better for phrase model adaptation as it captures bilingual dependencies, unlike the LM scheme. In an experiment we do not report

[2] www.hltpr.rwth-aachen.de/jane

System	dev		test2010		test2011	
	BLEU	TER	BLEU	TER	BLEU	TER
German-to-English						
IN	31.0	48.9	29.3	51.0	32.7	46.8
ALL	31.2	48.3	29.5	50.5	33.6	46.1
Forced alignment based adaptation						
ALL-FA-IN	31.8	47.4†	29.7	49.7†	33.6	45.5
Weighted phrase extraction						
LM	31.1	48.7	29.2	51.1	33.6	46.2
PM	31.5	48.8	29.2	50.9	33.1	46.4
TM	31.7	48.4	29.8	50.2	33.8	45.8
TM+LM	32.2†	47.5†	30.1	49.5‡	34.4†	44.8‡
Arabic-to-English						
IN	27.2	54.1	25.3	57.1	24.3	59.9
ALL	27.1	54.8	24.4	58.6	23.8	61.1
ALL-FA-IN	27.7	53.7	25.3	56.9	24.7	59.3
LM	28.1†	52.9‡	26.0	56.2†	24.6	59.3
PM	27.2	54.4	25.1	57.5	24.1	60.3
TM	27.4	53.9	25.4	57.0	24.4	59.5
TM+LM	28.3‡	52.8‡	26.2†	55.9‡	25.1†	58.7‡

Table 2: TED 2011 translation results. BLEU and TER are given in percentages. *IN* denotes the TED lectures in-domain corpus and *ALL* is using all available bilingual data (including *IN*). Significance is marked with † for 90% confidence and ‡ for 95% confidence, and is measured over the best heuristic system.

here, we tried to remove one translation model at a time from the TM scheme, the results always got worse. Therefore, we conclude that using all translation models is important to achieve robust weighting and generate the best results.

Combining TM with LM weighting (TM+LM) generates the best system overall. Significant improvements at the 95% level are observed for TER, BLEU is significantly improved for test11. TM+LM is significantly better than LM weighting on both test sets. In comparison to ALL-FA-IN, TM+LM is significantly better on test11 BLEU. TM+LM combines the advantages of both scoring methods, where TM ensures in-domain lexical choice while LM achieves better sentence fluency.

6.2 Arabic-to-English

To verify our results, we repeat the experiments on the Arabic-to-English TED task. The scenario is different here as using the OD data (UN) deteriorates the results of the IN system by 0.9% and 0.5% BLEU on test2010 and test2011 correspondingly. We attribute this deterioration to the large size of the UN data (a factor of 100 bigger than IN) which causes bias to OD. In addition, UN is more distinct from the TED lecture domain. We use the IN system as baseline and perform significance testing in comparison to this system.

FA adaptation (ALL-FA-IN) results are similar to the German-to-English section, with consistent improvements over the baseline but no significance is observed in this case.

For the weighting experiments, combining the translation models into the TM scheme improves over the standalone schemes. The LM scheme is performing better than TM in this case. We hypothesize that this is due to the big gap between the in-domain TED corpus and the other-domain UN corpus. The LM scheme is combining a term which overweights sentences further from the other-domain. This factor proves to be crucial in the case of a big gap between IN and OD. Such a term is not present in the translation model weighting schemes, we leave its incorporation for future work.

Finally, similar to the German-to-English results, the combined TM+LM achieves the best results, with significant improvements at the 90% level for all sets and error measures, and at the

Type	**DE-EN**		**AR-EN**	
	base	TM+LM	base	TM+LM
lexical	23695	23451	26679	25813
reorder	1193	1106	935	904

Table 3: Error analysis. A comparison of the error types along with the error counts are given. The systems include the baseline system and the TM+LM weighted system.

Sample sentences	
src	es fuehlt sich grossartig an .
ref	it feels great .
base	it feels like a lot .
TM+LM	it feels great .
src	es haelt dich frisch .
ref	it keeps you fresh .
base	it’s got you fresh .
TM+LM	it keeps you fresh .
src	كيف ستقوم بإطعام العالم
ref	How are you going to feed the world
base	How will feed the world
TM+LM	How are you going to feed the world
src	ولماذا ؟ غذاء معدل جينيا
ref	And why? Genetically engineered food
base	And why ? Food rate genetically
TM+LM	And why ? Genetically modified food

Table 4: Sample sentences. The source, reference, baseline hypothesis and TM+LM weighted system hypothesis are given.

95% level for most. TM+LM improves over the baseline with +1.1% BLEU and -1.3% TER on dev, +0.9% BLEU and -1.2% TER on test2010 and +0.8% BLEU and -1.2% TER on test2011.

7 Error Analysis

In this section, we perform automatic and manual error analysis. For the automatic part, we use addicter[3] (Berka et al., 2012), which performs HMM word alignment between the reference and the hypothesis and measures lexical (word insertions, deletions and substitutions) and reordering errors. Addicter is a good tool to measure tendencies in the errors, but the number of errors might be misleading due to alignment errors. The summary of the errors is given in Table 3. From the table we clearly see that the majority of the improvement comes from lexical errors reduction. This is an indication of an improved lexical choice, due to the improved phrase model probabilities.

Translation examples are given in Table 4. The examples show that the lexical choice is being improved when using the weighted TM+LM phrase extraction. For the first example in German, “grossartig” means “great”, but translated by the baseline as “a lot”, which causes the meaning to be distorted. For the second Arabic example, the word معدل is ambiguous and could mean both “rate” and “modified”. The TM+LM system does the correct lexical choice in this case.

8 Conclusion

In this work, we investigate several weighting schemes for phrase extraction adaptation. Unlike previous work where language model scoring is used for adaptation, we utilize several translation models to perform the weighing.

The translation models used for weighting are calculated over phrase aligned general-domain sentences using an in-domain phrase table.

Experiments on two language pairs show significant improvements over the baseline, with gains up-to +1.0% BLEU and -1.3% TER when using a combined TM and LM (TM+LM) weighting scheme. The TM+LM scheme also shows improvements over previous work, namely scoring using LM and using FA training to adapt a general-domain phrase table to the in-domain (ALL-FA-IN method).

In future work, we plan to investigate using translation model scoring in a fashion similar to the cross entropy difference framework. In this case, the general-domain data will be phrase aligned and scored using a general-domain phrase table, and the difference between the in-domain based scores and the general-domain ones can be calculated. Another interesting scenario we are planning to tackle is when only monolingual in-domain data exists, and whether our methods could be still applied and gain improvements, for example using automatic translations.

Acknowledgments

This material is based upon work supported by the DARPA BOLT project under Contract No. HR0011-12-C-0015.

[3] https://wiki.ufal.ms.mff.cuni.cz/user:zeman:addicter

References

Axelrod, Amittai, Xiaodong He, and Jianfeng Gao. 2011. Domain adaptation via pseudo in-domain data selection. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 355–362, Edinburgh, Scotland, UK., July. Association for Computational Linguistics.

Berka, Jan, Ondrej Bojar, Mark Fishel, Maja Popovic, and Daniel Zeman. 2012. Automatic MT Error Analysis: Hjerson Helping Addicter. In *LREC*, pages 2158–2163, Istanbul, Turkey.

Cettolo, M Federico M, L Bentivogli, M Paul, and S Stüker. 2012. Overview of the iwslt 2012 evaluation campaign. In *International Workshop on Spoken Language Translation*, pages 12–33, Hong Kong, December.

Foster, George and Roland Kuhn. 2007. Mixture-model adaptation for SMT. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 128–135, Prague, Czech Republic, June. Association for Computational Linguistics.

Foster, George, Cyril Goutte, and Roland Kuhn. 2010. Discriminative instance weighting for domain adaptation in statistical machine translation. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 451–459, Cambridge, MA, October. Association for Computational Linguistics.

Goto, Isao, Bin Lu, Ka Po Chow, Eiichiro Sumita, and Benjamin K Tsou. 2013. Overview of the patent machine translation task at the ntcir-10 workshop. In *Proceedings of the 10th NTCIR Conference*, volume 10, pages 260–286, Tokyo, Japan, June.

Koehn, Philipp. 2004. Statistical Significance Tests for Machine Translation Evaluation. In *Proc. of the Conf. on Empirical Methods for Natural Language Processing (EMNLP)*, pages 388–395, Barcelona, Spain, July.

Mansour, Saab and Hermann Ney. 2012. A simple and effective weighted phrase extraction for machine translation adaptation. In *International Workshop on Spoken Language Translation*, pages 193–200, Hong Kong, December.

Mansour, Saab and Hermann Ney. 2013. Phrase training based adaptation for statistical machine translation. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 649–654, Atlanta, Georgia, June. Association for Computational Linguistics.

Matsoukas, Spyros, Antti-Veikko I. Rosti, and Bing Zhang. 2009. Discriminative corpus weight estimation for machine translation. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 708–717, Singapore, August. Association for Computational Linguistics.

Moore, Robert C. and William Lewis. 2010. Intelligent selection of language model training data. In *Proceedings of the ACL 2010 Conference Short Papers*, pages 220–224, Uppsala, Sweden, July. Association for Computational Linguistics.

Och, Franz J. 2003. Minimum Error Rate Training in Statistical Machine Translation. In *Proceedings of the 41th Annual Meeting of the Association for Computational Linguistics*, pages 160–167, Sapporo, Japan, July.

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA, July.

Popović, M. and H. Ney. 2006. POS-based Word Reorderings for Statistical Machine Translation. In *International Conference on Language Resources and Evaluation*, pages 1278–1283.

Roth, Ryan, Owen Rambow, Nizar Habash, Mona Diab, and Cynthia Rudin. 2008. Arabic morphological tagging, diacritization, and lemmatization using lexeme models and feature ranking. In *Proceedings of ACL-08: HLT, Short Papers*, pages 117–120, Columbus, Ohio, June. Association for Computational Linguistics.

Snover, Matthew, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A Study of Translation Edit Rate with Targeted Human Annotation. In *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas*, pages 223–231, Cambridge, Massachusetts, USA, August.

Wuebker, Joern and Hermann Ney. 2013. Length-incremental phrase training for smt. In *ACL 2013 Eighth Workshop on Statistical Machine Translation*, pages 309–319, Sofia, Bulgaria, August.

Wuebker, Joern, Arne Mauser, and Hermann Ney. 2010. Training phrase translation models with leaving-one-out. In *Proceedings of the 48th Annual Meeting of the Assoc. for Computational Linguistics*, pages 475–484, Uppsala, Sweden, July.

Wuebker, Joern, Matthias Huck, Stephan Peitz, Malte Nuhn, Markus Freitag, Jan-Thorsten Peter, Saab Mansour, and Hermann Ney. 2012. Jane 2: Open source phrase-based and hierarchical statistical machine translation. In *International Conference on Computational Linguistics*, Mumbai, India, December.

Data Selection for Discriminative Training in Statistical Machine Translation

Xingyi Song and **Lucia Specia**
Department of Computer Science
University of Sheffield
S1 4DP, UK
{xsong2,l.specia}@sheffield.ac.uk

Trevor Cohn
Computing and Information Systems
The University of Melbourne
VIC 3010, Australia
t.cohn@unimelb.edu.au

Abstract

The efficacy of discriminative training in Statistical Machine Translation is heavily dependent on the quality of the development corpus used, and on its similarity to the test set. This paper introduces a novel development corpus selection algorithm – the LA selection algorithm. It focuses on the selection of development corpora to achieve better translation quality on unseen test data and to make training more stable across different runs, particularly when hand-crafted development sets are not available, and for selection from noisy and potentially non-parallel, large scale web crawled data. LA does not require knowledge of the test set, nor the decoding of the candidate pool before the selection. In our experiments, development corpora selected by LA lead to improvements of over 2.5 BLEU points when compared to random development data selection from the same larger datasets.

1 Introduction

Discriminative training – also referred to as *tuning* – is an important step in log-linear model in Statistical Machine Translation (SMT) (Och and Ney, 2002). The efficacy of training is closely related to the quality of training samples in the development corpus, and to a certain extent, to the proximity between this corpus and the test set(s). Hui et al. (2010) in their experiments show that by using different development corpora to train the same SMT system, translation performance can vary up to 2.5 BLEU points (Papineni et al., 2002) with a standard phrase-based system (Koehn et al., 2007). How to build a 'suitable' development corpus is a important problem in SMT discriminative training.

A suitable development corpus should aid discriminative training achieve higher quality models, and thus yield better translations. Previous research on selecting training samples for the development corpus can be grouped into two categories: i) selecting samples based on the test set (transductive learning), or ii) selecting samples without knowing the test set (inductive learning). Research in the first category focuses on how to find similar samples to the ones the system will be tested on. Li et al. (2010), Lu et al. (2008), Zheng et al. (2010), and Tamchyna et al. (2012) measure similarity based on information retrieval methods, while Zhao et al. (2011) selects similar sentences based on edit distance. These similarity based approaches have been successfully applied to the local discriminative algorithm proposed in (Liu et al., 2012). The limitation of these approaches is that the test set needs to be known before model building, which is rarely true in practice.

Our research belongs to the second category. Previous work on development data selection for unknown test sets include Hui et al. (2010). They suggest that training samples with high oracle BLEU scores[1] will lead to better training quality. Cao and Khudanpur (2012) confirmed this and further showed that better training data will offer high variance in terms of BLEU scores and feature vector values between oracle and non-oracle hypotheses, since these are more easily separable by

[1] Oracle BLEU scores are those computed for the closest candidate translation to the reference in the n-best list of the development set.

the machine learning algorithms used for tuning. Both of the above studies achieved positive results, but these approaches require decoding the candidate development data to obtain BLEU scores and feature values, which may be difficult apply if the pool for data selection is extremely large.

Another potential way of improving training quality based on a development corpus is to increase the size of this corpus. However, high-quality sentence aligned parallel corpora are expensive to obtain. In contrast to data used for rule extraction in SMT, data used for SMT discriminative training is required to be of better quality for reliable training. Development data is therefore often created by professional translators. In addition, increasing the corpus size also increases the computational cost and the time required to train a model. Therefore, finding out how much data is enough to build a suitable development corpus is also an important question. Web crawled or crowdsourcing data are much cheaper than professionally translated data, and research towards exploiting such type of data (Zaidan and Callison-Burch, 2011; Uszkoreit et al., 2010; Smith et al., 2010; Resnik and Smith, 2003; Munteanu and Marcu, 2005) has already been successfully applied to machine translation, both in phrase extraction and discriminative training. However, they do not provide a direct comparison between their selected data and professionally built development corpora.

In order to address these problems, in this paper we introduce a novel development corpus selection algorithm, the **LA Selection** algorithm. It combines sentence length, bilingual alignment and other textual clues, as well as data diversity for sample sentence selection. It does not rely on knowledge of the test sets, nor on the decoding of the candidate sentences. Our results show that the proposed selection algorithm achieves improvements of over 2.5 BLEU points compared to random selection. We also present experiments with development corpora for various datasets to shed some light on aspects that might have an impact on translation quality, namely showing a substantial effect of the sentence length in the development corpus, and that with the right selection process large development corpora offer little benefits over smaller ones.

The remainder of this paper is structured as follows: We will describe our novel LA selection algorithm in Section 2. Experimental settings and results are presented in Sections 3 and 4, respectively, where we also discuss the training quality and scalability over different corpus size.

Algorithm 1 Development Data Selection

```
Require: Data Pool $D = (f^t, r^t, a^t)_{t=1}^{T}$, Number
    of words $N$, length limits $\lambda_{low}$ and $\lambda_{top}$
 1: Select = [], Cand = [], $L = 0$
 2: for $d_i = (f^i, r^i, a^i)$ in $D$ do
 3:   if $\lambda_{low} < \mathrm{length}(f^i) < \lambda_{top}$ then
 4:     Calculate feature score
          $s^i = \mathrm{score}(f^i, r^i, a^i)$
 5:     Add $(s^i, d^i)$ to Cand
 6:   end if
 7: end for
 8: Sort Cand by score from high to low
 9: while Selected length $L < N$ do
10:   for $d^i$ in Cand do
11:     if $\mathrm{maxSim}(f^i, \mathrm{Select}[f^j]_{j=J-200}^{J}) < 0.3$
        and $\mathrm{sim}(f^i, r^i) < 0.6$ then
12:       Add $(f^i, r^i)$ to Select
13:       $L = L + \mathrm{length}(f^i)$
14:     end if
15:   end for
16: end while
17: return $Selected$
```

2 Development Corpus Selection Algorithm

The proposed development corpus selection algorithm has two main steps: (i) selecting training sentence pairs by sentence **L**ength, and (ii) selecting training sentence pairs by **A**lignment and other textual clues. We call it **LA selection**. It also has an further step to reward diversity in the set of selected sentences in terms of the words they contain. The assumption of the LA algorithm is that a good training sample should have a "reasonable" length, be paired with a good quality translation, as mostly indicated by the word alignment clues between the candidate pair, and add to the existing set in terms of diversity.

LA selection is shown in Algorithm 1. Assume that we have T sentence pairs in our data set D. Each sentence pair d_i in D contains a foreign sentence f^i, a translation of the foreign sentence r^i and the word alignment between them a^i. We first filter out sentence pairs below the low length threshold λ_{low} and above the high length threshold λ_{top} (Line 3). Sentence length has a major im-

+/-	**Alignment Features**
+	Source/Target alignment ratio
-	Source/Target top three fertilities ratio
+	Source/Target largest contiguous span ratio
-	Source/Target largest discontiguous span
	Text only Features
+	Source and target length ratio
-	Target function word penalty

Table 1: Features used to score candidate sentence pairs.

pact on word alignment quality, which constitute the basis for the set of features we use in the next step. Shorter sentences tend to be easier to align than longer sentences and therefore our algorithm would naturally be biased to selecting shorter sentences. However, as we show later in our experiments, sentences that are either too short or too long often harm model accuracy. Therefore, is important to set both bottom and top limits on sentence length. Based on empirical results, we suggest set $\lambda_{low} = 10$ $\lambda_{top} = 50$, as we will further discuss in Section 4.1.

After filtering out sentences by the length thresholds, the next step is to extract the feature values for each remaining candidate sentence pair. The features used in this paper are listed in Table 1. The first column of the Table is an indicator of the sign of the feature value, where a negative sign indicates that the feature will return a negative value, and positive sign indicates that the feature will return a positive value. The actual features, which we describe below, are given in the second column. These include word alignment features, which are computed based on GIZA++ alignments for the candidate development set, and simpler textual features. The alignment features used here are mostly adapted from (Munteanu and Marcu, 2005).

The **alignment ratio** is the ratio between the number of aligned words and length of the sentence in words:

$$\text{Alignment Ratio} = \frac{\text{No. Aligned Words}}{\text{Sentence Length}}$$

A low alignment ratio means that the data is most likely non-parallel, or else a highly non-literal translation. Either way, these are likely to prove detrimental.

Word fertility is the number of foreign words aligned to each target word. The **word fertility ratio** is the ratio between word fertility and sentence length. We use the top three largest fertility ratio as three features:

$$\text{Fertility Ratio} = -\frac{\text{Word fertility}}{\text{Sentence Length}}$$

This feature can detect garbage collection, where the aligner uses a rare word to erroneously account for many difficult words in the parallel sentence.

Our definition of **contiguous span** differs from that in (Munteanu and Marcu, 2005): we define it as a substring in which all words have an alignment to words in the other language. A **discontiguous span** is defined as a substring in which all words have no alignment to any word in the other language. The **contiguous span ratio**, CSR, is the length of the largest contiguous span over the length of the sentence:

$$CSR = \frac{LC}{\text{Sentence Length}}$$

The **discontiguous span ratio**, $DCSR$, is the length of the largest discontiguous span over the length of the sentence:

$$DCSR = -\frac{LDC}{\text{Sentence Length}}$$

where LC is the length of the contiguous span and LDC is the length of the discontiguous span.

In addition to the word alignment features, we use **source and target length ratio**, LR, to measure how close the source and target sentences in the pair are in terms of length:

$$LR = \begin{cases} \frac{TL}{SL} & \text{if } SL > TL \\ \frac{SL}{TL} & \text{if } TL > SL \end{cases}$$

where TL is target sentence length and SL is source sentence length.

Finally, the **target function words penalty**, FP, penalises sentences with a large proportion of function words or punctuation:

$$FP = -\exp\left(-\frac{n_{\text{func}}}{TL}\right)$$

where n_{func} is number of function words and punctuation symbols, and TL is the target sentence length. We only consider a target language penalty, but a source language penalty could also be used.

Once we obtained these feature values for all candidate sentence pairs, we apply two approaches to calculate an overall score for the candidate. The first is a heuristic approach, which simply sums over the scores of all features for each sentence (with some features negated as shown in Table 1). The second approach uses machine learning to combine these features, similar to what was done in (Munteanu and Marcu, 2005) to distinguish between parallel and non-parallel sentences. Here a binary SVM classifier is trained to predict samples that are more similar to professionally created sentences. The labelling of the data was therefore done by contrasting professionally created translations against badly aligned translations from web crawled data. The heuristic approach achieved better performance than the machine learning approach, as we will discuss in Section 4.2.

Lines 8 through 16 in Algorithm 1 describe the sentence pair selection procedure based on this overall feature score. The candidate sentence pair and its features are stored in the Cand list, and sorted from high to low according to their overall feature scores. The algorithm takes candidate sentence pairs from the Cand list until the number of words in the selected training corpus Select reaches the limit N. If the candidate sentence pair passes the condition in Line 11, the sentence pair is added to the selected corpus Select.

Line 11 has two purposes: first, it aims at increasing the diversity of the selected training corpus. Based on our experiments, candidate sentence pairs with similar feature scores (and thus similar rankings) may be very similar sentences, with most of their words being identical. We therefore only select a sentence pair whose source sentence has less than 0.3 BLEU similarity as compared to the source sentences in last 200 selected sentence pairs.[2] The second purpose is to filter out sentence pairs that are not translated, i.e., sentence pairs with same words in the source and target sides. Untranslated or partially untranslated sentence pairs are common in web crawled data. We therefore filter out the sentence pairs whose source and target have a BLEU similarity score of over 0.6.

3 Experimental Settings

SMT system: We build standard phrase-based SMT systems for each corpus using Moses with its 14 default features. The word alignment and language models were learned using GIZA++ and IRSTLM with Moses default settings. A trigram language model was trained on English side of the parallel data. For discriminative training we use the popular MERT (Och, 2003) algorithm.

Two language pairs are used in the experiments, French to English and Chinese to English, with the following corpora:

French-English Corpora: To build a French to English system we used the Common Crawl corpus (Smith et al., 2013). We filtered out sentence with length over 80 words and split the corpus into training (Common Crawl training) and tuning (Common Crawl tuning). The **training** subset was used for phrase table, language model and reordering table training. It contains $3,158,523$ sentence pairs (over 161M words) and average source sentence length of 27 words. The **tuning** subset is used as "Noisy Data Pool" to test our LA selection algorithm. It contains $31,929$ sentence pairs (over 1.6M words), and average source sentence length of 27 words. We compare the performance of our selected corpora against a concatenation of four professionally created development corpora (Professional Data Pool) for the news test sets distributed as part of the WMT evaluation (Callison-Burch et al., 2008; Callison-Burch et al., 2009; Callison-Burch et al., 2010): 'newssyscomb2009', 'news-test2008', 'newstest2009' and 'newstest2010'. Altogether, they contain $7,518$ sentence pairs (over 392K words) with average source sentence length of 27 words. As **test data**, we take the WMT13 (average source sentence length = 24 words) and WMT14 (average source sentence length = 27 words) news test sets.

Chinese-English Corpora: To build the Chinese to English translation system we use the non-UN and non-HK Hansards portions of the FBIS (LDC2003E14) training corpus ($1,624,512$ sentence pairs, over 83M words, average source sentence = 24) and **tuning** ($33,154$ sentence pairs, over 1.7M words, average sentence length = 24). The professionally created development corpus in this case is the NIST MT06 test set[3] ($1,664$ sentence pairs, 86K words, average sentence length = 23 words). As **test data**, we use the NIST

[2] The 200 sentence pair limit is used to reduce the runtime on large datasets.

[3] It contains 4 references, but we only apply the first reference to make it comparable to our selection algorithm.

MT08 test set (average source sentence length = 24 words).

Note that for both language pairs, the test sets and professionally created development corpora belong to the same domain: news, for both French-English and Chinese-English. In addition, the test and development corpora for each language pair have been created in the same fashion, following the same guidelines. Our pool of noisy data, however, includes not only a multitude of domains different from news, but also translations created in various ways and noisy data.

4 Results

Our experiments are split in three parts: Section 4.1 examines how sentence length in development corpora affects the training quality. Section 4.2 compares our LA selection algorithm against randomly selected corpora and against professionally created corpora. Section 4.3 discusses the effect of development corpus size by testing translation performance with corpora of different sizes.

4.1 Selection by Sentence Length

In order to test how sentence length affects the quality of discriminative training, we split the tuning corpus into six parts according to source sentence length ranges (in words): [1-10], [10-20], [20-30], [30-40], [40-50] and [50-60]. For each range, we randomly select sentences to total $30,000$ words as a small training set, train a discriminative model based on the small training set, and test the translation performance on WMT13 and NIST MT08 test sets. We repeat the random selection and training procedure five times and report average BLEU scores in Table 2.

The top half of Table 2 shows the results for French-English translation. From this Table, we can see that corpora with sentence lengths of [30-40] and [30-50] lead to better translation quality than random selection, with a maximum average BLEU score of 25.62 for sentence length [30-40], outperforming random length selection by 1.26 BLEU points. Corpora with sentences in [10-20] and [20-30] perform slightly worse than random selection. The worst performance is obtained for corpora with very short or very long sentences.

The lower half of Table 2 shows the results for Chinese-English translation. Lengths [10-20], [20-30], [30-40] and [40-50] lead to better translation performance than random selection. As for French-English translation, the worst performance is obtained for corpora with very short or very long sentences, with a lower BLEU score than random selection.

According to above results, the best sentence length for discriminative training is not fixed, as it may depend on language pairs and corpus type. However, sentences below 10 words or above 50 words lead to poor results for both language pairs. We conduct another experiment selecting development corpora excluding sentences with length below 10 or above 50. Results are shown in column [10-50] of both Tables. Compared to random selection, [10-50] improved BLEU scores by 1.18 for French-English, and by 0.54 for Chinese-English. Note that our systems were developed on corpora with average sentence length of around 25 words, which is typical in most freely available training corpora,[4] the thresholds may differ for corpora with very different sentence lengths.

4.2 Selection by LA Algorithm

In what follows we compare the performance of our LA selection algorithm against randomly selected and professionally created corpora. We set $\lambda_{low} = 10$ and $\lambda_{top} = 50$ and select a development corpus with no more than $30,000$ words. Results are reported in Table 3, again with averages over five runs.

Considering first the results for the French-English WMT13 test set, the LA selection improves BLEU by 1.36 points compared with random selection, and also improves over sentence length-based selection (10-50). The performance of the LA selected corpus is only slightly lower (0.1 BLEU) than that of the professionally created corpus (Prof.), but the system is much more robust with much lower standard deviation (std). This is a surprising outcome as the professionally created development sets are drawn from the same domain as the test sets (news), and were created using the same translation guidelines as the test set, and therefore better results were expected for these corpora. We have similar findings for the French-English WMT14 and Chinese-English MT08 test sets. Systems trained on corpora selected by LA increase 1.21 and 2.53 BLEU points over random selection, respectively. For the WMT14 test set, the corpus selected by LA show slight im-

[4] For example, both Europarl and News-Commentary WMT corpora have an average of 25 words on their English side.

		Rand.	1-10	10-20	20-30	30-40	40-50	50-60	10-50
WMT13	**avg.**	24.36	22.85	23.61	24.43	25.62	24.62	22.94	25.54
	std.	0.84	0.65	0.80	0.51	0.40	1.06	0.99	0.84
MT08	**avg.**	18.79	18.11	20.00	19.63	18.85	19.29	18.53	19.33
	std.	0.83	0.29	1.45	1.00	0.85	1.38	0.81	1.16

Table 2: Average BLEU scores and standard deviation on French to English (WMT13) and Chinese to English (MT08) test sets for different ranges of sentence length. The leftmost **Rand.** column has no length restrictions.

		Rand.	10-50	LA_{10-50}	Prof.
WMT13	**avg.**	24.36	25.54	25.72	25.82
	std.	0.84	0.84	0.01	0.23
WMT14	**avg.**	25.19	25.31	26.40	26.31
	std.	0.30	0.14	0.04	0.16
MT08	**avg.**	18.79	19.33	21.32	23.49
	std.	0.83	1.16	0.83	0.31

Table 3: Average BLEU scores and standard deviation for French-English (WMT13, WMT14) news test sets and Chinese-English (MT08) test set with development corpora selected by length (10-50), LA algorithm (LA_{10-50}), randomly (Rand.), or created by professionals (Prof.).

	WMT13	WMT14
avg.	25.42	26.08
std.	0.08	0.08

Table 4: Average BLEU scores and standard deviation for SVM-based LA selection on French-English WMT13 and WMT14 test sets.

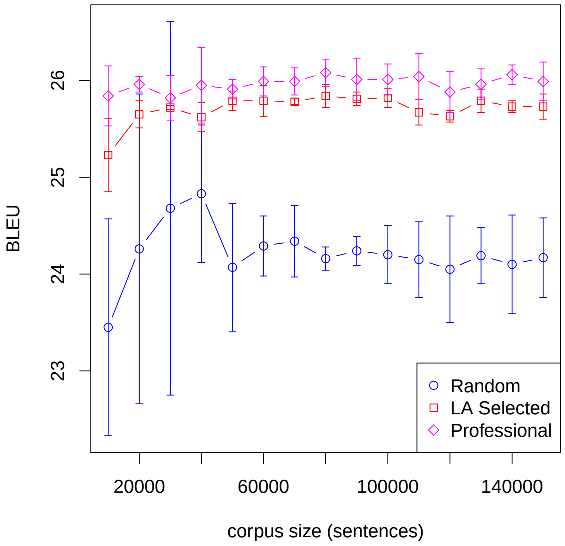


Figure 1: BLEU score changes for development corpora of different sizes with the French-English WMT13 corpus. The horizontal axis shows corpus size, and the vertical axis, BLEU scores. Points show the mean results and whiskers denote ± one standard deviation.

provements over the professionally created corpus (26.40 vs. 26.31) with a lower variance.

We also experiment with using the SVM classifier to combine features in the LA selection algorithm, as previously discussed. The classifier was trained using the SVMlight[5] toolkit with RBF kernel with its default parameter settings. We selected $30,000$ words from the professionally created WMT development corpus as positive training samples, and used as negative examples $30,000$ words from our corpus with the lowest LA selection score. Different from the LA selection method, here sentence length is not limited to 10-50, but rather the sentence length is provided as a feature to the classifier. The motivation was to test the ability of the algorithm in learning a suitable sentence length for tuning. Nevertheless, on average sentences have similar lengths: 16 for the corpus selected with the SVM classifier against 18 for the corpus selected with the heuristic method. Results for sentence selection using the highest classification scores are shown in Table 4.

LA selection with the SVM classifier outperforms random selection, but does worse than our heuristic approach (compare to LA_{10-50} in Table 3). The reason may be the quality of the training data: both our positive and negative training examples will contain considerable noise. The WMT professionally created corpora includes some odd translations, so the alignment features will be less reliable. Also, we stress that this is a harder problem than the one introduced in (Munteanu and Marcu, 2005), since their pool of candidate samples contained either parallel or non-parallel sentences, which are easier to label and to distinguish based on word alignment features. Our pool of candidate samples is assumed to be parallel, with our selection procedure aiming at selecting from this the highest quality translations.

[5] `http://svmlight.joachims.org/`

4.3 Effect of Training Corpus Size

Next, we consider the question of how much development data is needed to train a phrase-based SMT system. To test this we experiment with corpora ranging in size from $10,000$ words to $150,000$ words, with an incremental step of $10,000$ words. At each step we run MERT training five times and report the average BLEU scores. The test set is the WMT13.

Figure 1 shows how BLEU changes as we increase the training corpus size. The three lines represent the BLEU scores of three systems: Random selection from the French-English tuning dataset (blue line), LA selection from the same pool (red line), and WMT professionally created development corpus (green line). According to this Figure, performance increases as corpora sizes increase, for all techniques, but only up to $70,000$ words, after which performance is stable. The professionally created corpus achieves the best performance for any corpus size. Note however that the LA selection technique is only slightly worse, with less than 0.1 BLEU difference, for corpora sizes $\geq 30,000$ words. Random selection clearly performs poorly compared to both.

Also shown in Figure 1 are the standard deviation from five runs of the experiment. Random selection presents the largest standard deviation (greater than 0.6 BLEU) for training corpora of sizes below $50,000$ words. The maximum standard deviation is 1.93 at $30,000$ words. With larger training corpus sizes, the standard deviation of random selection is still higher than that of LA selected and professional data. LA selection has a much lower average standard deviation, even lower than the professionally created data. This is important for real application settings, where repeated runs are not practical and robust performance from a single run is imperative.

These results confirm some findings of previous research (Hui et al., 2010), namely that enlarging the tuning corpus leads to more accurate models. However we find that increasing the amount of data is not the best solution when creating a development corpus: much greater improvements are possible by instead focusing on selecting better quality data. Using data selection reduces the need for large development sets, in fact as few as 70k words is sufficient for robust tuning.

5 Conclusions

In this paper we have shown how the choice of the development corpus is critical for machine translation systems' performance. The standard practice of resorting to expensive human translations is not practical for many SMT application scenarios, and consequently making better use of existing parallel resources is paramount. Length is the most important single criterion for selecting effective sentences for discriminative training: overly short and overly long training sentences often harm training performance. Using large development sets brings only small improvements in accuracy, and a modest development set of 30k-70k words is sufficient for good performance. The key innovation in this paper was the LA sentence selection algorithm, which selects high quality and diverse sentence pair for translation. We have shown large improvements over random selection, of up to 2.53 BLEU points (Chinese-English). The approach is competitive with using manually translated development sets, despite having no knowledge of the test set, test set domain, nor using expensive expert translators. In future work, we plan to improve the classification technique for automatically predicting training quality through alternative methods for extracting training examples and additional features to distinguish between good and bad translations.

6 Acknowledgement

Dr. Specia has received funding from the European Union's Seventh Framework Programme for research, technological development and demonstration under grant agreement no. 296347 (QT-LaunchPad). Dr. Cohn is the recipient of an Australian Research Council Future Fellowship (project number FT130101105).

References

Callison-Burch, Chris, Cameron Fordyce, Philipp Koehn, Christof Monz, and Josh Schroeder. 2008. Further meta-evaluation of machine translation. In *Proceedings of the Third Workshop on Statistical Machine Translation*, pages 70–106, Columbus, Ohio, June. Association for Computational Linguistics.

Callison-Burch, Chris, Philipp Koehn, Christof Monz, and Josh Schroeder. 2009. Findings of the 2009 Workshop on Statistical Machine Translation. In *Proceedings of the Fourth Workshop on Statistical Machine Translation*, pages 1–28, Athens, Greece, March. Association for Computational Linguistics.

Callison-Burch, Chris, Philipp Koehn, Christof Monz, Kay Peterson, Mark Przybocki, and Omar Zaidan. 2010. Findings of the 2010 joint workshop on statistical machine translation and metrics for machine translation. In *Proceedings of the Joint Fifth Workshop on Statistical Machine Translation and MetricsMATR*, pages 17–53, Uppsala, Sweden, July. Association for Computational Linguistics. Revised August 2010.

Cao, Yuan and Sanjeev Khudanpur. 2012. Sample selection for large-scale mt discriminative training. In *AMTA*.

Hui, Cong, Hai Zhao, Yan Song, and Bao-Liang Lu. 2010. An empirical study on development set selection strategy for machine translation learning. In *Proceedings of the Joint Fifth Workshop on Statistical Machine Translation and MetricsMATR*, WMT '10, pages 67–71, Uppsala, Sweden. Association for Computational Linguistics.

Koehn, Philipp, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Bertoldi Nicola Federico, Marcello, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of ACL 2007, Demonstration Session*, Prague, Czech Republic.

Li, Mu, Yinggong Zhao, Dongdong Zhang, and Ming Zhou. 2010. Adaptive development data selection for log-linear model in statistical machine translation. In *Proceedings of the 23rd International Conference on Computational Linguistics*, COLING '10, pages 662–670, Beijing, China. Association for Computational Linguistics.

Liu, Lemao, Hailong Cao, Taro Watanabe, Tiejun Zhao, Mo Yu, and CongHui Zhu. 2012. Locally training the log-linear model for smt. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, EMNLP-CoNLL '12, pages 402–411, Jeju Island, Korea.

Lu, Yajuan, Jin Huang, and Qun Liu. 2008. Improving statistical machine translation performance by training data selection and optimization.

Munteanu, Dragos Stefan and Daniel Marcu. 2005. Improving machine translation performance by exploiting non-parallel corpora. *Comput. Linguist.*, 31(4):477–504, December.

Och, Franz Josef and Hermann Ney. 2002. Discriminative training and maximum entropy models for statistical machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 295–302, Philadelphia, Pennsylvania.

Och, Franz Josef. 2003. Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 1*, ACL '03, pages 160–167, Sapporo, Japan.

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 311–318, Philadelphia, Pennsylvania.

Resnik, Philip and Noah A. Smith. 2003. The web as a parallel corpus. *Comput. Linguist.*, 29(3):349–380, September.

Smith, Jason R., Chris Quirk, and Kristina Toutanova. 2010. Extracting parallel sentences from comparable corpora using document level alignment. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, HLT '10, pages 403–411, Los Angeles, California.

Smith, Jason R., Philipp Koehn, Herve Saint-Amand, Chris Callison-Burch, Magdalena Plamada, and Adam Lopez. 2013. Dirt cheap web-scale parallel text from the common crawl. In *Proceedings of the 2013 Conference of the Association for Computational Linguistics (ACL 2013)*.

Tamchyna, Aleš, Petra Galuščáková, Amir Kamran, Miloš Stanojević, and Ondřej Bojar. 2012. Selecting data for english-to-czech machine translation. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, WMT '12, pages 374–381, Montreal, Canada.

Uszkoreit, Jakob, Jay M. Ponte, Ashok C. Popat, and Moshe Dubiner. 2010. Large scale parallel document mining for machine translation. In *Proceedings of the 23rd International Conference on Computational Linguistics*, COLING '10, pages 1101–1109, Beijing, China.

Zaidan, Omar F. and Chris Callison-Burch. 2011. Crowdsourcing translation: professional quality from non-professionals. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 1220–1229, Portland, Oregon.

Zhao, Yinggong, Yangsheng Ji, Ning Xi, Shujian Huang, and Jiajun Chen. 2011. Language model weight adaptation based on cross-entropy for statistical machine translation. In *Proceedings of the 25th Pacific Asia Conference on Language, Information and Computation*, pages 20–30, Singapore, December. Institute of Digital Enhancement of Cognitive Processing, Waseda University.

Zheng, Zhongguang, Zhongjun He, Yao Meng, and Hao Yu. 2010. Domain adaptation for statistical machine translation in development corpus selection. In *Universal Communication Symposium (IUCS), 2010 4th International*.

TRANSLAIDE

The translaide.pl system: an effective real world installation of translation memory searching and EBMT

Rafał Jaworski
rjawor@amu.edu.pl
Adam Mickiewicz University in Poznań

Renata Ziemlińska
renata.ziemlinska@poleng.pl
PolEng Sp. z o.o.

http://translaide.pl

Description

translaide.pl is a CAT system developed by the Polish company PolEng Sp. z o.o. that supports multiple input and output languages. The main idea of the system is to enable the sharing of resources among translators. A demo version of the system is available on the internet (http://translaide.pl), yet it is primarily intended for exclusive use in a single corporation. The system has been successfully implemented in two companies dealing with high-volume content to be translated.

The system is fully operable and has hundreds of users. To be able to meet the requirements of so many users using the system simultaneously, *translaide.pl* uses state-of-the-art, well optimized, natural language processing algorithms. The main features of the system include:

- Specialized, narrow-domain statistical translators.
- Automated dictionary lookup.
- Automated concordance lookup.
- Multiple translation memories.

The main challenges encountered in the course of developing the translation memory module were efficiency and accuracy of TM suggestions. The solution was using a modern TM search algorithm to ensure high lookup speed and an EBMT mechanism to improve the quality of TM suggestions. As shown in experiments, the solution is robust and performs well even with large translation memories. Returned suggestions are useful and their resemblance scores reflect human intuition of sentence resemblance.

The focus of future work would be on implementing automatic statistical translators for narrow text domains. During experiments, English-to-Polish SMT translators for technical texts were created and tested. The quality of the translations was relatively high, leaving human translators little room for post-editing. With such promising results, more SMT translators will be developed, covering other text domains and language pairs.

Collaborative Project

EU-BRIDGE – Bridges Across the Language Divide

Part of the Seventh Framework Programme
ICT-2011.4.2 Language Technologies
Funded by the EC – DG CONNECT
Grant Number 287658

http://www.eu-bridge.eu

List of partners (March 2014)
Karlsruhe Institute of Technology, Germany (coordinator)
Fondazione Bruno Kessler, Italy
Polish Japanese Institute of Information and Technology, Poland
RWTH Aachen University, Germany
University of Edinburgh, United Kingdom
The Hong Kong University of Science and Technology, Hong Kong
Red Bee Media Ltd, United Kingdom
PerVoice S.p.A, Italy
Accipio Projects GmbH, Germany
Andrexen, France

Project duration: Feb 2012 – Jan 2015

Summary

EU-BRIDGE aims to develop speech and machine translation capabilities that exceed the state-of-the-art in new and more challenging use cases. EU-BRIDGE seeks to achieve rapid technology transition and market insertion by creating a cloud-based speech translation service infrastructure upon which four use cases are built:

Captioning Translation for TV Broadcasts: Language technology will improve the work progress in captioning and translating the captions into multiple languages at Red Bee Media. More media content can be subtitled and translated to serve more European citizens.

University Lecture Translation: Spoken content of university lectures is translated in real time, now routinely running in three university lecture halls at Karlsruhe Institute of Technology and serving international students.

Speech Translation Support within the European Parliament: A first prototypical application aims to support the interpreters in their preparation stage by extracting relevant terminology; a next prototype is to aid the interpreters during their work in a booth during a session by highlighting named entities.

Webinar Translation: A web-based transcription and speech translation service within Andrexen's unified communication platform will help non-native participants participating in a webinar even when their language skills require a little help.

European Commission
FP7
STREP
287688
http://www.matecat.com

List of partners
Fondazione Bruno Kessler, Italy (coordinator)
University of Edinburgh, UK
Universitè du Maine, France
Translated, Italy

Project duration: November 2011 — October 2014

Summary

The objective of MateCat is to improve the integration of machine translation and human translation within the so-called computer aided translation (CAT) framework. Several recent studies have shown that post-editing suggestions of a statistical MT engine can substantially improve productivity of professional translators. MateCat leverages the growing interest and expectations in statistical MT by advancing the state of the art along directions that will hopefully accelerate its adoption by the translation industry. In particular, MateCat investigates the integration of MT into the CAT workflow along three main research directions:

- **Self-tuning M** that adapts MT to specific domains or translation projects;
- **User adaptive MT** that quickly adapts from user corrections and feedback;
- **Informative MT** that supplies additional hints to enhance the user experience.

These new MT functionalities have been integrated in a new Web-based CAT tool, that was specifically developed by the industrial partner. At this time, the **MateCat Tool** provides an enterprise level workbench for professional translators, which integrates advances MT functions such as online adaptation and quality estimation scores. The tool is currently field-tested by hundreds of translators and is freely is available in beta version under the LGPL license.

PEDAL: Post-Editing with Dynamic Active Learning

Science Foundation Ireland
Technology Development Innovation Award (TIDA) Feasibility Study
Grant: 12/TIDA/I2438

List of partners	
DCU	Dublin City University, Ireland
CNGL CENTRE FOR GLOBAL INTELLIGENT CONTENT	CNGL Centre for Global Intelligent Content, Ireland

Project duration: July 2013 — June 2014

Summary

Machine translation, in particular statistical machine translation (SMT), is making big inroads into the localisation and translation industry. In typical workflows (S)MT output is checked and (where required) manually post-edited by human translators. Recently, a significant amount of research has concentrated on capturing human post-editing outputs as early as possible to incrementally update/modify SMT models to avoid repeat mistakes. Typically in these approaches, MT and post-edits happen sequentially and chronologically, following the way unseen data (the translation job) is presented. In this project, we add to the existing literature addressing the question whether, and if so, to what extent, this process can be improved upon by Active Learning, where input is not presented chronologically but dynamically selected according to criteria that maximise performance with respect to (whatever is) the remaining data. The criteria we use are novel and allow the **MT system to improve its performance earlier**. Because these criteria are computationally cheap and language independent, our technology, together with incremental retraining, can be **easily integrated into the industry workflows**.

CASMACAT: Cognitive Analysis and Statistical Methods for Advanced Computer Aided Translation

Philipp Koehn, University of Edinburgh, pkoehn@inf.ed.ac.uk
Michael Carl, Copenhagen Business School, mc.isv@cbs.dk
Francisco Casacuberta, Polytechnic University of Valencia, fcn@dsic.upv.es
Eva Marcos, Celer Solutions, eva.marcos@celersol.com
http://www.casmacat.eu/

Description

In its third year, the CASMACAT project has developed – in collaboration with the MATECAT project – a new open source workbench for translators that is deployed over the web and as a stand-alone tool. With insights from cognitive studies of translator behavior, new types of assistance have been developed and tested in field trials.

The cognitive studies of translator behavior with the CASMACAT workbench include

- identification of translator types and styles
- cognitive and user models of translation processes

The advances of the CASMACAT project include:

- interactive translation prediction with machine learning methods
- interactive translation prediction for syntax-based models
- active learning applied to translation tasks
- sentence and word level confidence measures
- synthesis of translation memories and machine translation
- word alignment visualization
- display of multiple translation options
- online learning (incremental updating of models)
- domain and user adaptation
- integration of e-pen as input device
- logging and replay mode
- integration of eye tracker for collection of user activity data
- visualization tools for logging data

Additional advances are currently under development:

- integration of paraphrasing and alternative translations on demand
- automatic reviewing

The tool comprising the CASMACAT workbench are currently integrated into a stand-alone version that runs on any computer.

ACCEPT : Automated Community Content Editing PorTal

European Commission
FP7-ICT-2011-7 - Language technologies
STREP
Grant agreement: 288769
http://www.accept-project.eu

List of partners
Université de Genève, Switzerland (coordinator)
University of Edinburgh, UK
Acrolinx GmbH, Germany
Lexcelera, France
Symantec Limited, Ireland

Project duration: January 2012 — December 2014

Summary

ACCEPT is a Collaborative Project – STREP aimed at developing new methods and techniques to make machine translation (MT) work better in the environment characterised by internet communities sharing specific information. Today, anyone can in principle create information and make it available to anyone in the world who has Internet access. Yet the language barrier remains: however accessible information is, it is still only available to those who speak the language it is written in. ACCEPT's mission is to help communities share information more effectively across the language barrier by improving the quality of machine-translated community content. The project proposes a new approach to help MT work better for community content, in order to ensure that the result is comprehensible and correct. The approach consists of the following main axes of research and development:

- Development of user-friendly (minimally intrusive) strategies for pre-editing content for statistical machine translation.
- Development of strategies for post-editing. Ideally, post-editing of the translation results is done by bilingual skilled experts, but the lack of such experts is a major bottleneck.
- Improvement of training methods and development of feedback loops to improve Statistical Machine Translation (SMT) for community data.
- Use of text analytics for SMT. The project will try to determine if knowledge of the content can help produce better translations (for instance, translations that preserve sentiment polarity).

Source Content Analysis and Training Data Selection Impact on an MT-driven Program Design

Olga Beregovaya, David Landan
<first.last>@welocalize.com
Welocalize, Inc.

Description

Clients requiring translation and localization services have come to require an ever-increasing volume of data to be processed, and an unprecedented diversity in the nature of the data to be translated. To meet the increasing demand for translation and the various requirements to the quality of the target output, nearly all language service providers (LSPs) have added machine translation (MT) and various levels of post editing (PE) as integral components of their service offerings.

It has been repeatedly shown that statistical MT engines trained on clean and relevant in-domain data lead to better quality of machine translation output, by using just one of the quality measurement metrics. The importance of corpus preparation and curation and matching the training corpus to the specifics of the content to be translated cannot be overstated. Because of the rapid growth of the amount of data that must be processed, it is imperative that LSPs replace human source content and training corpora evaluations, which are costly both in terms of time and money spent, with a range of programmatic methodologies, which allow for predicting the quality of machine-translated output when specific training data is used, selecting the most suitable translation and post-editing approach and assembling the right workforce for the task.

We employ a large and still-growing suite of tools (both proprietary and through joint academic partnerships) for selecting the best suited dataset matched to the source content to be translated, and estimating the quality of the machine-translated output and the subsequent post-editing effort. To that end, we present several ways that we are working towards automating training data selection and matching it with the source content using a suite of source content analysis tools including:

- Candidate Scorer – a proprietary tool; uses part of speech (POS) n-grams to identify hard-to-translate segments, using a pre-selected corpus that is known to give the worst results, based on human ranking of such segments and post-editing time and distance.
- Source Content Profiler (alpha) – an Industry Partnership CNGL project; uses several features to classify documents into profiles and flags challenges for both machine and human translation
- Perplexity Evaluator – a proprietary tool; generates a matrix of perplexity scores for candidate and control documents against various language models (LMs) built from pre-selected corpora for good and bad results and one custom LM built from historical client in-domain data
- TMTPrime – an Industry Partnership CNGL project; provides a mechanism for automating selection between multiple MT engines, based on source input, using in-domain training data.
- StyleScorer (alpha) – a proprietary tool; scores and ranks candidate source documents according to established style guidelines. In training document selection, StyleScorer learns from a monolingual client corpus that adheres to a desired style, then combines scores from several NLP-based algorithms to generate a final score between 0 and 4 (with 4 being best match to established style).

It has become evident to us that the tools originally created specifically for a single task of either target data selection or source content profiling are often beneficial for both tasks. We present details of the above tools in conjunction with case studies that highlight where each tool has led to improved MT output and/or reductions in post-editing effort. We also present support tools that, while not strictly related to content analysis and data selection, make the outcome of the aforementioned tools and processes easier to interpret.

TaaS – Terminology as a Service

EU Seventh Framework Programme (FP7)
ICT Call - SME initiative on Digital Content and Languages
Small or medium scale focused research project (STREP)
Project ID: 296312
http://www.taas-project.eu

List of partners
Tilde, Latvia (coordinator)
Cologne University of Applied Sciences, Germany
Kilgray, Hungary
University of Sheffield, United Kingdom
TAUS, Netherlands

Project duration: May 2012 — May 2014

Summary

The project implements a new paradigm in terminology work creating an online platform termunity.com to automate terminology identification, acquisition and processing tasks. The automation of individual tasks is provided as a set of interoperable cloud-based services integrated into workflows. These services automate identification of term candidates in user-provided documents, the lookup of translation equivalents in online terminology resources and on the Web by automatically extracting multilingual terminology from comparable and parallel online resources. Although term identification is very challenging even to human annotators, we can achieve a comparable precision with automatic methods using the extended term tagging system. For example, for Latvian an average precision of 53.8% was achieved in comparison to an average annotator agreement rate of 63.3%. Collaborative involvement of users contributes to refinement and enrichment of raw terminological data. An API is provided for usage of the terminology services and terminology data by external systems. This API-level integration is currently implemented by the memoQ CAT tool and the LetsMT statistical MT system. In the framework of the project several methods have been elaborated to use terminology data for customization and quality improvement of domain specific statistical machine translation. Training level integration includes enrichment of monolingual and parallel data with terminology, adaptation of translation model by adapting and filtering phrase table, and adaptation of language model. Translation level integration is provided by pre-processing the source text to identify terms and mark-up them with translation hypotheses. Evaluation results show that combination of these methods significantly improves translation quality.

Oral Session 1
Research Papers

Alignment Symmetrization Optimization
Targeting Phrase Pivot Statistical Machine Translation

Ahmed El Kholy and **Nizar Habash**
Center for Computational Learning Systems, Columbia University
475 Riverside Drive New York, NY 10115
{akholy,habash}@ccls.columbia.edu

Abstract

An important step in mainstream statistical machine translation (SMT) is combining bidirectional alignments into one alignment model. This process is called symmetrization. Most of the symmetrization heuristics and models are focused on direct translation (source-to-target). In this paper, we present symmetrization heuristic relaxation to improve the quality of phrase-pivot SMT (source-[pivot]-target). We show positive results (1.2 BLEU points) on Hebrew-to-Arabic SMT pivoting on English.

1 Introduction

One of the main issues in statistical machine translation (SMT) is the scarcity of parallel data for many language pairs especially when the source and target languages are morphologically rich. A common SMT solution to the lack of parallel data is to pivot the translation through a third language (called pivot or bridge language) for which there exist abundant parallel corpora with the source and target languages. The literature covers many pivoting techniques. One of the best performing techniques, phrase pivoting (Utiyama and Isahara, 2007), builds an induced new phrase table between the source and target.

Our effort in this paper is based on phrase pivoting. We focus on word alignment to improve translation quality. Word alignment is an essential step in building an SMT system. The most commonly used alignment models, such as IBM Model serial (Brown et al., 1993) and HMM (Och and Ney, 2003), all assume one-to-many alignments. However, the target is to produce a many-to-many word alignment model. A common practice solution in most state-of-the-art MT systems is to create two sets of one-to-many word alignments (bidirectional alignments), source-to-target and target-to-source, and then combine the two sets to produce the final many-to-many word alignment model. This combination process is called "Symmetrization".

In this paper, we propose a symmetrization relaxation method targeting phrase-pivot SMT. Unlike the typical symmetrization methods, the process is carried out as an optimization for phrase-pivot SMT and eventually increase the matching on the pivot phrases. We show positive results (1.2 BLEU points) on Hebrew-Arabic phrase-pivot SMT (pivoting through English).

Next, we briefly present some background information on symmetrization (Section 2) and discuss previous related work in Section 3. This is followed by our symmetrization approach in Section 4. We present our experimental results in Section 5.

2 Background

In this section, we briefly describe different symmetrization heuristics. We then explain how symmetrization affects phrase extraction and discuss the motivation for our approach.

2.1 Symmetrization Heuristics

The simplest approach is to merge the two directional alignment functions using a symmetrization heuristic to produce a many-to-many alignment matrix (Och et al., 1999; Och and Ney, 2003; Koehn et al., 2003).

One of the approaches is to take the intersection (I) of the two directional alignments. Intersection alignment matrices are very sparse and express only one-to-one relationship between words. As a result, we get a high precision in alignment due to the agreement of both models and a very low recall.

An alternative approach is to look at the two alignments as containing complementary information. Therefore, the union (U) of the two models can capture all complementary information. Unlike the intersection (I), many-to-many relationship between words are covered and the resulting matrices are dense. As a result, we get the opposite effect of intersection where we have a higher recall of alignment points but at the cost of losing in precision.

Many mid-way solutions between intersection (I) and (U) can be achieved which aim to balance between precision and recall. Some solutions start from high precision intersection points, and progressively add reliable links from the union to increase recall. Other solutions start from a high recall union points and remove unreliable links to increase precision. One of most commonly used heuristic is **Grow-diag-final-and** (GDFA) (Koehn et al., 2003).

The GDFA heuristic is composed of two steps and one constraint. The first step (**Grow-diag**) starts from the intersection of two directional alignments then gradually considers the neighborhood of each alignment point between the source and target words. The considered neighbors of an alignment point at position (i, j) span over the range of $[i - 1, i + 1]$ for source words and $[j - 1, j + 1]$ for target words. Points in this neighborhood are progressively added to the alignment if neither the source word nor the target word is already aligned and the corresponding point exists in the union (U). The second step (**-final**) adds alignment points that are not neighbor intersection alignment points. This is done for alignment points between words, of which at least one is currently unaligned and exists in the union (U). Adding the constraint (**-and**), only allows alignment points between two unaligned words to be added.

2.2 Symmetrization vs. Phrase Extraction

There is a direct relationship between the final alignment matrix after symmetrization and the phrase extraction process. One way to look at the role of alignment points in extracting phrases is that they act as constraints for which phrase pairs can be extracted. In the standard heuristic (Koehn et al., 2003) for phrase pair extraction, the extracted phrase pair should be consistent and contain at least one word-based link. Moreover, no word inside the phrase pair is aligned to a word outside it. Figure 1 shows examples of phrase pairs that obey or violate the consistency constraint.

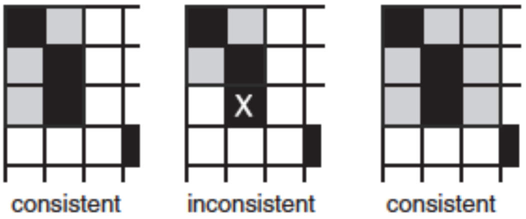


Figure 1: Phrase-pairs consistency constraints with word alignment (black squares are alignment points and the shaded area is a proposed phrase pair): The first example from the left obeys the consistency heuristic, which is violated in the second example (one alignment point in the second column is outside the phrase pair). The third example obeys the consistency heuristic despite the fact that it includes an unaligned word on the right. This diagram is taken from Koehn (2010).

The consistency constraint leads to an inverse relationship between the number of alignment points and the number of phrase pairs extracted; the fewer alignment points, the more phrase pairs can be extracted. This relationship is not valid in the extreme situation with no alignment points at all; in this extreme case, no phrase pairs are extracted.

A major issue in this heuristic is its sensitivity to word alignment errors. Since the consistency constraint is based on the alignment, an error could prevent the extraction of many good phrase pairs. In the context of phrase pivoting, this eventually leads to much less chances to pivot on potential good phrases. This problem motivates our approach to relax the symmetrization process (discussed in Section 4) and generate new pivot phrases in both systems used in pivoting. These new pivot phrases can connect potential source to target phrase pairs.

3 Related Work

Many researchers have investigated the use of pivoting (or bridging) approaches to solve the data scarcity problem (Utiyama and Isahara, 2007; Wu and Wang, 2009; Khalilov et al., 2008; Bertoldi et al., 2008; Habash and Hu, 2009). The main idea is to introduce a pivot language, for which there exist large source-pivot and pivot-target bilingual corpora. Pivoting has been explored for closely related languages (Hajič et al., 2000) as well as unrelated languages (Koehn et al., 2009; Habash and Hu, 2009). Many different pivot strategies have been presented in the literature. The following three are perhaps the most common.

The first strategy is the sentence pivoting technique in which we first translate the source sentence to the pivot language, and then translate the pivot language sentence to the target language (Khalilov et al., 2008).

The second strategy is based on phrase pivoting (Utiyama and Isahara, 2007; Cohn and Lapata, 2007; Wu and Wang, 2009; El Kholy et al., a; El Kholy et al., b). In phrase pivoting, a new source-target phrase table (translation model) is induced from source-pivot and pivot-target phrase tables. Lexical weights and translation probabilities are computed from the two translation models.

The third strategy is to create a synthetic source-target corpus by translating the pivot side of source-pivot corpus to the target language using an existing pivot-target model (Bertoldi et al., 2008).

In this paper, we build on the phrase pivoting approach, which has been shown to be the best with comparable settings (Utiyama and Isahara, 2007).

There are some recent efforts regarding alignment symmetrization or combination. In a related work to our approach but for direct SMT systems, Deng and Zhou (2009) performs alignment symmetrization as an optimization process to maximize the number of phrase translations that can be extracted within a sentence pair. There are other approaches that do not depend on heuristics. Among these are efforts that depend on unsupervised methods (Liang et al., 2006; DeNero and Macherey, 2011) where they jointly learn two directional alignment models. In another direction, Graça et al. (2007) improve bidirectional models by incorporating agreement constraints to EM training using Posterior Regularization (PR). Moreover, DeNero and Macherey (2011) proposed a model based aligner combination using dual decomposition.

Since both Hebrew and Arabic are morphologically rich, we should mention that there has been a lot of work on translation to and from morphologically rich languages (Yeniterzi and Oflazer, 2010; Elming and Habash, 2009; El Kholy and Habash, 2010a; Habash and Sadat, 2006; Kathol and Zheng, 2008). Most of these efforts are focused on syntactic and morphological processing to improve translation quality.

Until recently, there has not been much parallel Hebrew-English (Tsvetkov and Wintner, 2010) and Hebrew-Arabic data, and consequently little work on Hebrew-English and Hebrew-Arabic SMT. Lavie et al. (2004) built a transfer-based translation system for Hebrew-English and so did Shilon et al. (2012) for translation between Hebrew and Arabic.

Algorithm 1 Symmetrization Relaxation Algorithm (starting with union symmetrization)

```
{ generate the list of possible pivot unigram $L_p$}
$A^U_{pt} = \overrightarrow{A_{pt}} \cup \overleftarrow{A_{pt}}$
$A^F_{pt} = A^U_{pt}$
for $(i,j) \in A^F_{pt}$ do
    if $W_i \notin L_p$ then
        $A^F_{pt} = A^F_{pt} - \{(i,j)\}$
    end if
end for
return $A^F_{pt}$
```

To our knowledge this is the first study improving phrase-pivot SMT for Hebrew-Arabic SMT. We successfully show that relaxing alignment symmetrization targeting pivoting and combining the extracted phrases with the best baseline system improve translation quality.

4 Approach

In this section, we explain our approach in relaxing the symmetrization process to improve the matching in phrase-pivot SMT. We then discuss our approach in combining the phrase pairs extracted from the basic pivot system and a pivot system using our relaxation approach which leads to our best results.

4.1 Symmetrization Relaxation

Our proposed approach is based on two parts. The first part is constructing a list of all possible pivot unigram phrases L_p that can be used in the pivoting process. This can simply be done by getting the intersection of all the pivot unigrams extracted from both the source-pivot and the pivot-target corpora.

In the second part, we start by building two directional alignment models: pivot-to-target $\overrightarrow{A_{pt}}$ and target-to-pivot $\overleftarrow{A_{pt}}$. Following Algorithm 1, we can start with union A^U_{pt} or grow-diag-final-and A^{GDFA}_{pt} alignment symmetrization. We then relax the symmetrization to allow the extraction of many new pivot phrases by removing a given word link that links a target word to a pivot word that is NOT in L_p. The final alignment matrix after all the deletions is A^F_{pt}. To remind the reader, alignment

Symm.	He-En				En-Ar				He-En-Ar	
	$\|A_{sp}\|$	$\frac{\|A_{sp}\|}{\|A^U_{sp}\|}$	$\|PT_{sp}\|$	$\frac{\|PT_{sp}\|}{\|PT^I_{sp}\|}$	$\|A_{pt}\|$	$\frac{\|A_{pt}\|}{\|A^U_{pt}\|}$	$\|PT_{pt}\|$	$\frac{\|PT_{pt}\|}{\|PT^I_{pt}\|}$	$\|PT_{st}\|$	$\frac{\|PT_{st}\|}{\|PT^I_{st}\|}$
I	0.6M	45%	15.0M	100%	0.7M	57%	11.4M	100%	1707M	100%
U	1.4M	100%	0.9M	6%	1.3M	100%	1.3M	12%	1M	0.1%
U_R	1.2M	89%	1.7M	11%	1.2M	91%	2.3M	21%	245M	14%
GDFA	1.1M	79%	3.0M	20%	1.0M	85%	3.0M	27%	267M	16%
GDFA_R	1.0M	73%	4.4M	30%	1.0M	78%	4.6M	40%	1105M	65%

Table 1: Comparison of symmetrization methods in terms of alignment set size, resulting phrase tables size (in millions) for each size SMT system used in pivoting and the final pivot phrase table.

points deletion (a.k.a alignment symmetrization relaxation) allows the extraction of more phrases.

We repeat the whole process in the other language pair of the pivoting, source-pivot, to get the final alignment set A^F_{sp}. Then, these final alignment matrices are used to extract two phrase tables PT_{sp} and PT_{pt} which are used in the phrase pivoting process to produce the final pivot phrase table PT_{st}.

Table 1 shows the impact of different word alignment symmetrization methods on phrase tables for each system used in Hebrew-Arabic phrase-pivot SMT (He-En & En-Ar) and the final phrase table (He-En-Ar).[1] We compare each method with and without our relaxation approach. The first row in the table is the intersection (I). The next two are union (U) without relaxation and then union with relaxation (U_R). The next two methods are heuristic grow-diagonal-final-and (GDFA) without relaxation and with relaxation (GDFA_R).

For each particular symmetrization method and each system used in pivoting, we compute the output alignment set size in first & fifth columns and their percentage of the union in second & sixth columns. We also compute the size of the resulting phrase tables. The numbers show the inverse relationship between the alignment set size and the phrase table sizes. The most sparse matrix in intersection leads to huge phrase tables which consequently leads a exponentially huge final pivot phrase table with potentially a lot of low quality phrase pairs. The union has an opposite effect. It has a higher recall of alignment points including some bad alignment points that can prevent the extraction of good pivoting phrase pairs.

Figure 2 illustrates how the proposed symmetrization relaxation approach can lead to good and bad English-Arabic phrase pairs.[2] The English-Arabic phrase pair (B1) is extracted into the original baseline phrase table. The word "phased" is erroneously aligned to the Arabic word وفق *wfq* 'according to/under' which prevents the extraction of smaller phrase pairs because of the consistency constraint (discussed in Section 2.2). Since the word "phased" does not appear in the English side of the Hebrew-English corpus, our relaxation method will drop all the alignment points which are connected to the word "phased". This allows the extraction of a couple of new phrase pairs (R1a & R1b). (R1a) is not a good phrase pair since it includes an extra word ("phased") in the English side that is absent in the Arabic. That said, it will not be used in the pivoting.(R1b), on the other hand, is a good phrase pair that could lead to a pivot match.

The lower half of Figure 2 illustrates how symmetrization relaxation does not always lead to good phrase pairs. The English-Arabic phrase pair (B2), which appears in the original baseline phrase table, is a perfectly good phrase pair. However, since the word "Saloniki" doesn't appear in the English side of the Hebrew-English corpora, deleting it leads to the creation of two bad phrase pairs (R2a & R2b) where the English and Arabic side do not have the same meaning.

4.2 Model Combination

The alignment symmetrization relaxation explained in Section 4.1 leads to an increase in the number of phrase pairs extracted in the translation model. Some of these phrase pairs would be useful but many others are of low quality which affects the translation choices during decoding and the overall translation quality as shown in Figure 2.

As a solution, we construct a combined phrase table using phrase pairs from the best baseline pivoting system without relaxation and then add any

[1] The experimental setup is discussed in details in Section 5.1

[2] We use the Habash-Soudi-Buckwalter Arabic transliteration (Habash et al., 2007).

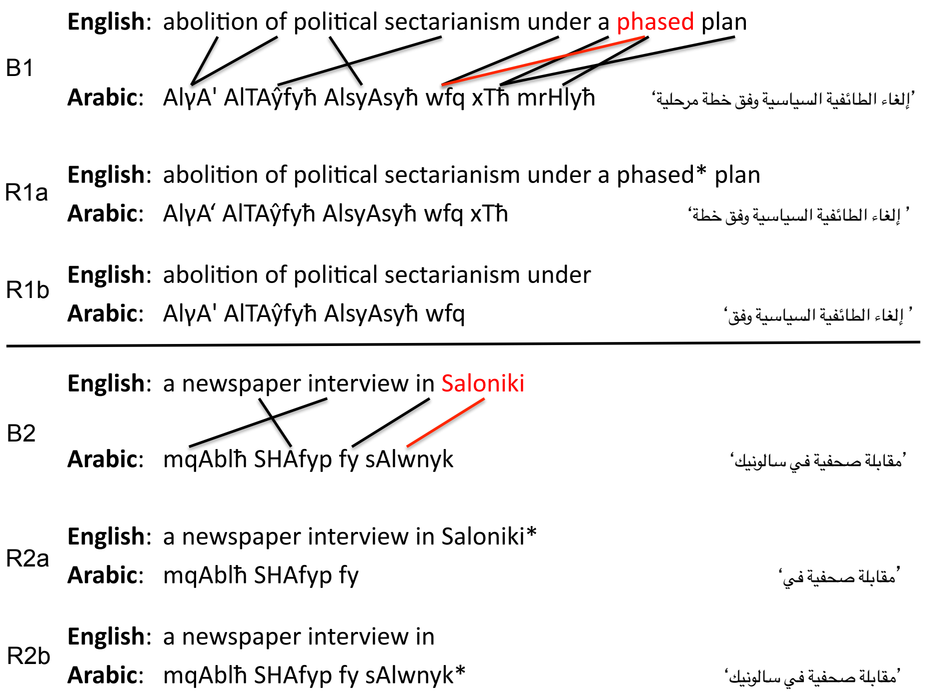

English: abolition of political sectarianism under a phased plan
B1
Arabic: AlγA' AlTAŷfyħ AlsyAsyħ wfq xTħ mrHlyħ
'إلغاء الطائفية السياسية وفق خطة مرحلية'
R1a
English: abolition of political sectarianism under a phased* plan
Arabic: AlγA' AlTAŷfyħ AlsyAsyħ wfq xTħ
'إلغاء الطائفية السياسية وفق خطة'
R1b
English: abolition of political sectarianism under
Arabic: AlγA' AlTAŷfyħ AlsyAsyħ wfq
'إلغاء الطائفية السياسية وفق'
English: a newspaper interview in Saloniki
B2
Arabic: mqAblħ SHAfyp fy sAlwnyk
'مقابلة صحفية في سالونيك'
R2a
English: a newspaper interview in Saloniki*
Arabic: mqAblħ SHAfyp fy
'مقابلة صحفية في'
R2b
English: a newspaper interview in
Arabic: mqAblħ SHAfyp fy sAlwnyk*
'مقابلة صحفية في سالونيك'

Figure 2: Two examples of baseline (GDFA) phrase pairs (B1 & B2) together with two pairs of phrases that are generated after symmetrization relaxation (R1a, R1b, R2a &R2b). The alignment links that are deleted as part of symmetrization relaxation are colored in red. The words marked with an asterisk do not have an equivalent in the opposite language in the phrase pair they appear in. The examples are discussed in detail in Section 4.1.

additional phrase pairs extracted after relaxation. We add a binary feature $f_{\mathbf{s},\mathbf{t}}$ to the log linear space of features in order to mark the source of the pivot phrase pairs as follows:[3]

$$f_{(\mathbf{s},\mathbf{t})} = \begin{cases} 2.718 & \text{if } (\mathbf{s},\mathbf{t}) \text{ from the baseline system} \\ 1 & \text{otherwise} \end{cases} \quad (1)$$

The aim from the added binary feature is to bias the translation model after tuning to favor phrase pairs from the baseline system over the complementary phrase pairs from the relaxed model.

5 Experiments

Next, we present a set of experiments on symmetrization relaxation for phrase-pivot SMT and on model combination.

5.1 Experimental Setup

In our pivoting experiments, we build two SMT models; one model to translate from Hebrew to English and another model to translate from English to Arabic. For both models, we use the same size of parallel corpus($\approx$ 1M words) despite the fact that more English-Arabic data are available. The English-Arabic parallel corpus is a subset of available data from LDC.[4] The Hebrew-English corpus is available from sentence-aligned corpus produced by Tsvetkov and Wintner (2010).

Word alignment is done using GIZA++ (Och and Ney, 2003). For Arabic language modeling, we use 200M words from the Arabic Gigaword Corpus (Graff, 2007) together with the Arabic side of our training data. We use 5-grams for all language models (LMs) implemented using the SRILM toolkit (Stolcke, 2002).

All experiments are conducted using the Moses phrase-based SMT system (Koehn et al., 2007). We use MERT (Och, 2003) for decoding weight optimization. Weights are optimized using a set of 517 sentences (single reference) developed by Shilon et al. (2010).

We use a maximum phrase length of size 8 across all models. We report results on a Hebrew-Arabic evaluation set of 300 sentences with three references developed by Shilon et al. (2010). We evaluate using BLEU-4 (Papineni et al., 2002),

[3] The log values of 2.718 and 1 will lead to a binary representation in the log linear space.

[4] LDC Catalog IDs: LDC2004T17, LDC2004E72, LDC2005E46, LDC2004T18

METEOR v1.4 (Lavie and Agarwal, 2007) and TER (Snover et al., 2006).

5.2 Linguistic Preprocessing

In this section we present our motivation and choice for preprocessing Arabic, Hebrew and English data. Both Arabic and Hebrew are morphologically complex languages (Fabri et al., 2014). One aspect of Arabic's complexity is its various attachable clitics and numerous morphological features (Habash, 2010). which include conjunction proclitics, e.g., +و *w*+ 'and', particle proclitics, e.g., +ل *l*+ 'to/for', the definite article +ال *Al*+ 'the', and the class of pronominal enclitics, e.g., هم+ +*hm* 'their/them'. Beyond these clitics, Arabic words inflect for person, gender, number, aspect, mood, voice, state and case. This morphological richness leads to thousands of inflected forms per lemma and a high degree of ambiguity: about 12 analyses per word, typically corresponding to two lemmas on average (Habash, 2010). We follow El Kholy and Habash (2010a) and use the PATB tokenization scheme (Maamouri et al., 2004) in our experiments which separates all clitics except for the determiner clitic *Al*+. We use MADA v3.1 (Habash and Rambow, 2005; Habash et al., 2009) to tokenize the Arabic text. We only evaluate on detokenized and orthographically correct (enriched) output following the work of El Kholy and Habash (2010b).

Similar to Arabic, Hebrew poses computational processing challenges typical of Semitic languages (Itai and Wintner, 2008; Shilon et al., 2012; Habash, 2010). Hebrew inflects for gender, number, person, state, tense and definiteness. Furthermore, Hebrew has a set of attachable clitics that are typically separate words in English, e.g., conjunctions (such as +ו *w*+ 'and'),[5] prepositions (such as +ב *b*+ 'in'), the definite article (+ה *h*+ 'the'), or pronouns (such as הם+ +*hm* 'their'). These issues contribute to a high degree of ambiguity that is a challenge to translation from Hebrew to English or to any other language. We use the best preprocessing scheme for Hebrew (HTAG) identified by Singh and Habash (2012) .

English, our pivot language, is quite different from both Arabic and Hebrew. English is morphologically poor and barely inflects for number, person and tense. English preprocessing simply includes down-casing, separating punctuation and splitting off "'s".

Symm.	BLEU	METEOR	TER
GDFA	20.4	33.4	62.7
GDFA_R	**20.8**	**34.0**	**62.4**
U	20.1	33.5	62.7
U_R	20.7	34.0	62.5
I	20.8	34.0	63.6

Table 2: Symmetrization relaxation results for different symmetrization methods. The best performer is the relaxed grow-diag-final-and (GDFA_R). (GDFA_R) BLEU score is statistically significant over the baseline (GDFA) with p-value $= 0.12$. All other results are not statistically significant.

5.3 Symmetrization Relaxation

We compare the performance of symmetrization relaxation in contrast with different symmetrization methods. The results are presented in Table 2. In general, as expected grow-diag-final-and (GDFA) outperforms all other symmetrization methods and it is considered our baseline. Moreover, the performance improves with the symmetrization relaxation for both union (U_R) and grow-diag-final-and (GDFA_R) and the best performer is the relaxed grow-diag-final-and (GDFA_R). While (I) leads to comparable results to (GDFA_R), BLEU score against the baseline (GDFA) is not statistically significant and TER is the worst across all methods.[6]

Since (GDFA_R) is the best performing model, we use (GDFA) and (GDFA_R) in our model combination experiments, next.

5.4 Model Combination

We test the performance of combining the baseline (GDFA) phrase table with the relaxed (GDFA_R) phrase table as explained in Section 4.2.

The results in Table 3 show that we get a nice improvement of 1.2/1/0.8 (BLEU/METEOR/TER) points by combing the two models (GDFA) and (GDFA_R). The difference in BLEU score is statistically significant with p-value < 0.01. This re-

[5] The following Hebrew 1-to-1 transliteration is used (in Hebrew lexicographic order): *abgdhwzxTiklmns'pcqršt*. All examples are undiacritized and final forms are not distinguished from non-final forms.

[6] Statistical significance is done using MultEval (https://github.com/jhclark/multeval) which implements statistical significance testing between systems based on multiple optimizer runs and approximate randomization (Resampling, 1989; Clark et al., 2011)

Symm.	BLEU	METEOR	TER
GDFA	20.4	33.4	62.7
GDFA_R	20.8	34.0	62.4
GDFA+GDFA_R	**21.6**	**34.4**	**61.6**

Table 3: Model combination experiment result. (GDFA+GDFA_R) shows a big improvement in BLEU score which is statistically significant with p-value < 0.01.

sult shows that our relaxation approach helps in combination with a baseline system to improve the overall translation quality. Moreover, since (GDFA_R) is a proper super-set of (GDFA) by design then the big jump in performance is due to the additional binary feature added to the log linear model. As we hoped, the binary feature biases the combined model towards the more trusted phrase pairs from (GDFA) and complement the translation model with the additional phrase pairs from symmetrization relaxation.

6 Conclusion and Future Work

We presented a symmetrization relaxation method targeting phrase-pivot SMT. The symmetrization is carried out as an optimization process to increase the matching on the pivot phrases. We show positive results (1.2 BLEU points) on Hebrew-Arabic phrase-pivot SMT. In the future, we plan to work on symmetrization based on morpho-syntactic information between Hebrew and Arabic. We expect an improvement in quality since both languages come from the same Semitic family. We also plan to work on techniques to determine the quality of pivot phrase pairs using alignment information and relationships between the three languages used in pivoting.

Acknowledgments

The work presented in this paper was funded by a Google research award. We would like to thank Reshef Shilon for helpful feedback and discussions. We also thank the anonymous reviewers for their insightful comments.

References

Bertoldi, Nicola, Madalina Barbaiani, Marcello Federico, and Roldano Cattoni. 2008. Phrase-based statistical machine translation with pivot languages. *Proc. of International Workshop on Spoken Language Translation (IWSLT'08)*.

Brown, P. F., S. A. Della Pietra, V. J. Della Pietra, and R. L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*.

Clark, Jonathan H, Chris Dyer, Alon Lavie, and Noah A Smith. 2011. Better hypothesis testing for statistical machine translation: Controlling for optimizer instability. In *Proc. of Association for Computational Linguistics (ACL'11)*.

Cohn, Trevor and Mirella Lapata. 2007. Machine translation by triangulation: Making effective use of multi-parallel corpora. In *Proc. of ACL'07*.

DeNero, John and Klaus Macherey. 2011. Model-based aligner combination using dual decomposition. In *Proc. of ACL'11*.

Deng, Yonggang and Bowen Zhou. 2009. Optimizing word alignment combination for phrase table training. In *Proc. of the International Joint Conference on Natural Language Processing (IJCNLP'09)*, Suntec,Singapore.

El Kholy, Ahmed and Nizar Habash. 2010a. Orthographic and Morphological Processing for English-Arabic Statistical Machine Translation. In *Proc. of Traitement Automatique du Langage Naturel (TALN'10)*. Montréal, Canada.

El Kholy, Ahmed and Nizar Habash. 2010b. Techniques for Arabic Morphological Detokenization and Orthographic Denormalization. In *Proc. of the International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta.

El Kholy, Ahmed, Nizar Habash, Gregor Leusch, Evgeny Matusov, and Hassan Sawaf. Language independent connectivity strength features for phrase pivot statistical machine translation. In *Proc. ACL'13*, Sofia, Bulgaria.

El Kholy, Ahmed, Nizar Habash, Gregor Leusch, Evgeny Matusov, and Hassan Sawaf. Selective combination of pivot and direct statistical machine translation models. In *Proc. IJCNLP'13*, Nagoya, Japan.

Elming, Jakob and Nizar Habash. 2009. Syntactic Reordering for English-Arabic Phrase-Based Machine Translation. In *Proc. of the EACL'09*, Athens, Greece.

Fabri, Ray, Michael Gasser, Nizar Habash, George Kiraz, and Shuly Wintner. 2014. Linguistic introduction: The orthography, morphology and syntax of semitic languages. In *Natural Language Processing of Semitic Languages*, pages 3–41. Springer.

Graça, Joao, Kuzman Ganchev, and Ben Taskar. 2007. Expectation maximization and posterior constraints. In *Proc. of Conference on Neural Information Processing Systems (NIPS'07)*.

Graff, David. 2007. Arabic Gigaword 3, LDC Catalog No.: LDC2003T40. Linguistic Data Consortium (LDC), University of Pennsylvania.

Habash, Nizar and Jun Hu. 2009. Improving Arabic-Chinese Statistical Machine Translation using English as Pivot Language. In *Proc. of the Workshop on Statistical Machine Translation (WMT'09)*, Athens, Greece, March.

Habash, Nizar and Owen Rambow. 2005. Arabic Tokenization, Part-of-Speech Tagging and Morphological Disambiguation in One Fell Swoop. In *Proc. of ACL'05*, Ann Arbor, Michigan.

Habash, Nizar and Fatiha Sadat. 2006. Arabic Preprocessing Schemes for Statistical Machine Translation. In *Proc. of NAACL'06*, New York City, USA.

Habash, Nizar, Abdelhadi Soudi, and Tim Buckwalter. 2007. On Arabic Transliteration. In *Arabic Computational Morphology: Knowledge-based and Empirical Methods*. Springer.

Habash, Nizar, Owen Rambow, and Ryan Roth. 2009. MADA+TOKAN: A toolkit for Arabic tokenization, diacritization, morphological disambiguation, POS tagging, stemming and lemmatization. In *Proc. of the Second International Conference on Arabic Language Resources and Tools*. The MEDAR Consortium.

Habash, Nizar. 2010. *Introduction to Arabic Natural Language Processing*. Morgan & Claypool Publishers.

Hajič, Jan, Jan Hric, and Vladislav Kubon. 2000. Machine Translation of Very Close Languages. In *Proc. of the Applied Natural Language Processing Conference (ANLP'00)*, Seattle.

Itai, Alon and Shuly Wintner. 2008. Language resources for Hebrew. In *Proc. of LREC'08*.

Kathol, Andreas and Jing Zheng. 2008. Strategies for building a Farsi-English smt system from limited resources. In *Proc. of the Annual Conference of the International Speech Communication Association (INTERSPEECH'08)*, Brisbane, Australia.

Khalilov, M., Marta R. Costa-juss, Jos A. R. Fonollosa, Rafael E. Banchs, B. Chen, M. Zhang, A. Aw, H. Li, Jos B. Mario, Adolfo Hernndez, and Carlos A. Henrquez Q. 2008. The talp & i2r smt systems for iwslt 2008. In *Proc. of IWSLT'08*

Koehn, Philipp, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proc. NAACL'03*.

Koehn, Philipp, Hieu Hoang, Alexandra Birch, Christopher Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Christopher Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: open source toolkit for statistical machine translation. In *Proc. of ACL'07*, Prague, Czech Republic.

Koehn, Philipp, Alexandra Birch, and Ralf Steinberger. 2009. 462 machine translation systems for europe. *Proc. of MT Summit XII*.

Koehn, Philipp. 2010. *Statistical machine translation*. Cambridge University Press.

Lavie, Alon and Abhaya Agarwal. 2007. Meteor: An automatic metric for mt evaluation with high levels of correlation with human judgments. In *Proc. of WMT'07*, Prague, Czech Republic.

Lavie, Alon, Katharina Probst, Erik Peterson, Stephan Vogel, Lori Levin, Ariadna Font-Llitjos, and Jaime Carbonell. 2004. A trainable transfer-based machine translation approach for languages with limited resources. In *Proc. of European Association for Machine Translation (EAMT'04)*, Malta.

Liang, Percy, Ben Taskar, and Dan Klein. 2006. Alignment by agreement. In *Proc. of ACL'06*.

Maamouri, Mohamed, Ann Bies, Tim Buckwalter, and Wigdan Mekki. 2004. The Penn Arabic Treebank: Building a Large-Scale Annotated Arabic Corpus. In *NEMLAR Conference on Arabic Language Resources and Tools*, Cairo, Egypt.

Och, Franz Josef and Herman Ney. 2003a. A Systematic Comparison of Various Statistical Alignment Models. Computational Linguistics.

Och, Franz Josef, Christoph Tillmann, and Hermann Ney. 1999. Improved alignment models for statistical machine translation. In *Proc. EMNLP'99*.

Och, Franz Josef. 2003. Minimum error rate training in statistical machine translation. In *Proc. of ACL'03*.

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a Method for Automatic Evaluation of Machine Translation. In *Proc. of ACL'02*, Philadelphia, PA.

Resampling, Bootstrap. 1989. Computer-intensive methods for testing hypotheses: an introduction. *Computer*.

Shilon, Reshef, Nizar Habash, Alon Lavie, and Shuly Wintner. 2010. Machine translation between Hebrew and Arabic: Needs, challenges and preliminary solutions. In *Proc. of the Association for Machine Translation in the Americas (AMTA'10)*.

Shilon, Reshef, Nizar Habash, Alon Lavie, and Shuly Wintner. 2012. Machine translation between Hebrew and Arabic. *Machine Translation*.

Singh, Nimesh and Nizar Habash. 2012. Hebrew morphological preprocessing for statistical machine translation. In *Proc. of The European Association for Machine Translation (EAMT'12)*.

Snover, Matthew, Bonnie J. Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proc. of AMTA'06*, Boston, Massachusetts.

Stolcke, Andreas. 2002. SRILM - an Extensible Language Modeling Toolkit. In *Proc. of the International Conference on Spoken Language Processing (ICSLP'02)*, Denver, CO.

Tsvetkov, Y. and S. Wintner. 2010. Automatic acquisition of parallel corpora from websites with dynamic content. In *Proc. of LREC'10*.

Utiyama, Masao and Hitoshi Isahara. 2007. A comparison of pivot methods for phrase-based statistical machine translation. In *Proc. of ACL'07*, Rochester, New York.

Wu, Hua and Haifeng Wang. 2009. Revisiting pivot language approach for machine translation. In *Proc. of ACL-AFNLP'09*, Suntec, Singapore.

Yeniterzi, Reyyan and Kemal Oflazer. 2010. Syntax-to-morphology mapping in factored phrase-based statistical machine translation from english to Turkish. In *Proc. of ACL'10*, Uppsala, Sweden.

Improving Machine Translation via Triangulation and Transliteration

Nadir Durrani
School of Informatics
University of Edinburgh
dnadir@inf.ed.ac.uk

Philipp Koehn
School of Informatics
University of Edinburgh
pkoehn@inf.ed.ac.uk

Abstract

In this paper we improve Urdu→Hindi⇄English machine translation through triangulation and transliteration. First we built an Urdu→Hindi SMT system by inducing triangulated and transliterated phrase-tables from Urdu–English and Hindi–English phrase translation models. We then use it to translate the Urdu part of the Urdu-English parallel data into Hindi, thus creating an artificial Hindi-English parallel data. Our phrase-translation strategies give an improvement of up to +3.35 BLEU points over a baseline Urdu→Hindi system. The synthesized data improve Hindi→English system by +0.35 and English→Hindi system by +1.0 BLEU points.

1 Introduction

Phrase-based machine translation models are capable of producing translations of reasonable quality but only with large quantities of parallel data. Unfortunately, bilingual data is abundantly available for only a handful of language pairs. The problem of reliably estimating statistical models for translation becomes more of a challenge under sparse data conditions. Previous research has tried to address this bottleneck in two ways i) by making the best use of the existing small corpus by using generalized representations such as morpho-syntactic analysis and suffix separation (Niessen and Ney, 2004; Popović and Ney, 2004; Haque et al., 2012), ii) by generating additional data/inducing models to overcome sparsity (Utiyama and Isahara, 2003; Resnik and Smith, 2003). Techniques like triangulation (Cohn and Lapata, 2007; Wu and Wang, 2007) and paraphrasing (Callison-Burch et al., 2006) have also been used to address the problem of data sparsity. Transliteration is shown to be useful when the languages in question are closely related (Durrani et al., 2010; Nakov and Tiedemann, 2012). Our work falls in this second category of generating additional/data and models.

Hindi and Urdu are widely spoken yet low resourced languages. Hindi descends from Sanskrit and is written in Devanagri script, where as Urdu inherits its vocabulary and language phenomenon from several languages (Arabic, Farsi and Turkish and Sanskrit) and is written in Arabic script. They are a closely related language pair that share grammatical structure and have a high vocabulary overlap.[1] This provides a motivation to build an MT system to create Hindi and Urdu resources by translating one into another.

In this paper, we exploit the relatedness of the two languages and bring together the ideas of triangulation and transliteration to effectively improve Urdu-to-Hindi machine translation. We make use of a tiny Hindi-Urdu parallel corpus, to build a Urdu-to-Hindi translation system. We then improve this system by synthesizing phrase-tables through triangulation and transliteration. We create a triangulated phrase-table using English as a pivot language following the well-known convolution model (Utiyama and Isahara, 2007; Wu and Wang, 2007). The new phrase-table is synthesized using Hindi-English and Urdu-English phrase-tables. We then use the interpolated phrase-table to also synthesize a transliteration phrase-

[1] A small study on 2 newspapers (Dainik Jagran and Hindustan), found that ≈ 40% of the Hindi types overlap with Urdu.

table. We run an EM-based unsupervised transliteration model induction (Durrani et al., 2014c) to extract a list of transliteration pairs from the extracted phrase-table. We use the mined corpus to train a transliteration system. The transliteration system is then used to synthesize a transliteration phrase-table. We produce an n-best list of transliterations for each Urdu word in the tune and test data and create a transliteration phrase table using the features of the transliteration system. The two synthesized phrase-tables are then used as additional set of features along with the regular phrase-table in a log-linear framework. Our integrations give a cumulative improvement of +3.35 BLEU points over the baseline Urdu-to-Hindi system.

In order to demonstrate that the resources in Urdu language can be used for Hindi, we perform an extrinsic evaluation. We use our Urdu-to-Hindi system and translate the Urdu part of the Urdu-English parallel data into Hindi to create an artificial Hindi-English parallel data. Our experiments show that the synthesized parallel data gives an average improvement of +0.35 in Hindi-to-English and +1.0 in English-to-Hindi standard shared task. Our approach is two-way: we use the information in Hindi-English and Urdu-English data to construct a Urdu-to-Hindi system, which we then use for synthesizing Hindi data and subsequently improving Hindi-English translation.

This paper is organized as follows. Section 2 discusses previous efforts on synthesizing parallel data and using pivoting and transliteration to improve MT. Section 3 describe our approach to create interpolated and transliterated phrase-tables and our integration of these into a phrase-based decoder. Section 4 presents the experimental setup and the results. Section 5 concludes the paper.

2 Related Work

There has been a considerable amount of work on synthesizing parallel data and on using triangulation and transliteration to improve machine translation quality. de Gispert and Mariño (2006) induced an English-Catalan parallel corpus by automatically translating Spanish part of English-Spanish parallel data into Catalan with a Spanish-Catalan SMT system. Galuscáková and Bojar (2012) improved English-to-Slovak translation using Czech-English parallel data. Our work is similar to both these efforts except that in their case a lot of parallel data is available for the aiding languages (Czech-English and Spanish-English). In our case both Urdu-English and Hindi-English data are under-resourced. Secondly Urdu and Hindi are written in different scripts so unlike them we need a high quality transliteration module to make use of the common vocabulary. Using pivoting to synthesize phrase-tables has been a widely applied method (Wu and Wang, 2007; Bertoldi et al., 2008; Paul et al., 2009) in SMT. An intermediate language is used to bridge the gap between source and target.

Another approach to alleviate data sparsity is the sentence translation strategy. Rather than building phrase-table source sentences are translated into n pivot sentences which are translated into m target sentences separately. Highest scoring sentences are selected. Utiyama and Isahara (2007) showed that phrase-translation approach is superior to the sentence selection strategy. We will use the phrase-translation strategy to improve the Urdu-to-Hindi translation and sentence selection method to improve Hindi$\rightleftarrows$English translation, although we only use 1-best pivot translation.

A second group of previous research that is related to our work is using transliteration to improve translation for closely related languages. Transliteration has been shown to be useful for more than just translating out-of-vocabulary words and named-entities. Nakov and Tiedemann (2012) built character-based model to improve Bulgarian-Macedonian translation. Durrani et al. (2010) integrated transliteration inside a word-based decoder for Hindi-to-Urdu machine translation. Our work is similar to them, but differs in the following aspects: i) their translation models are based on 1-1/1-N translation links, we do not put any restriction on the alignments ii) their transliteration system is built from hand-crafted rules, our approach is unsupervised and language independent and iii) we additionally integrate pivoting method along with transliteration and demonstrate the usefulness of the synthesized Hindi data. The idea to integrate transliteration module inside of decoder was earlier used by Hermjakob et al. (2008) for the task of disambiguation in Arabic-English machine translation. Much work (Al-Onaizan and Knight, 2002; Zhao et al., 2007) has been done on transliterating named entities and OOVs. Most previous approaches however train a supervised transliteration system separately outside of an MT pipeline, and naïvely replace the OOV words with their 1-best transliterations in the post/pre-processing step of decoding is commonly used. This work distinguishes from the previous approaches in that it builds a transliteration model automatically from the phrase-tables in an unsupervised fashion and it is language independent.

3 Phrase Translation Strategies

Monolingual data is usually available in a reasonable quantity for many language pairs, but bilingual corpus does not exist or is very sparse. We only need to construct a phrase-table to train a phrase-based SMT system. In this section we will describe our approaches to construct synthetic phrase-tables that help us improve Urdu-to-Hindi translation, and subsequently improve Hindi-to-English and English-to-Hindi translation quality. We used two approaches namely **Triangulation** and **Transliteration** to generate artificial phrase-tables.

3.1 Triangulation

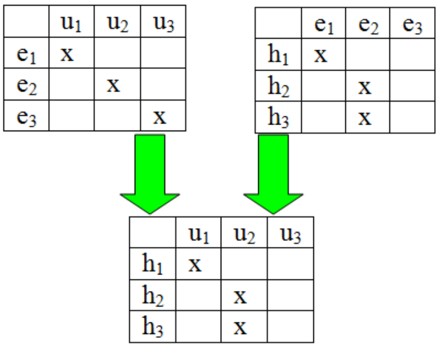
The approach of triangulation is based on using a pivot language to bridge the gap between Urdu and Hindi. Pivot is usually a language closely related to either source or target, or English for which parallel data with either source or target is available in a reasonable quantity. In this work we will use English as a pivot language because we have some Hindi-English and Urdu-English parallel data but very little Urdu-Hindi parallel data. We directly construct Urdu-Hindi phrase-table from Urdu-English and English-Hindi phrase-tables. We train two phrase translation tables $p(\bar{u}_i|\bar{e}_i)$ and $p(\bar{e}_i|\bar{h}_i)$, using the Urdu-English and English-Hindi bilingual corpora. Given the phrase-table for Urdu-English $p(\bar{u}_i|\bar{e}_i)$ and the phrase-table for English-Hindi $p(\bar{e}_i|\bar{h}_i)$, we estimate a Urdu-Hindi phrase-table ($p(\bar{u}_i|\bar{h}_i)$) using the following model:

$$p(\bar{u}_i|\bar{h}_i) = \sum_{\bar{e}_i} p(\bar{u}_i|\bar{e}_i, \bar{h}_i) p(\bar{e}_i|\bar{h}_i)$$

The phrase translation probability $p(\bar{u}_i|\bar{e}_i, \bar{h}_i)$ does not depend on the phrase $\bar{h}_i$, because it is estimated from Urdu-English bilingual corpus. The above equation can therefore be rewritten as:

$$p(\bar{u}_i|\bar{h}_i) = \sum_{\bar{e}_i} p(\bar{u}_i|\bar{e}_i) p(\bar{e}_i|\bar{h}_i)$$

A phrase-pair $(\bar{u}_i, \bar{h}_i)$ is synthesized if there exists an English phrase $\bar{e}_i$ such that $(\bar{u}_i, \bar{e}_i)$ exists in the phrase-table $p(\bar{e}_i|\bar{u}_i)$ and $(\bar{e}_i, \bar{h}_i)$ exists in the phrase-table $p(\bar{h}_i|\bar{e}_i)$. The probability of the new phrase-pair is estimated by taking the product of the two and by taking a summation over all such phrases $\bar{e}_i$ for which this condition is true.



	u1	u2	u3
e1	x		
e2		x	
e3			x

	e1	e2	e3
h1	x		
h2		x	
h3		x	

	u1	u2	u3
h1	x		
h2		x	
h3		x	

Figure 1: Alignment Induction for Phrase $(\bar{u}, \bar{h})$

Lexical Weighting: Apart from the direct and inverse phrase-translation probabilities, phrase-based translation models also estimate direct and inverse lexical weighting to judge the reliability of a phrase-pair using IBM Model 1. We could estimate these probabilities in the same way as the phrase-translation probabilities as done in Utiyama and Isahara (2007), but we use the more principled *phrase method* as described in Wu and Wang (2007). Given a phrase-pair $(\bar{u}, \bar{h})$ with source words $\bar{u} = u_1, u_2, \ldots, u_n$, target words $\bar{h} = h_1, h_2, \ldots, h_m$ and an alignment a between the source word positions $x = 1, \ldots, n$ and the target word positions $y = 1, \ldots, m$, the lexical feature $p_w(u|e)$ is computed as follows:

$$p_w(\bar{u}|\bar{h}, a) = \prod_{x=1}^{n} \frac{1}{|\{y : (x, y) \in a\}|} \sum_{\forall(x,y) \in a} w(u_x|h_y)$$

But in this scenario we do not have the co-occurring frequencies $c(u_x|h_y)$ to compute $w(u_x|h_y)$. The *phrase method* computes it in the following way: The first step is to extract the alignment information a between the Hindi and Urdu phrases $\bar{h}$ and $\bar{u}$. The alignment information from the phrase-pairs $(\bar{u}, \bar{e})$ and $(\bar{e}, \bar{h})$, can be induced in the following way. Let a_1 and a_2 represent the alignment information inside the phrase-pair $(\bar{u}, \bar{e})$ and $(\bar{e}, \bar{h})$ respectively, then the alignment a between phrase $(\bar{u}, \bar{h})$ can be extracted with the following criteria (See Figure 1 for Example):

$$a = \{(u, h) | \exists e : (u, e) \in a_1 \wedge (e, h) \in a_2\}$$

Given the induced word-alignment a, we can estimate the $w(u|h)$ as follows:

$$w(u|h) = \frac{c(u|h)}{\sum_{u'} c(u'|h)}$$

the co-occurring frequency $c(u|h)$ can be computed with the following criteria:

$$c(u|h) = \sum_{k=1}^{K} p_k(\bar{u}|\bar{h}) \sum_{i=1}^{n} \delta(u, u_i)\delta(h, h_{a_i}))$$

where $\delta(x, y) = 1$ if $x = y$, otherwise $\delta(x, y) = 0$ and $p(\bar{u}|\bar{h})$ is the phrase-translation probability of the k^{th} phrase and K is the total number of phrases synthesized.

3.2 Transliteration

Hindi and Urdu are written in different scripts. A high quality transliteration system is therefore required to capitalize on the vocabulary overlap between Hindi and Urdu. Our second approach attempts to synthesize transliteration phrase-table. A transliteration module can be handy for closely related languages because it can generate novel words that are unknown to the translation corpus but may be justified through the abundantly available language model.

Transliteration Mining: In order to create a transliteration phrase-table, we require a transliteration system and to build such a system we need training data, a list of transliteration pairs for Hindi-Urdu. Such data is not readily available. Instead of creating manually hand-crafted mapping rules for Urdu-to-Hindi transliteration as done in Durrani et al. (2010), we induce a transliteration corpus that we can use to train a character-based SMT system. We induced unsupervised transliteration model (Durrani et al., 2014c) adapting the approach of unsupervised transliteration mining described in (Sajjad et al., 2011; Sajjad et al., 2012) for the task of machine translation. The algorithm is based on EM. It takes a list of word pairs and extracts transliteration corpus from it. The mining model is a mixture of two components, a transliteration and a non-transliteration sub-model. The overall idea is that the transliteration model ($p_{tr}(h, u)$) tries to maximize the probability of the transliteration pairs in the word-list and the non-transliteration ($p_{ntr}(h, u)$) component tries to fit the rest of the data.

For a Urdu-Hindi word pair (h, u), the **transliteration model** probability for the word pair is defined as follows:

$$p_{tr}(h, u) = \sum_{a \in Align(h,u)} \prod_{j=1}^{|a|} p(q_j)$$

where $Align(e, f)$ is the set of all possible sequences of character alignments, a is one alignment sequence and q_j is a character alignment. The **non-transliteration model** deals with the word pairs that have no character relationship between them. It is modeled by multiplying source and target character unigram models:

$$p_{ntr}(h, u) = \prod_{i=1}^{|h|} p_H(h_i) \prod_{i=1}^{|u|} p_U(u_i)$$

The probabilities of the two models are refined iteratively to extract a list of transliteration corpus. The model is defined as:

$$p(h, u) = (1 - \lambda)p_{tr}(h, u) + \lambda p_{ntr}(u, u)$$

where λ is the prior probability of non-transliteration.

We initially ran mining over Hindi-Urdu parallel data and were able to extract around 2800 transliteration pairs from a word-list of 17000 pairs. Although a transliteration corpus of this size should be enough to build a transliteration model, note that miner's accuracy is not 100%, because of which we also extract pairs that are not transliterations. To extract more transliterations, we additionally feed the interpolated phrase-table ($p(\bar{u}|\bar{h})$), as described above, to the miner. Surprisingly we were able to mine additional 21K transliteration pairs from a list of 95K word pairs.[2]

Transliteration System: Now that we have transliteration word pairs, we can learn a transliteration model. We segment the training corpus into characters and learn a phrase-based system over character pairs. The transliteration model assumes that source and target characters are generated monotonically.[3] Therefore we do not use any reordering models. We use 4 basic phrase-translation features (direct, inverse phrase-translation, and lexical weighting features), language model feature (built from the target-side of mined transliteration corpus), and word and phrase penalties. The feature weights are tuned[4] on a dev-set of 1000 transliteration pairs.

Transliteration Phrase Table: We transliterate tune and test set and extract 100-best transliteration options for each word. We carry forward the 4 translation model features used in the transliteration system to build a transliteration phrase-table.

[2] Miner only uses word pairs with 1-to-1 alignments because M-N/1-N alignments are less likely to be transliterations.

[3] Mining algorithm also makes this assumption.

[4] Tuning data is subtracted from the training corpus while tuning to avoid over-fitting. After the weights are tuned, we add it back, retrain GIZA, and estimate new models.

System	PT	Tune	Test	System	Tune	Test	System	Tune	Test
$\mathbf{B_{u,h}}$	254K	34.01	34.64						
$\mathbf{\bar{B}_{u,h}}$	254K	34.18	34.79	$\mathbf{\bar{B}_{u,h}T_g}$	37.65	37.58			
$\mathbf{T_g}$	10M	15.60	15.34	$\mathbf{\bar{B}_{u,h}T_r}$	34.77	35.76	$\mathbf{\bar{B}_{u,h}T_gT_r}$	38.0	37.99
$\mathbf{T_r}$		9.54	9.93	$\mathbf{T_gT_r}$	17.63	18.11		Δ+3.89	Δ+3.35

Table 1: Evaluating Triangulated and Transliterated Phrase-Tables in Urdu-to-Hindi SMT – $\mathbf{B_{u,h}}$ = Baseline Phrase Table, $\mathbf{\bar{B}_{u,h}}$ = Modified Baseline Phrase Table, $\mathbf{T_g}$ = Triangulated Phrase Table, $\mathbf{T_r}$ = Transliteration Phrase Table

3.3 Integration into Decoder

We simply integrate the synthesized phrase-tables into phrase-based decoder using the standard log-linear approach (Och and Ney, 2004). We search for a Hindi string H which maximizes a linear combination of feature functions:

$$\hat{H} = \arg\max_{H} \left\{ \sum_{j=1}^{J} \lambda_j h_j(U, H) \right\}$$

where λ_j is the weight associated with the feature $h_j(U, H)$.

Overlapping Translation Options: In the default baseline formulation, the three phrase-tables compete with each other during decoding and create a separate decoding path. In some cases however, phrase-tables might agree on the translation options and hypothesize the same translation, in which case they do not have to compete. To avoid such a scenario, and to reward the common translation options, we modify the baseline phrase-table (created from the Hindi-Urdu parallel data) and edit the probabilities for the translation options that exist in the synthesized triangulated and transliterated phrase-tables. For each phrase (u_i, h_i) that exists in the triangulated or transliterated phrase-table we modify its estimates as the following:

$$p_{\bar{b}}(u_i|h_i) = p_b(u_i|h_i) + p_{tg}(u_i|h_i) + p_{tr}(u_i|h_i)$$
$$\forall(u_i, h_i)|(u_i, h_i) \in p_{tg}(u|h) \vee (u, h) \in p_{tr}(u_i|h_i)$$

where $p_b(u|h)$ is the baseline phrase-table, $p_{tg}(u|h)$ is the triangulated phrase-table and $p_{tr}(u|h)$ is the transliterated phrase-table. This modification slightly improves the performance of the baseline system.

LM-OOV Feature: A lot of the words that the transliteration model produces would be unknown to the language model. To create a bias towards the transliteration options that are known to the language model, we additionally use an LM-OOV feature which counts the number of words in a hypothesis that are unknown to the language model. Language model feature alone can not handle the unknown transliterations, because the smoothing methods such as Kneser-Ney assign significant probability mass to unseen events, thus giving high probability to such unknown transliterations. The LM-OOV feature acts as a prior to penalize such hypotheses. The feature is tuned along with the regular features. Therefore if such transliterations are useful, the optimizer can assign positive weight to this feature. But we noticed that optimizer assigned a high negative weight to this feature, thus heavily penalizing the unknown words.

4 Evaluation

4.1 Data

Our baseline Urdu-to-Hindi system is built using a small EMILLE corpus (Baker et al., 2002) which contain roughly 12000 sentences of Hindi and Urdu sentences which are not exactly parallel. After sentence alignment, we were able to extract a little more than 7000 sentence pairs. The model for Urdu-English data was build using Urdu-English segment of the Indic[5] multi-parallel corpus (Post et al., 2012) which contain roughly 87K sentences. The Hindi-English systems were trained using Hindi-English parallel data (Bojar et al., 2014) composed by compiling several sources including the Hindi-English segment of the Indic parallel corpus. It contains roughly 273K parallel sentences. The tune and test sets for Hindi-Urdu task were created by randomly selecting 1800 sentences from the EMILLE corpus which were then removed from the training data to avoid overfitting. We use half of the selected sentences for tuning and other half for test. The dev and test sets for Hindi-English translation task are the standard sets news-dev2014 and news-test2014 con-

[5] The multi-indic parallel corpus also have Hindi-English segment, but the data is completely disjoint from the Urdu-English segment.

taining 1040 and 2507 sentences respectively. We split news-dev2014 into two halves and used the first half for tuning and second as a test along with the official news-test2014 set. To get more stabilized tuning weights, the tune set is concatenated with Hindi-English dev-set (1400 Sentences) made available with the Hindi-English segment of the Indic parallel corpus. We trained the language model using all the English (287.3M Sentences) and Hindi (43.4M Sentences) monolingual data made available for the 9^{th} Workshop of Statistical Machine Translation.

4.2 Baseline System

Urdu-to-Hindi: We trained a phrase-based Moses system with the following settings: A maximum sentence length of 80, GDFA symmetrization of GIZA++ alignments, an interpolated Kneser-Ney smoothed 5-gram language model with KenLM (Heafield, 2011) used at runtime, 100-best translation options, MBR decoding (Kumar and Byrne, 2004), Cube Pruning (Huang and Chiang, 2007), using a stack size of 1000 during tuning and 5000 during test. We tuned with the k-best batch MIRA (Cherry and Foster, 2012). Because Hindi and Urdu have same grammatical structure, we used a distortion limit of 0 and no reordering.[6]

Hindi-English: For Hindi-English systems, we additionally used hierarchical lexicalized reordering (Galley and Manning, 2008), a 5-gram OSM, (Durrani et al., 2013), sparse lexical and domain features, (Hasler et al., 2012), class-based models (Durrani et al., 2014b), a distortion limit of 6, and the no-reordering-over-punctuation heuristic.

4.3 Experiments

Urdu-to-Hindi: In the initial experiments, we evaluated the effect of integrating the synthesized phrase-tables into Urdu-to-Hindi machine translation. Table 1 shows results on our Urdu-to-Hindi system. Our modification ($\mathbf{\bar{B}_{u,h}}$) to the baseline phrase-table ($\mathbf{B_{u,h}}$) to reward the translation options common between the phrase-tables improve the performance of the baseline system slightly (+0.15). Both triangulated ($\mathbf{T_g}$) and transliterated ($\mathbf{T_r}$) phrase-tables show value when used in isolation, although their performance (BLEU (Papineni et al., 2002)) is a lot worse in comparison to the baseline. Some of this difference in performance can be attributed to the fact that the tune and test sets used for evaluation were extracted from the training data. The real difference of performance can not be studied without a dedicated Urdu-Hindi test set which unfortunately is not available. However, our Hindi-English evaluation shows that this speculation is correct (See below for further discussion). Adding triangulated phrase-table ($\mathbf{\bar{B}_{u,h}T_g}$) to the modified baseline system gives an improvement of +2.79. In comparison, adding transliteration phrase-table ($\mathbf{\bar{B}_{u,h}T_r}$) gives an improvement of +0.97. An overall improvement of +3.35 is observed when all three phrase-tables ($\mathbf{\bar{B}_{u,h}T_gT_r}$) are used together.

Hindi-English: We carried out an extrinsic evaluation to measure the quality of our Urdu-to-Hindi translation systems. We translated the Urdu part of the Urdu-English parallel data using the Urdu-to-Hindi SMT systems described above. We then used the translated corpus to from a synthetic Hindi-English parallel corpus and evaluated its performance by adding it to the baseline Hindi-English systems and in isolation. Table 2 shows the results on adding the synthesized Hindi-English parallel data on top of competitive Hindi-English baseline systems and in isolation. The system ($\mathbf{B_{h,e}D_{\bar{B}_{u,h}T_gT_r}}$) which uses the data generated from our best Urdu-to-Hindi system ($\mathbf{\bar{B}_{u,h}T_gT_r}$) gives an average improvement of +0.35 BLEU points on Hindi-to-English translation task and +1.0 BLEU points on English-to-Hindi. The performance of the system built using synthesized data only ($\mathbf{D_{\bar{B}_{u,h}T_gT_r}}$) is significantly worse than the baseline system on Hindi-to-English task, however the difference is not as much in the other direction. We believe this is still a positive result given the fact that our data is artificially created and is three times smaller than the Hindi-English parallel data used to build the baseline system.

We also synthesized data using the baseline system only trained on the EMILLE corpus ($\mathbf{\bar{B}_{u,h}}$) and using synthesized phrase-tables ($\mathbf{T_gT_r}$) separately. The results in row $\mathbf{\bar{B}_{h,e}D_{B_{u,h}}}$ shows that the data synthesized from the baseline Urdu-Hindi system ($\mathbf{\bar{B}_{u,h}}$) is harmful in both the Hindi-English tasks. In comparison the data synthesized from triangulated and transliterated Urdu-to-Hindi system still showed an average improvement of +0.65 in English-to-Hindi task. No gains were observed in the other direction. Doing an error anal-

[6] Results do not improve with reordering enabled.

System	**Hindi-to-English**		**English-to-Hindi**	
	new-dev$_{14}$	news-test$_{14}$	new-dev$_{14}$	news-test$_{14}$
Baseline ($\mathbf{B_{h,e}}$)	17.11	15.77	11.74	11.57
$\mathbf{B_{h,e}D_{\bar{B}_{u,h}T_gT_r}}$	17.60 Δ+0.49	15.97 Δ+0.20	12.83 Δ+1.09	12.47 Δ+0.90
$\mathbf{D_{\bar{B}_{u,h}T_gT_r}}$	13.13 Δ-3.98	10.96 Δ-4.79	11.14 Δ-0.60	10.51 Δ-1.06
$\mathbf{B_{h,e}D_{\bar{B}_{u,h}}}$	16.91 Δ-0.20	15.39 Δ-0.36	10.63 Δ-1.11	9.87 Δ-1.7
$\mathbf{B_{h,e}D_{T_gT_r}}$	17.15 Δ+0.04	15.84 Δ+0.10	12.47 Δ+0.73	12.13 Δ+0.56

Table 2: Evaluating Synthesized Hindi-English Parallel Data on Standard Translation Task – $\mathbf{D_{\bar{B}_{u,h}T_gT_r}}$ = System using data synthesized from the best Urdu-to-Hindi System that additionally use Triangulated and Transliterated Phrase Tables

ysis we found that the baseline Urdu-Hindi system suffers from data sparsity. The number of out-of-vocabulary tokens when translating the Urdu corpus using baseline phrase-table were 310K. In comparison the number of words unknown to the interpolated phrase-table were 50K and these were translated using in-decoding transliteration method (Durrani et al., 2014c).[7]

5 Conclusion

In this paper we applied a combination of pivoting and transliteration to improve Urdu→Hindi⇄English machine translation using a two-way approach. First we use the Urdu-English and English-Hindi phrase-tables to induce a Urdu-to-Hindi translation model. We then use the resultant model to synthesize additional Hindi-English parallel data. Both transliteration and triangulated phrase-tables showed improvements over the baseline system. A cumulative improvement of +3.35 BLEU points was obtained using both in tandem. The artificially induced parallel data gives an improvement of +0.35 for Hindi-to-English and +1.0 for English-to-Hindi over a competitive baseline system. Our English-to-Hindi system was ranked highest for EN-HI and third for HI-EN in this year's WMT translation task (Durrani et al., 2014a).

Acknowledgements

The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreements n ° 287658 (EU-Bridge) and n ° 287688 (MateCat).

[7] Note that we also applied transliteration for the words that were known to the interpolated phrase-table.

References

Al-Onaizan, Yaser and Kevin Knight. 2002. Translating named entities using monolingual and bilingual resources. In *Proceedings of ACL*, pages 400–408.

Baker, Paul, Andrew Hardie, Tony McEnery, Hamish Cunningham, and Robert J. Gaizauskas. 2002. EMILLE, a 67-million word corpus of indic languages: Data collection, mark-up and harmonisation. In *LREC*.

Bertoldi, Nicola, Madalina Barbaiani, Marcello Federico, and Roldano Cattoni. 2008. Phrase-based statistical machine translation with pivot languages. *Proceeding of IWSLT*, pages 143–149.

Bojar, Ondřej, Vojtěch Diatka, Pavel Rychlý, Pavel Straňák, Aleš Tamchyna, and Dan Zeman. 2014. Hindi-english and hindi-only corpus for machine translation. In *Proceedings of LREC'14*, Reykjavik, Iceland, May.

Callison-Burch, Chris, Philipp Koehn, and Miles Osborne. 2006. Improved statistical machine translation using paraphrases. In *Proceedings of the HLT-NAACL'06*, pages 17–24, New York City, USA, June.

Cherry, Colin and George Foster. 2012. Batch tuning strategies for statistical machine translation. In *NAACL'12*, pages 427–436, Montréal, Canada, June.

Cohn, Trevor and Mirella Lapata. 2007. Machine translation by triangulation: Making effective use of multi-parallel corpora. In *Proceedings of the ACL'07*, pages 728–735, Prague, Czech Republic, June.

de Gispert, Adrià and José B. Mariño. 2006. Statistical machine translation without parallel corpus: Bridging through spanish. In *Proceedings of LREC 5th Workshop on Strategies for developing Machine Translation for Minority Languages*, pages 65–68.

Durrani, Nadir, Hassan Sajjad, Alexander Fraser, and Helmut Schmid. 2010. Hindi-to-urdu machine translation through transliteration. In *Proceedings of ACL'10*, pages 465–474, Uppsala, Sweden, July.

Durrani, Nadir, Alexander Fraser, Helmut Schmid, Hieu Hoang, and Philipp Koehn. 2013. Can Markov Models Over Minimal Translation Units Help Phrase-Based SMT? In *Proceedings of ACL'13*, Sofia, Bulgaria, August.

Durrani, Nadir, Barry Haddow, Philipp Koehn, and Kenneth Heafield. 2014a. Edinburgh's phrase-based machine translation systems for WMT-14. In *Proceedings of WMT'14*, Baltimore, MD, USA, June.

Durrani, Nadir, Philipp Koehn, Helmut Schmid, and Alexander Fraser. 2014b. Investigating the usefulness of generalized word representations in SMT. In *Proceedings of COLING'14*, Dublin, Ireland, August. To Appear.

Durrani, Nadir, Hassan Sajjad, Hieu Hoang, and Philipp Koehn. 2014c. Integrating an unsupervised transliteration model into statistical machine translation. In *Proceedings of EACL 2014*, Gothenburg, Sweden, April.

Galley, Michel and Christopher D. Manning. 2008. A simple and effective hierarchical phrase reordering model. In *Proceedings of EMNLP'08*, pages 848–856, Honolulu, Hawaii, October.

Galuscáková, Petra and Ondrej Bojar. 2012. Improving smt by using parallel data of a closely related language. In *Baltic HLT*, pages 58–65.

Haque, Rejwanul, Sergio Penkale, Jie Jiang, and Andy Way. 2012. Source-side suffix stripping for bengali-to-english smt. In *IALP*, pages 193–196.

Hasler, Eva, Barry Haddow, and Philipp Koehn. 2012. Sparse lexicalised features and topic adaptation for smt. In *Proceedings of IWSLT'12*, pages 268–275.

Heafield, Kenneth. 2011. Kenlm: Faster and smaller language model queries. In *Proceedings of WMT'11*, pages 187–197, Edinburgh, Scotland, United Kingdom, July.

Hermjakob, Ulf, Kevin Knight, and Hal Daumé III. 2008. Name translation in statistical machine translation - learning when to transliterate. In *Proceedings of ACL'08*, Columbus, Ohio.

Huang, Liang and David Chiang. 2007. Forest rescoring: Faster decoding with integrated language models. In *Proceedings of ACL'07*, pages 144–151, Prague, Czech Republic, June.

Kumar, Shankar and William J. Byrne. 2004. Minimum bayes-risk decoding for statistical machine translation. In *Proceedings HLT-NAACL'04*, pages 169–176.

Nakov, Preslav and Jörg Tiedemann. 2012. Combining word-level and character-level models for machine translation between closely-related languages. In *Proceedings of ACL'13 (Volume 2: Short Papers)*, pages 301–305, Jeju Island, Korea, July.

Niessen, Sonja and Hermann Ney. 2004. Statistical Machine Translation with Scarce Resources using Morpho-Syntactic Information. *Computational Linguistics*, 30(2):181–204.

Och, Franz J. and Hermann Ney. 2004. The Alignment Template Approach to Statistical Machine Translation. *Computational Linguistics*, 30(1):417–449.

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: A method for automatic evaluation of machine translation. In *Proceedings of ACL'02*, pages 311–318, Morristown, NJ, USA.

Paul, Michael, Hirofumi Yamamoto, Eiichiro Sumita, and Satoshi Nakamura. 2009. On the importance of pivot language selection for statistical machine translation. In *Proceedings of HLT-NAACL'09 (Short Papers)*, pages 221–224.

Popović, M. and H. Ney. 2004. Towards the use of word stems and suffixes for statistical machine translation. In *Proceedings of COLING*, Geneva, Switzerland.

Post, Matt, Chris Callison-Burch, and Miles Osborne. 2012. Constructing parallel corpora for six indian languages via crowdsourcing. In *Proceedings of WMT'12*, pages 401–409, Montréal, Canada, June.

Resnik, Philip and Noah A. Smith. 2003. The Web As a Parallel Corpus. *Comput. Linguist.*, 29(3):349–380, September.

Sajjad, Hassan, Nadir Durrani, Helmut Schmid, and Alexander Fraser. 2011. Comparing two techniques for learning transliteration models using a parallel corpus. In *Proceedings of IJCNLP'11*, pages 129–137, Chiang Mai, Thailand, November.

Sajjad, Hassan, Alexander Fraser, and Helmut Schmid. 2012. A statistical model for unsupervised and semi-supervised transliteration mining. In *Proceedings of ACL'12*, Jeju, Korea.

Utiyama, Masao and Hitoshi Isahara. 2003. Reliable measures for aligning japanese-english news articles and sentences. In *Proceedings of ACL'03*, pages 72–79, Sapporo, Japan, July.

Utiyama, Masao and Hitoshi Isahara. 2007. A comparison of pivot methods for phrase-based statistical machine translation. In *Proceedings of NAACL'07*, pages 484–491.

Wu, Hua and Haifeng Wang. 2007. Pivot language approach for phrase-based statistical machine translation. In *Proceedings of ACL'07*, pages 856–863, Prague, Czech Republic, June.

Zhao, Bing, Nguyen Bach, Ian Lane, and Stephan Vogel. 2007. A log-linear block transliteration model based on bi-stream hmms. In *Proceedings of HLT-NAACL'07*, Rochester, New York.

Oral Session 2
User Papers

SEECAT: ASR & Eye-Tracking Enabled Computer-Assisted Translation

Mercedes García-Martínez[1], Karan Singla[2], Aniruddha Tammewar[2], Bartolomé Mesa-Lao[1], Ankita Thakur[3], Anusuya M.A.[4], Srinivas Bangalore[5], Michael Carl[1]

[1]CRITT - Copenhagen Business School, Denmark, [2]IIIT Hyderabad, India, [3]JSSATE, Noida, India
[4]SJCE, Mysore,India, [5]AT&T Research Labs, United States

Abstract

Typing has traditionally been the only input method used by human translators working with computer-assisted translation (CAT) tools. However, speech is a natural communication channel for humans and, in principle, it should be faster and easier than typing from a keyboard. This contribution investigates the integration of automatic speech recognition (ASR) in a CAT workbench testing its real use by human translators while post-editing machine translation (MT) outputs. This paper also explores the use of MT combined with ASR in order to improve recognition accuracy in a workbench integrating eye-tracking functionalities to collect process-oriented information about translators' performance.

1 Introduction

Human-aided machine translation is gradually becoming a common practice for language service providers (LSPs) as opposed to machine-aided human translation. Depending on the nature of the text, more and more LSPs pre-translate the source text using existing translation memories (TMs) and then automatically translate the remaining text using an MT engine. Then human translators correct and adapt, i.e. post-edit, the output from both TMs and MT to produce different levels of translation quality. Improving and maximizing the potentials of a post-editing workbench is thus one of the priorities set by both the industry and the research community (Mesa-Lao, 2012). The motivation behind this paper comes from a desire to know how different input modalities in a computer-assisted translation (CAT) workbench can be of greater support to translation professionals.

Keyboards are the most widely used input device for text production and they seem to be the easiest input method when only minor changes are needed. However, in the context of post-editing, when the text requires major changes (e.g. editing larger segments of text), typing could be optimized using other input modalities. Moreover, if the post-editor is not a touch typist, then she has to switch visual attention back and forth between the screen and the keyboard making the task more complex. A possible solution for this profile of users could be the use of other input methods, such as ASR or hand-writing, in addition to traditional typing (Hauptmann and Rudnicky, 1990).

The comparison between ASR and typing as input methods can be done based on task duration, i.e. measuring the time needed to type against the ASR rate including possible corrections to fix recognition inaccuracies. Studies on input durations have shown that ASR input can be faster (Chen, 2006; Vidal et al., 2006).

This paper is structured along the following lines: The first section presents the CasMaCat[1]

[1] CasMaCat: Cognitive Analysis and Statistical Methods for Advanced Computer Aided Translation. Project co-funded by the European Union under the Seventh Framework Programme Project 287576 (FP7 ICT-2011.4.2). URL: http://casmacat.eu. Demo: http://casmacat.prhlt.upv.es/mail-demo/askdemo.php

workbench as an introduction to the SEECAT[2] project. A description follows on how new ASR modules for English, Spanish and Hindi have been added to the SEECAT workbench. The last section presents the experimental data collected in two pilot studies with human translators performing a series of tasks using ASR and keyboard as input methods.

2 Background: The CASMACAT workbench

The CASMACAT (Alabau et al., 2014a) project aims at developing the next-generation translation workbench to improve productivity, quality, and work practices in the translation industry. The current CASMACAT prototype (version 2) allows users to upload documents and work with a first MT draft for post-editing. In its current implementation the workbench only supports keyboard and mouse as input modes, but it will also support e-pen(Alabau et al., 2014b) in the next prototype. A diagram of the major components of the CASMACAT workbench is shown in Figure 1.

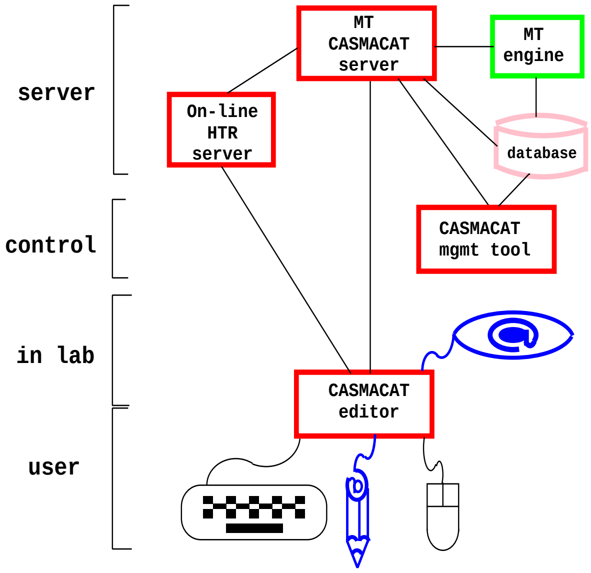


Figure 1: Major components of the CASMACAT workbench v.3

2.1 Machine translation server

The machine translation server converts the source text (in the form of XLIFF files) into a target text. The output is provided by the MATECAT (Bertoldi et al., 2012) component. This server works in parallel with Translation Memories (TMs) to retrieve the data from the translation server. TMs are basically a repository of previously translated segments. During the translation process, the translation server queries a TM to search for exact or fuzzy matches of the current source segment and these matches are then proposed to the translator as translation suggestions. When no matches are found in the TM, suggestions from the MT engine are supplied to the translator.

2.2 The editor

The CASMACAT editor is a web-based client with configurable visualization options for interactive translation prediction and interactive editing. The editor has several interfaces to communicate with a remote MT system via the CASMACAT MT server and it will be able to interface with an e-pen (Alabau et al., 2014b) in the next prototype. The editor features logging functions to record translator's keystrokes and mouse clicks as well as gaze activity captured by an eye-tracking device (i.e. EyeLink 1000).

Taking the CASMACAT workbench as a starting point, the SEECAT project aimed at testing the potential of speech recognition for translator-computer interaction. A description of the SEECAT workbench is provided in the next section.

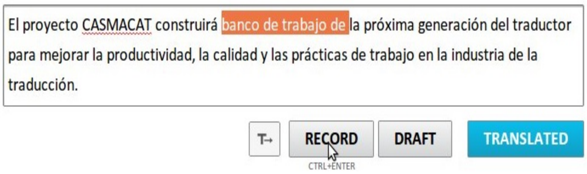
3 The SEECAT workbench

The main aim of the SEECAT (Speech & Eye-Tracking Enabled Computer-Assisted Translation) project was to provide ASR as an input method for post-editing MT using the GUI of CASMACAT. The SEECAT workbench is able to recognize speech in English, Hindi and Spanish from any user without previous training.

User interaction is triggered after pressing the record button in the GUI to dictate text. The text to be replaced in the editor has to be selected before pressing the record button. The audio signal is then sent to the SEECAT server and the recognized text is sent back to the GUI. An example is shown in the figure 2.

Figure 3 shows the communication architecture between the client, the browser/GUI, the audio plug-in, the SEECAT server and the ASR server. It shows how the integration process is done with the server, and the client through the GUI. "Click

[2] SEECAT Project: Speech & Eye-Tracking Enabled Computer Assisted Translation. URL: http://bridge.cbs.dk/platform/?q=SEECAT



El proyecto CASMACAT construirá banco de trabajo de la próxima generación del traductor para mejorar la productividad, la calidad y las prácticas de trabajo en la industria de la traducción.

Figure 2: Recording functions in the SEECAT GUI.

RECORD” stands for clicking the record button to capture the speech signal, and ”Click STOP” stands for clicking the stop button to stop the recording.

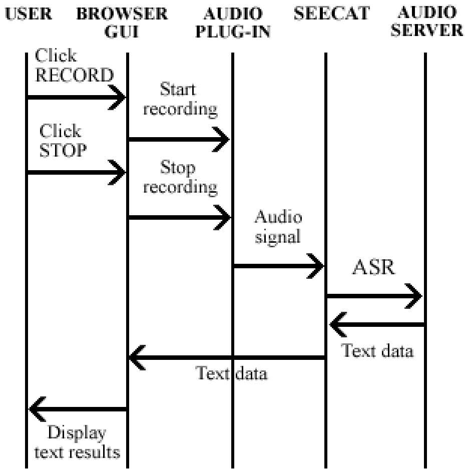


Figure 3: Case diagram for the interaction between browser, audio plug-in and server.

The data flow diagram proposed in the SEECAT project can be represented at a higher level as shown in Figure 4. The left part of the Figure 4 (CASMACAT) is represented in more detail in the previous Figure 1 and the right part of the Figure 4 (SEECAT) is described in more detail in Figure 5. CASMACAT sends the cursor position plus the recorded audio file to the SEECAT server. The SEECAT server sends back the ASR transcription.

3.1 Automatic Speech Recognition (ASR)

In SEECAT, AT&T Watson Toolkit has been trained for ASR in three languages, namely English, Hindi and Spanish.

3.1.1 English and Spanish

The English and Spanish ASR systems were provided by AT&T Labs-Research. Table 1 shows the details about the data. The data was recorded by female native speakers of the language.

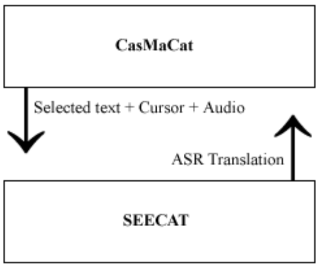


Figure 4: High level diagram of the proposed system.

Data Statistics	EN-ES	
	EN	ES
#sentences	7,792,118	7,792,118
#words	98,347,681	111,006,109
Vocabulary	501,450	516,906

Table 1: English and Spanish ASR data in SEECAT.

3.1.2 Hindi

The Hindi ASR was trained on more than 20 hours of audio data (7k training sentences) with transcriptions. The data was collected from various sources as described in table 2. The training details of Hindi ASR can be found in a previous work (Pandey et al., 2013).

Contributed By	**Domain**
KIIT	General text messages
McGill University	News
IIIT Hyderabad	Wikipedia Articles
SEECAT workshop	Text messages, Tourism

Table 2: Hindi ASR training data in SEECAT

For a test set of 67 sentences of a general domain, the recognition accuracy for Hindi ASR was 69.0.

3.2 SEECAT modules

SEECAT captures the speech signal using the WebRTC API(Bergkvist et al., 2012) (Web Real time communication) from the browser. WebRTC API is a browser API that enables browser to browser applications for voice calling/streaming, video streaming and peer-to-peer file sharing in real time communication without plug-ins. We have used this API for audio data. SEECAT uses SoX conversion for down sampling the speech signal from 16 khz to 8 khz. This signal is passed to the ASR server for the recognition of the speech signal uttered by the user.

The CasMaCat logging functions have been extended with the information coming from ASR in order to be able to check when the ASR input starts and finishes.

Figure 5 describes how the SEECAT components interact.

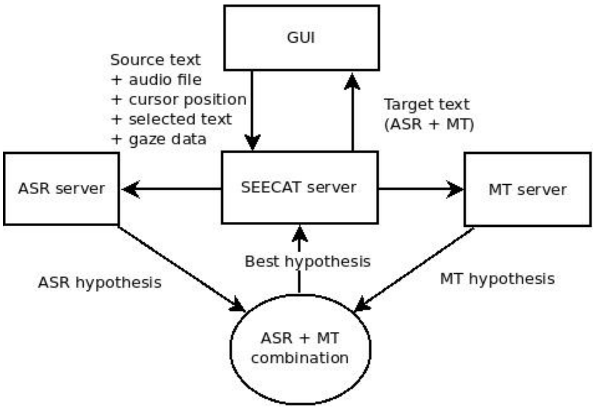


Figure 5: SEECAT architecture.

The SEECAT server receives from the GUI: The source text, an audio file, the cursor position, the selected text (optionally) and the gaze data whenever an eye-tracker device is connected. In order to improve the accuracy of the ASR module, the ASR in SEECAT is trained following previous work in this field (Paulik et al., 2005a) and (Paulik et al., 2005b). As described in Figure 5, in a post-editing task the ASR n-best hypotheses are generated using the audio file provided by the GUI to the SEECAT server. Also, the MT hypotheses are generated for the source string. These MT hypotheses are used to rescore ASR hypotheses. The experiments related to ASR MT integration are described in the following section.

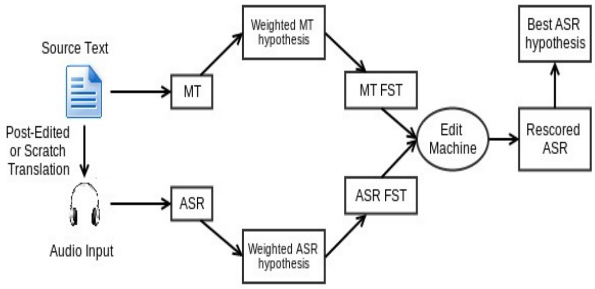


Figure 6: SEECAT combination of MT+ASR for better recognition.

In the future, we would also like to improve the translation output combining gaze data with ASR+MT output. The data from the eye-tracker will not only provide information about the translation process, but will also help to improve the output provided by the MT server base in gaze information coming from the user.

4 Experiments and results

This section presents experimental data using the current version of the SEECAT workbench.

4.1 Integration of ASR and MT

MT can improve ASR (Khadivi et al., 2006; Lecouteux et al., 2006) in a computer-assisted translation scenario. The same technique used to improve ASR through MT can be used with semantic information (Tammewar et al., 2013). In SEECAT, the hypotheses produced by ASR and MT are converted into lattices and are then composed using Edit Machine with the help of OpenFst toolkit (Allauzen et al., 2007). The synset information from WordNet is used while composing for the semantic matching of words. According to the edit distance scores, ASR hypotheses are rescored. We further extend this approach for the two language pairs Hindi-English and Spanish English, where the target language is English along with incorporating semantic information from English WordNet (Miller, 1995).

4.1.1 Experiments

In the Hindi-English MT system, it was found that the translated sentences were very poor and hence the POS tagger could not assign correct POS tags to the words. So we modified the technique to merge the senses not only from the predicted POS category but from all the four POS categories. This way the wrong POS tag will not affect the sense selection. Then this technique was also extended to Spanish-English system. This approach reduced the processing time, as now the POS tagger is not needed and time complexity is a very important factor in a real-time system such as SEECAT.

4.1.2 Results

For the evaluation, we used a test dataset of 132 sentences for Hindi-English and 96 sentences for Spanish-English. Table 3 enumerates the results for various experiments. Overall the word accuracy increased by *3.4%* for Hindi-English and *2.0%* for Spanish-English system over the baseline ASR. We performed the integration taking MT hypotheses as sequence (Seq.) of words and unweighted (Unw.) bag of words and found that the latter strategy performs better (Tammewar et al., 2013).

Experiment		Language Pair		
		Hin-Eng		Span-Eng
POS		Yes	No	No
Only ASR		68.3		79.1
ASR+MT	Seq.	68.7		80.2
	Unw.	69.7		80.3
ASR+MT+Synset	Seq.	71.1	70.7	80.8
	Unw.	71.4	71.7	81.1

Table 3: ASR Word Accuracy in SEECAT.

There was not much difference in ASR word accuracy for experiments with POS and without POS tag, so we performed experiments without POS information for Spanish-English as the system performs faster in a real translation task without assigning POS tag for each word in the hypothesis.

4.2 Integration of ASR and Gaze

An eye-tracker plug-in has been integrated to the SEECAT interface to collect gaze information while a human translator interacts with the workbench. In a previous work (Kulkarni et al., 2013) was provided information on the use of gaze data to map gaze fixations to source words to improve ASR. For the integration, lattices were created and composed using the same ASR-MT composition.

Experiments showed that ASR as weighted bag-of-words and gaze as unweighted bag-of-words improved by 4.6% word accuracy in ASR for the English-Hindi pair.

4.3 Post-editing typing and using ASR

In this section the results of a pre-pilot and a pilot study assessing the potential of integrating ASR in a post-editing workbench are presented.

4.3.1 Pre-pilot test

Two native Spanish speakers volunteered to interact with the SEECAT workbench across the following six tasks:

- Task 1: Translation from scratch through typing (only using keyboard interaction)
- Task 2: Translation from scratch through ASR (only using speech interaction)
- Task 3: Post-editing through typing (only using keyboard interaction)
- Task 4: Post-editing through ASR (only using speech interaction)
- Task 5: Translation from scratch through typing + ASR
- Task 6: Post-editing through typing + ASR

Participant 1 was a professional translator while participant 2 did not have previous experience in translation. In each of these six tasks, the two participants worked from English into Spanish and the text domain involved in the experiments was tourism (the domain for which the ASR had previously been trained). Time to complete the task was considered as the dependent variable in order to measure the productivity gains derived from incorporating ASR as an input method for both translation from scratch and post-editing of MT.

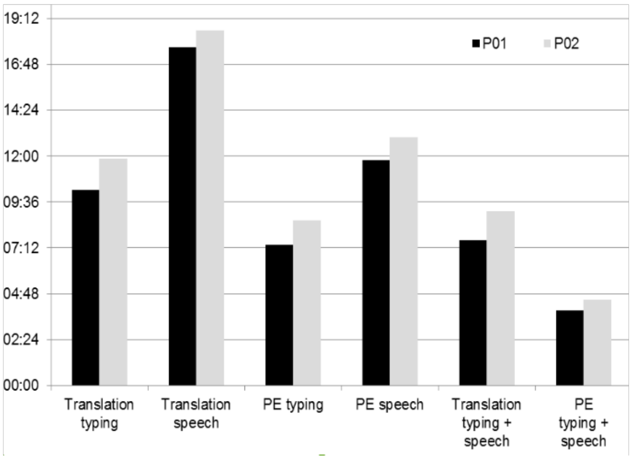


Figure 7: Time in minutes per tasks for participants 1 and 2.

Figure 7 shows overall times per task for the two participants in this pre-pilot test.

These preliminary results show that combining ASR and typing in post-editing tasks can produce faster turnaround when considering the task time overall as opposed to just providing ASR or typing as input method for the same tasks.

When looking at the time spent across individual segments in each of the six tasks for the two participants (see Figures 8 and 9), it can be seen that the majority of segments with the fastest turnaround belong to the post-editing task combining both ASR and typing.

In Figures 8 and 9, it is observed that the combination of ASR and typing requires the shortest time when compared to other tasks. This combination of input methods could still benefit from enhancing the ASR module adding new vocabulary and new domains.

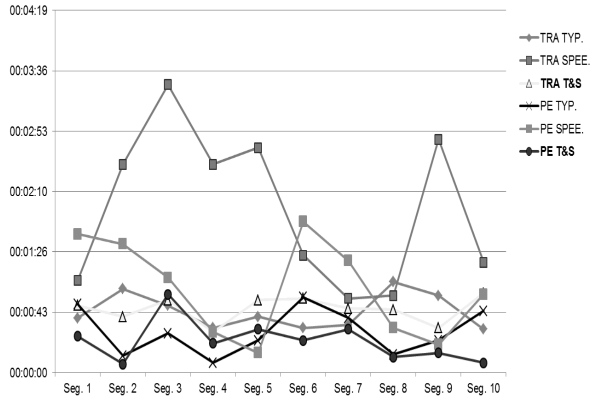


Figure 8: Time in minutes across segments and tasks for participant 1.

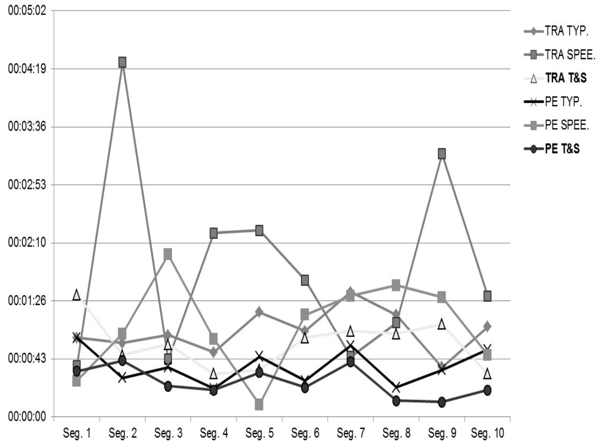


Figure 9: Time in minutes across segments and tasks for participant 2.

The results of this pre-pilot test encouraged the pilot test reported in the next section, where only post-editing tasks were included.

4.3.2 Pilot test

A group of 10 professional translators (7 women and 3 men) aged between 24 and 32 volunteered to perform the evaluation of the SEECAT workbench described in section 3. All participants had a degree in translation studies and were regular users of computer-aided translation tools (mainly SDL Trados and Déjà Vu X2). None of them had ever used ASR technology, but 90% of them claimed to have previous experience in post-editing MT as a professional service.

The pilot text involved two different texts (T1 and T2), of ten segments each, in the following two tasks:

1. Post-editing through typing (only using keyboard interaction)

2. Post-editing through typing + ASR

Task and text order were counterbalanced across participants. The language pair involved was English into Spanish and the text domain was tourism, the domain for which the ASR was trained. Following the design tested in the pre-pilot, this pilot study involved time to complete the task as the dependent variable and the input methods used while post-editing as the two independent variables, i.e. *i*) only typing or *ii*) typing and dictating (ASR).

Looking at the overall time spent to complete the task across participants (see Figure 10), 6 out of 10 benefited from integrating ASR as an input method, being able to complete the task faster than only typing. Participants P02, P03, P08 and P09 needed more time to complete the task when working with ASR. These four participants are also the ones who registered a greater time difference when comparing both tasks (up to an extra time-span of 4 minutes between task 1 and task 2 in the case of P02). There are no big time differences between the two tasks for the rest of the participants (12:30 minutes on average for the task involving only keyboard and 13:02 minutes for the task involving keyboard and ASR).

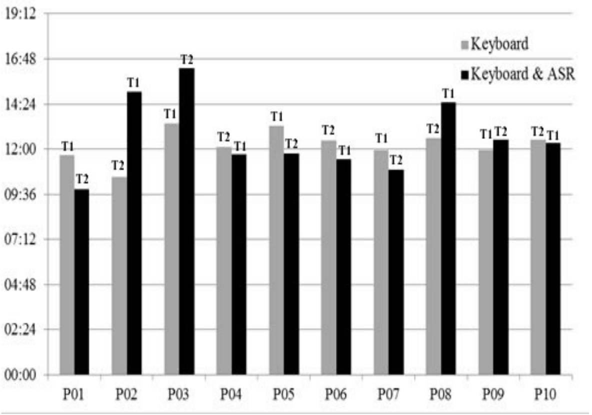


Figure 10: Time in minutes across tasks and texts using SEECAT.

When asked to provide feedback about the experimental tasks in a retrospective interview, all participants stated that ASR seems to be a promising feature for a CAT workbench, but they all also underpinned that they would need more time to get acquainted with this technology in the context of post-editing.

5 Conclusions and future enhancements

As a result of the SEECAT project, ASR has been integrated to a computer-assisted translation tool as an additional input method. From these preliminary experiments, it seems reasonable to assume that working both with ASR and typing in post-editing tasks can be of help to boost translators' productivity. More experiments with a larger sample will have to be run in order to further explore the benefits of multimodal interaction both in translation and post-editing tasks. In addition, lab experiments showing that ASR can benefit from MT and semantic information for better re-scoring of ASR hypotheses have been presented.

Since WebRTC API has been used, future investigations will explore possibilities for online audio streaming of the data making the events synchronous rather than asynchronous. By doing this, we want to minimize the delay while the user receives the response from the system.

Future enhancements are foreseen integrating interactive machine translation and hand-written recognition using e-pen for the benefit of the human translator. More experiments in the context of professional translation over a longer period of time will be done to measure if productivity results increase after more hours of interaction with the workbench.

Acknowledgments

Work supported by the European Union 7$^{\text{th}}$ Framework Program (FP7/2007-2013) under the CASMACAT project (grants agreement n$^{\text{o}}$ 287576), Copenhagen Business School, JSS Mahavidyapeetha and AT&T Labs-Research.

References

Alabau, Vicent, Christian Buck, Michael Carl, Francisco Casacuberta, Mercedes García-Martínez, Ulrich Germann, Jesús González-Rubio, Robin Hill, Philipp Koehn, Luis Leiva, Bartolomé Mesa-Lao, Daniel Ortiz-Martínez, Hervé Saint-Amand, Germán Sanchis-Trilles, and Chiara Tsoukala. 2014a. Casmacat: A computer-assisted translation workbench. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*. Software demonstrations.

Alabau, Vicent, Alberto Sanchis, and Francisco Casacuberta. 2014b. Improving on-line handwritten recognition in interactive machine translation. *Pattern Recognition*, 47(3):1217–1228.

Allauzen, Cyril, Michael Riley, Johan Schalkwyk, Wojciech Skut, and Mehryar Mohri. 2007. Openfst: A general and efficient weighted finite-state transducer library. In *Implementation and Application of Automata*, pages 11–23. Springer.

Bergkvist, Adam, Daniel C Burnett, Cullen Jennings, and Anant Narayanan. 2012. Webrtc 1.0: Real-time communication between browsers. *Working draft, W3C*.

Bertoldi, Nicola, Alessandro Cattelan, and Marcello Federico. 2012. Machine translation enhanced computer assisted translation. First report on lab and field tests. Available from: http://www.matecat.com/wp-content/uploads/2013/01/MateCat-D5.3-V1.2-1.pdf.

Chen, Fang. 2006. *Designing human interface in speech technology*. Springer.

Hauptmann, Alexander G and Alexander I Rudnicky. 1990. A comparison of speech and typed input. In *Proceedings of the Speech and Natural Language Workshop*, pages 219–224.

Khadivi, Shahram, Richard Zens, and Hermann Ney. 2006. Integration of speech to computer-assisted translation using finite-state automata. In *Proceedings of the COLING/ACL on Main conference poster sessions*, pages 467–474. Association for Computational Linguistics.

Kulkarni, Rucha, Kritika Jain, Himanshu Bansal, Srinivas Bangalore, and Michael Carl. 2013. Mutual disambiguation of eye gaze and speech for sight translation and reading. In *Proceedings of the 6th workshop on Eye gaze in intelligent human machine interaction: gaze in multimodal interaction*, pages 35–40. ACM.

Lecouteux, Benjamin, Georges Linares, Pascal Nocéra, and Jean-François Bonastre. 2006. Imperfect transcript driven speech recognition. In *InterSpeech*.

Mesa-Lao, Bartolomé. 2012. The next generation translator's workbench: post-editing in casmacat v. 1.0. In *Proceedings of the 34th Translating and the Computer Conference, ASLIB*.

Miller, George A. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Pandey, Dipti, Tapabrata Mondal, SS Agrawal, and Srinivas Bangalore. 2013. Development and suitability of indian languages speech database for building watson based asr system. *Corpus*, 41282(90140):67522.

Paulik, Matthias, Christian Fügen, Sebastian Stüker, Tanja Schultz, Thomas Schaaf, and Alex Waibel. 2005a. Document driven machine translation enhanced asr. In *INTERSPEECH*, pages 2261–2264. Citeseer.

Paulik, Matthias, S Stuker, C Fugen, Tanja Schultz, Thomas Schaaf, and Alex Waibel. 2005b. Speech translation enhanced automatic speech recognition. In *Automatic Speech Recognition and Understanding, 2005 IEEE Workshop on*, pages 121–126. IEEE.

Tammewar, Aniruddha, Karan Singla, Srinivas Banglore, and Michael Carl. 2013. Enhancing asr by mt using semantic information from hindiwordnet.

Vidal, Enrique, Francisco Casacuberta, Luis Rodriguez, Jorge Civera, and Carlos D Martnez Hinarejos. 2006. Computer-assisted translation using speech recognition. *Audio, Speech, and Language Processing, IEEE Transactions on*, 14(3):941–951.

Moses SMT as an Aid to Translators in the Production Process

Falko Schaefer
SAP AG / Dietmar-Hopp-Allee 16, 69190 Walldorf, Germany
falko.schaefer@sap.com

Joeri Van de Walle
CrossLang N.V. / W. Wilson-plein 7, Gent, Belgium
joeri@crosslang.com

Joachim Van den Bogaert
CrossLang N.V. / W. Wilson-plein 7, Gent, Belgium
joachim@crosslang.com

Abstract

SAP has been heavily involved in the implementation and deployment of machine translation (MT) within the company since the early 1990s. In 2013, SAP initiated an extensive proof of concept project, based on the statistical MT system Moses (Koehn, et al., 2007), in collaboration with the external implementation partner CrossLang. The project focused on the use of Moses SMT as an aid to translators in the production process. This paper describes the outcome of the productivity evaluation for technical documents pertaining to SAP's Rapid Deployment Solutions (RDS), which was performed as part of this proof of concept project.

1 Background and Project Description

The use of machine translation at SAP dates back to the early 1990s. Originally the rule-based approach was deployed mainly for the translation of technical troubleshooting documents (SAP Notes), test cases, documentation, training materials, and as a gist translation tool for customer messages. MT systems used were METAL (German-English/English-German) and Logos (English-French mainly), followed by the next generation system Lucy LT (for these same languages). In 2012, SAP started experimenting with statistical machine translation (SMT). A prototype system was built at SAP Language Services (SLS) for the Chinese-to-English and English-to-Chinese language pairs. This prototype was based on the Moses SMT technology. In 2013, SLS initiated a more extensive proof of concept project, again based on Moses, in collaboration with the external implementation partner CrossLang. The project focused on the use of Moses SMT as an aid to translators in the production process. In that context CrossLang developed a plugin for SDL Trados Studio, thus enabling a seamless integration of Moses SMT into the SDL Trados Studio environment. MT suggestions were provided to translators during the proof of concept projects in addition to translation memory (TM) segments, which translators were free to accept, edit or discard just as they would TM matches. The overall timeline for the project was rather ambitious as all project phases (MT engine development, piloting, evaluation and engine improvement) had to be run between July and December 2013. In 2014, the SLS MT team will take additional steps to align machine translation landscapes and further extend the MT offering to various usage scenarios and more content types.

The proof of concept projects were carried out for two different content types: sap.com and RDS (Rapid Deployment Solutions) texts. While sap.com materials are typically texts used for SAP's official website, RDS texts are technical documents related to SAP's RDS product offering. Consequently the former content type can be classified as being of a more creative nature and thus more marketing-like than the latter, which is more technical by nature and hence more similar to documentation. The present paper will focus on the RDS content type.

The language scope of the proof of concept phase comprised the eight target languages Chinese, French, German, Italian, Japanese, Portuguese (Brazil), Russian and Spanish with source language English as well as the respective reverse language directions. However, the evaluations subject to this paper were carried out only for the target languages Chinese, French, German, and Russian.

For each language pair and content type Moses engines were built in three iterations:

- Iteration 1: Engines built with content type-specific data only (in-domain engines)

- Iteration 2: Engines built with a combination of content type-specific data and general SAP-related data to which domain adaptation techniques were applied (in-domain engines + domain adaptation)
- Iteration 3: Systems built in iteration 2 enhanced with natural language processing (NLP) components and techniques (in-domain engines + domain adaptation + NLP)

The size of the training data sets used for the relevant engines ranged from approximately 1 million to nearly 2 million tokens for sap.com and from roughly 2.2 million to 5.5 million tokens for the RDS content type.

While a total of more than 150 Moses engines were built throughout the project phase, not all of them could be run through the human evaluation rounds due to time and budget constraints. Instead, the best-performing systems for each content type and language pair were selected for human evaluation.

2 Evaluation Setup

In the proof of concept project, we looked at the machine translation output from different perspectives, which is reflected in the various types of evaluation that were performed: (i) engine development progress (automatic evaluation), (ii) translation quality (adequacy and fluency evaluation), (iii) translation productivity increase potential (productivity evaluation), and (iv) translation process (pilot projects).

The main goal of the automatic evaluations was to measure development progress. At the beginning of the project different test sets, each consisting of 1000 sentences, were extracted per content type and per language pair. Those test sets were held out of the training data and were used to score the engines after each development iteration. Three metrics were used for scoring: BLEU (Papineni, Roukos, Ward, & Zhu, 2002), METEOR (Banerjee & Lavie, 2005), and TER (Snover, Dorr, Schwartz, Micciulla, & Makhoul, 2006).

As automatic evaluation metrics are known to not always be reliable indicators of users' appreciation of the machine translation output quality, the automatic assessments were complemented with human judgments. With the adequacy & fluency evaluations, the focus was on the linguistic quality of the translations. With this evaluation we tried to answer the question 'how good is the translation?'. Two aspects of the translation were assessed: (i) in how far did the machine translation system succeed in transferring the meaning of the source sentence (adequacy), and (ii) in how far did the machine translation output respect the formal rules of the target languages (fluency). For both of these aspects, informants rated 400 machine translated sentences on a scale of 1 to 5, where 1 equals very poor performance and 5 excellent performance. For this evaluation, informants were linguists (professional and experienced translators). As with the automatic evaluations, sentences used for evaluation were kept apart from the training data.

With the productivity evaluation we tried to assess to what extent productivity increases might be obtained by using automated translation as an aid to speed up human translation. To evaluate this, informants were given a mix of (i) sentences pre-translated with MT, (ii) sentences without translation suggestion, and (iii) sentences with translations taken from the TM. The main reason for including the latter type of segments was to get an indication of the informants' possible bias against MT. Informants were asked to review and correct the translation of those sentences for which a translation was provided (MT or TM), and to come up with a translation from scratch for those sentences for which no translation was provided. In the background, the time informants spent on editing the translation output or translating the sentence was recorded. Recorded times were then used to calculate the average throughput for sentences in each of the categories (MT post-editing, TM match review, and translation from scratch). The sentences used in this evaluation were the same as those used for the adequacy and fluency evaluations and the informants taking part in this evaluation were provided by SAP's regular translation vendors.

When it comes to incorporating MT into the translation production process, a common concern is that the use of MT will negatively influence the quality of the translation output. The main objective of the pilot projects, finally, was to assess whether end-users of the translations would effectively notice quality differences between translations produced as the result of post-editing MT output and translations produced the traditional way. At the same time, the pilot projects served as a means to assess the complexity of integrating MT into the existing translation processes at SAP. To evaluate these aspects, MT was integrated into a real translation project, namely the translation of an update of existing contents for both content types in the pilot project. Sets of about 400 sentences per language were processed by SAP's regular translation vendors in two ways: once with a translation suggestion from MT and once as translation from scratch. The resulting translation variants were then compared and ranked by SAP employees in the target language countries.

3 Evaluation Results

Because of the space constraints for this paper, we will limit the discussion of the evaluation results to the adequacy/fluency evaluations and productivity evaluations of one particular content type, i.e. RDS. We discuss these results per evaluation type.

3.1 Adequacy/Fluency Evaluations

Table 1 shows the average ratings across informants for adequacy and fluency for all evaluated language pairs and the difference between the average adequacy and fluency ratings.

	En-to-De	**En-to-Fr**	**En-to-Ru**	**En-to-Zh**
Adequacy	4.11	4.16	3.54	3.89
Fluency	3.77	3.73	3.35	3.81
Difference	0.34	0.43	0.19	0.07

Table 1: RDS Adequacy/Fluency Results

Table 1 shows that the adequacy and fluency ratings vary per language, with the German and French output scoring best in terms of adequacy and the German and Chinese output scoring best as far as fluency is concerned. The lowest scores, both for adequacy and fluency, were observed for Russian.

The biggest difference between the adequacy and fluency rating was noted for the French output; the smallest difference for the Chinese MT translations.

3.2 Productivity Evaluations

Tables 2 through 5 show, for the four language pairs that were evaluated, the throughput per category per informant (in words per hour) and the productivity increase that is obtained by comparing the throughput for post-editing against that for translation.

	Translation	**MT Post-editing**	**Full match revision**	**Productivity increase**
Informant 1	288	358	458	24%
Informant 2	206	341	388	66%
Informant 3	347	599	1023	73%
Average	**280**	**433**	**623**	**54%**

Table 2: RDS Productivity Results En-to-De

For the English-to-German language pair, we observed an average productivity increase of 54% across informants for the RDS content type. A striking observation regarding the productivity evaluation for this content type is that, on average, 48% of the exact matches that were included in the evaluation set were changed by the informants. Informant 2 changed as much as 63% of the segments (i.e. 25 out of 40), which explains why his throughput for full match review is relatively lower than that of the other two informants.

	Translation	**MT Post-editing**	**Full match revision**	**Productivity increase**
Informant 1	531	906	1340	70%
Informant 2	451	628	617	39%
Informant 3	328	712	932	117%
Average	**437**	**749**	**963**	**76%**

Table 3: RDS Productivity Results En-to-Fr

For the English-to-French language pair, we observed an average productivity increase of 76% across informants. Looking at the results more closely, we found that there are considerable differences between the results of the different informants. For this content type, there is a difference of 78 percentage points between the increase noted for informant 3 (117%) and that noted for informant 2 (39%). We found evidence of the potential productivity gains in the fact that on average 40% of the segments with machine translation output remained unchanged. The lower increase noted with informant 2 might be explained by this informant having a more critical attitude towards MT. This becomes apparent when looking at the change rate for full match review segments. Informant 2 changed 58% (i.e. 23 out of 40) of the exact matches from TM as opposed to 50% for informant 1 and 35% for informant 3.

	Translation	**MT Post-editing**	**Full match revision**	**Productivity increase**
Informant 1	335	534	1541	59%
Informant 2	951	1833	4608	93%
Informant 3	296	443	889	50%
Average	**527**	**936**	**2346**	**67%**

Table 4: RDS Productivity Results En-to-Ru

For the English-to-Russian language pair, we observed an average productivity increase of 67% across informants. Although the average increase might be a little inflated by the high increase reported for informant 2, the fact that on average 41% of the sentences in the evaluation set was left unchanged by the informants, provides a good basis for explaining the observed increases. Interesting to see is that, compared to the languages already discussed, the Russian informants were less tempted to change full matches (on average only 28% were changed as opposed to 48% for both German and French).

	Translation	**MT Post-editing**	**Full match revision**	**Productivity increase**
Informant 1	266	333	453	25%
Informant 2	325	473	739	46%
Informant 3	264	312	420	18%
Average	**285**	**373**	**537**	**30%**

Table 5: RDS Productivity Results En-to-Zh

For the English-to-Chinese language pair, we observed an average productivity increase of 30% across informants. Again, we found that informants were very much inclined to change full match segments: on average 49% of the segments got changed. Whereas the change rate in the full match review category for the English-to-German and English-to-French languages pairs were found to be similar, the degree of change was a lot higher for the English-to-Chinese language pair (similarity score for full match review of 81.80) than for the language pairs discussed above (94.14 for English-to-German and 90.47 for English-to-French). This suggests that either the informants were very "picky" or that there was a problem with the quality of the TM. Further investigation revealed that the latter was the case.

4 Conclusions

The overall results of the quality evaluation as measured in the adequacy and fluency assessment appear rather encouraging across language pairs with a distribution of average scores between 3.35 and 4.16. There were, however, noticeable differences between individual languages with German and French scoring particularly well and Russian performing comparatively poorly in that part of the evaluation program. Apart from differing quality levels between the MT engines built for the various language pairs the fact that there is always an element of subjectivity involved in human quality judgments may serve as an explanation for this observation.

Besides the assessment of quality perception, another important question addressed in the evaluation rounds was obviously whether MT actually speeds up translation in the production process. As could be seen in the previous section, this question requires a differentiated answer depending on the target language, the main reasons for this being not only the varying quality levels of the engines for each language but also the fact that cultural aspects may impact the acceptance and hence perceived usefulness of machine translation as a translation aid. This became particularly apparent in the evaluation rounds for the target languages Russian and Chinese, where results of human quality and productivity evaluation were somewhat contradictory. However, this does not substantially affect the overall trend revealed by the productivity evaluation, which did prove clear productivity gains for all languages. This observation was confirmed by the translation vs. post-editing comparison in the pilot project evaluation (not discussed in this paper), which showed that the use of MT did not seem to have a negative impact on the quality of the final translations. As such translations produced with the help of MT were in no instance rated as being of lower quality than translations done from scratch. In fact quite the contrary was observed: For all languages a clear preference for translations resulting from MT plus post-editing could be established. This could be explained by the technical nature of RDS contents, where adequacy and fluency are considered more important than style and hence informants were less inclined to edit the MT output.

Finally it needs to be stressed that the evaluation results presented in this paper only reflect a snapshot of the quality of the engines built at the point in time the pilot and evaluation projects were conducted. Detailed system improvement activities are currently underway at SAP in order to further optimize MT engines and reach the defined quality levels for the various MT usage scenarios in the company.

References

Banerjee, S., & Lavie, A. (2005). METEOR: an automatic metric for MT evaluation with improved correlation with human judgments. *ACL-2005: Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization* (pp. 65-72). Ann Arbor: University of Michigan.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., . . . Zens, R. (2007). Moses: Open source toolkit for statistical machine translation. *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions* (pp. 177-180). Association for Computational Linguistics.

Papineni, K., Roukos, S., Ward, T., & Zhu, W.-j. (2002). BLEU: a Method for Automatic Evaluation of Machine Translation. *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics* (pp. 311-318). Philadelphia, Pennsylvania: Association for Computational Linguistics.

Snover, M., Dorr, B., Schwartz, R., Micciulla, L., & Makhoul, J. (2006). A study of translation edit rate with targeted human annotation. *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas*, (pp. 223-231). Massachusetts.

Assumptions, Expectations and Outliers in Post-Editing

Laura Casanellas
Welocalize / Dublin, Ireland
laura.casanellas@welocalize.com

Lena Marg
Welocalize / London
lena.marg@welocalize.com

Abstract

As a multilingual vendor, we have access to machine translation (MT) scoring and other evaluation data on a wide range of language combinations and content types; we also have experience with different MT systems in production. Our daily work involves the collaboration with a wide spectrum of translation partners, from very MT-savvy to novices in this area. Being exposed to MT in such a varied and large-scale setup, we would like to share some of our insights into assumptions, expectations and outliers observed with regard to MT quality, productivity and suitability with a particular focus on the challenges that (individual) post-editor behavior presents in this context. Our observations are based on data correlations carried out at the end of 2013 from a database that contains all evaluation data produced during this year, as well as recent surveys with some of our very MT-savvy translation partners for deeper, locale-specific insights.

1 Introduction

In our company machine translation (MT) is typically integrated in the translation workflow as a productivity tool complementing translation memories, glossaries etc., with translators carrying out the required levels of post-editing. Content is translated into a multitude of languages (mostly from English) and MT is currently being used in production on a wide range of content types, from technical communication, user interface and corporate communication to user-generated contents. Additionally, we do not work with a specific MT system, but rather a variety of MT systems are evaluated and used – based on our own or our client's recommendations. At the end of 2013, we created a comprehensive database of results from automatic and human scorings of MT output as well as results from productivity tests obtained in that year, covering all these variables (locales, content types, MT systems).

Our productivity test shows the potential productivity gains obtained by moving from the task of translating to post-editing.

While the analysis and correlations drawn from this database confirmed certain assumptions, it allowed us to reassess expectations and also provided insights into outliers. In this abstract and our presentation, we would like to discuss these assumptions, expectations and outliers, benefitting from the wide range of variables used in the company. In this context, we want to draw attention to individual translator behavior, which might need to be considered more strongly when assessing MT output quality and usability.

2 The Database

The database mentioned above was created with all available data related to MT evaluation from 2013. The timeframe was delimited to one year.

Objectives for creating the database were multiple, but a key aim was to see if a correlation of currently available, internal data would help us make productivity predictions on future MT post-editing effort with the metrics currently in use in the company.

The categories included in the 2013 database are: client name, content type, locale, translation partner carrying out any human evaluations, BLEU, PE distance, human adequacy & fluency scores, productivity test deltas (in percent), productivity test throughputs (words post-edited versus words translated), MT system provider, owner of MT system maintenance (e.g. client, provider or Welocalize), comment on whether the test resource had received training on MT and PE, final quality scores (i.e.: the final translated / post-edited product).

2.1 Data Correlations

After populating the database with data on all the above categories, we started looking into correlations between different variables, e.g.: Adequacy versus BLEU, Fluency versus PE distance, Adequacy versus productivity delta, Fluency versus productivity delta, etc., using ***Pearson's r***. At this stage, we *intentionally* tried to keep the data sets broad, e.g. include all locales that had partaken in a given productivity test, rather than limit it to a few; include a range of MT systems rather than focusing only on one; including all post-editor profiles, rather than distinguishing between experienced and novice. To some extend the idea was precisely not to start with assumptions from the outset (like "engine X will anyway perform better than engine Y for Russian", "your translators are more open-minded to MT and will perform better" etc.). We wanted to see whether trends would emerge at a high level - trends that could be useful for us to dig into deeper in future or to exploit more with regard to productivity predictions for instance. This approach is further justified by the fact that our MT programs tend to cover various languages and content types and MT systems are often defined by the client, who would typically only invest into one MT system, unless this system offers only limited language pairs. In other words: MT systems are not only chosen on the basis of what the general assumption of their performance is, but also for cost and maintenance reasons.

Some assumptions were certainly confirmed by the data correlation. For instance the Adequacy score proved to be more strongly correlated to productivity deltas and the Fluency score to PE distance.

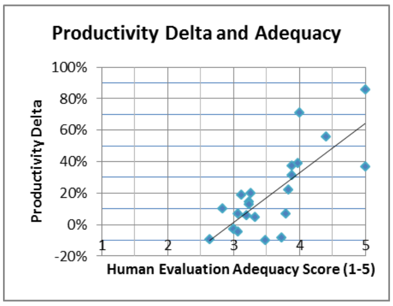


Fig 1 Productivity and Adequacy across all locales with a cumulative Pearson's r of 0.71, a very strong correlation

We find these correlations meaningful, as the final productivity tests are measured against our standard Quality Metrics and requirements for the respective content. For example, if Fluency scores and productivity delta do not correlate strongly, this suggests that post-editing changes required to improve fluency have less impact on productivity. Since post-editors frequently dismiss MT and post-editing for Fluency issues (word order, word from agreements…), it is highly relevant for our daily work around educating the supply chain.

2.2 Assumptions confirmed

As mentioned, the Adequacy score showed a strong correlation with the productivity delta and gives us an indication of the type of post-editing effort required for the particular program. On the other hand, we found a strong negative correlation between BLEU and PE distance, providing evidence that automatic scores alone cannot be relied upon as a sole indication of the quality of raw MT output.

Among all our language groups, Romance languages render the highest productivity rates. In relation to content, user assistance produces the best productivity rates when publishable quality standards are required. Content types with lower final (i.e. after post-editing) quality expectations like UGC, have even higher productivity gains.

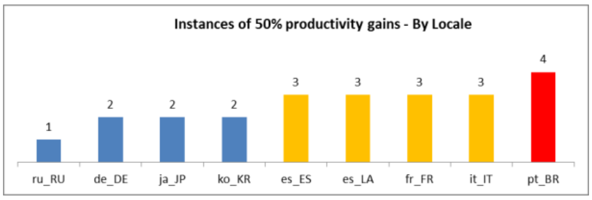


Fig. 2 Instances of productivity gains over 50% by locale, the numbers reflect the quantity of tests that received a score over 50%

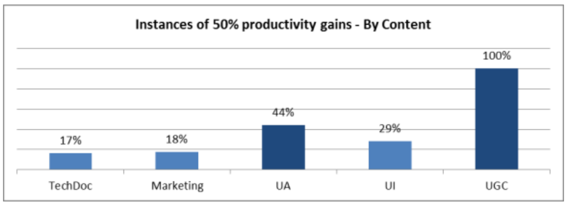


Fig 3 Instances of productivity gains over 50% by content type, the numbers reflect what percentage of tests received a score over 50%

Finally, we could not link negative productivity to a specific content type, even though a traditionally difficult type like marketing was among the content types contained on the dataset.

2.3 Outliers

Throughout the analysis, we observed some results that did not align with our expectations. These findings were particularly interesting to us and we want to focus on them in our presentation, as they give insights into post-editor behavior, variation in input methods, truth and myths regarding best performing languages for MT, etc. The term *outlier* in that sense is here not to be understood as "data to be ignored", but quite on the contrary, "data to take note of".[1]

For instance within the group of the above mentioned romance languages, there were still noticeable differences. While Brazilian Portuguese topped the raw MT quality assessments and productivity throughputs (irrespective of the underlying MT system or content type), results for French were a lot less consistent and generally lower.

3 Individual Productivity Influencers

Before talking about variations in individual productivity gains from MT in post-editing, it is important to point out that Adequacy & Fluency scoring exercises, when carried out by several speakers of the same locale on the same content, tend to lead to similar results. Of course, here, too, there is some individual variation, but overall scores tend to move in similar directions, confirming the scores and trends of the other evaluators.

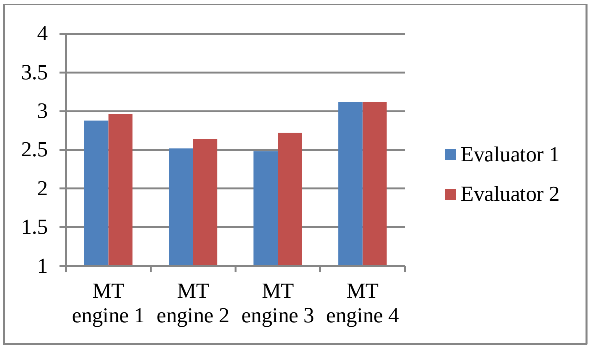


Fig. 4 Accuracy scores of two German evaluators for four different MT engines, using identical sample content. Despite minor variation, trends are the same.

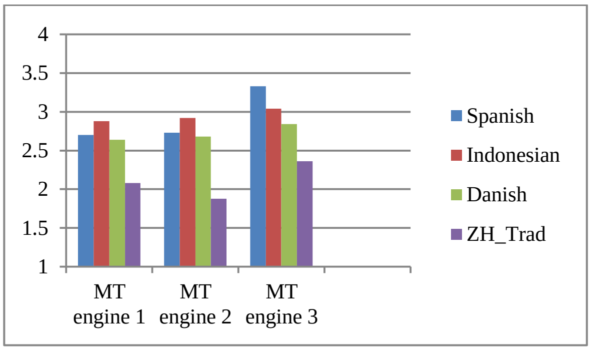


Fig. 5 Human Evaluation scores by evaluators of four different languages for three different systems. The content sample was identical

With productivity gains in productivity tests, however, we see strong variation from one translator to the next. Although some content types do lend themselves better to MT, the correlations were not as clear-cut or within our own expectations (see Marketing earlier on). Language pairs are expected to yield different results with MT, but as the Brazilian / French example shows, are not a sole explanation.

Earlier papers have called out factors such as translators' experience and technical skills (Guerberof, 2009; Almeida and O'Brien 2010). Verleysen (2013) also mentions translators' working methods in the European Commission's Newsletter. While experience and technical skills probably play a part one way or another, they do not as yet show to be consistent factors in our data. Working methods strike us as very interesting and relevant, as the case in 3.1 further suggests.

For some languages (e.g. Romance), trends are more uniform, for others (e.g. German, Russian, Japanese, Korean, as mentioned later on) they vary greatly, making it difficult at times to establish a fair average of what could be the expected productivity gains for this content and language.

With the aim of learning from individual behaviors and predicting future productivity gains, we ask ourselves two questions:

- *What circumstances or variables most reliably facilitate good-quality, highly productive post editing?*
- *Do conditions and parameters outside the post-editor's control facilitate or hamper his or her success?*

In our analysis of over a hundred cases we noticed that the deviations between individuals are very significant, especially when it comes to MT

[1] We should note here that outliers caused by corrupted data, faulty results, errors in human annotation etc. had been discarded from the database from the outset.

post-editing. It is tempting to assume that the increase between HT and MT is progressive and that every individual improves their performance when they change from translation to post-editing. The reality is not that simple; not all translators benefit from MT output in the same manner and some do not benefit at all.

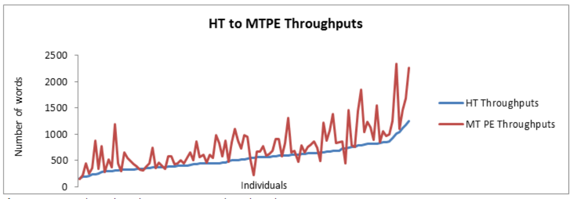


Fig. 6 Translators benefit from MT output in a different way.

In terms of productivity gains, two groups in particular are interesting:

1. Individuals who gain 50% productivity or higher when they move from translation to post-editing.
2. Individuals whose *translation* throughputs are well above the average. We focus on translators who produce 600 or more words per hour.

Our initial analysis has shed some light on potential common characteristics among the first group:

- Language combination: English into Romance languages. Note: Above 50% productivity gains were also seen for Russian, German, Japanese and Korean, but Romance languages (with some internal variation) are still showing higher productivity gains and more consistently so.
- Content type: User Assistance.

But what about the other individuals, the ones who outperform in translation, the ones who can translate at a pace well above the average? Are they able to gain good productivity gains when moving onto the task of post-editing or is there something like a "plateau" in terms of daily individual throughputs? Do they share common characteristics? These are questions we want to further investigate and share first insights at the summit.

Another group of whose translation behavior particularly caught our interest are the English into Japanese translators.

3.1 The Japanese case

Japanese continuously proves to be one of the most challenging locales for MTPE programs, not only with regard to achieving raw MT output of a good quality level.

Through our evaluations and working with a range of translation partners for this locale, we discovered a few aspects how Japanese translators as a group deviate from other languages (e.g. often no formal translation training, very different translation volumes on specific programs compared to FIGS for instance,…) that could potentially influence post-editing productivity. The one that intrigued us most relates to Input Method Editors (IME): it appears that Japanese translators always use some form of IME when working in CAT tools. Some of these IMEs are more elaborate than others, and also some translators are savvier in making best use of them than others. While they certainly have an impact on translation speed, the impact on post-editing speed is not entirely clear to us at this stage, but it is possible that good skills around IME contribute more to productivity for Japanese than MT does.

4 Conclusion

An exhaustive correlation of MT evaluation data was carried out across a wide range of locales, content types and MT systems at our company on 2013. The initial analysis of correlations and data confirmed certain assumptions, but also highlighted the complexity around MT quality and predicting productivity gains, especially with regard to individual translators' behavior.

With regard to translator behavior, there are two areas in particular we would like to analyse further through extensive surveys, in order to share results at the summit: firstly, those translators that already have above average throughputs for translation – how, if, do they benefit from MT? Secondly, IME for Japanese translators: what tools and options are available, what are different levels of sophistication, how are people using them etc., always with a focus on potential impact on post-editing.

Acknowledgement

The authors would like to acknowledge the support of Olga Beregovaya, Andy Mallett, Elaine O'Curran, David Clarke, Karen McLaughlin and Louise Quinn.

References

Guerberof, Ana. 2009. *Productivity and quality in MT post-editing*. MT Summit XII - Workshop: Beyond Translation Memories: New Tools for Translators MT, Ottawa, Ontario, Canada.

De Almeida, Giselle and O'Brien, Sharon. 2010. *Analysing Post-Editing Performance: Correlations with Years of Translation Experience*. Proceedings of the EAMT Summit at St. Raphael.

Verleysen, Piet et al. 2013. MT@Work Conference: by practitioners for practitioners. *European Commission Languages and Translation Newsletter*. Issue #6 on Machine Translation. 6-9.

Oral Session 3
Research Papers

Document-level translation quality estimation: exploring discourse and pseudo-references

Carolina Scarton and **Lucia Specia**
Department of Computer Science
University of Sheffield
S1 4DP, UK
{c.scarton,l.specia}@sheffield.ac.uk

Abstract

Predicting the quality of machine translations is a challenging topic. Quality estimation (QE) of translations is based on features of the source and target texts (without the need for human references), and on supervised machine learning methods to build prediction models. Engineering well-performing features is therefore crucial in QE modelling. Several features have been used so far, but they tend to explore very short contexts within sentence boundaries. In addition, most work has targeted sentence-level quality prediction. In this paper, we focus on document-level QE using novel discursive features, as well as exploiting pseudo-reference translations. Experiments with features extracted from pseudo-references led to the best results, but the discursive features also proved promising.

1 Introduction

The purpose of machine translation (MT) **quality estimation (QE)** is to provide a quality prediction for new, unseen machine translated texts, without relying on reference translations (Blatz et al., 2004; Specia et al., 2009; Bojar et al., 2013). This task is usually addressed with machine learning models trained on datasets composed of source texts, their machine translations, and a quality label assigned by humans or by an automatic metric (e.g.: BLEU (Papineni et al., 2002)). A common use of quality predictions is the estimation of post-editing effort in order to decide whether to translate a text from scratch or post-edit its machine translation. Another use is the ranking of translations in order to select the best text from multiple MT systems.

Feature engineering is an important component in QE. Although several feature sets have already been explored, most approaches focus on sentence-level quality prediction, with sentence-level features. These disregard document structure or wider contexts beyond sentence boundaries. To the best of our knowledge, only Rubino et al. (2013) considered discourse-related information by studying topic model features for sentence-level prediction. Soricut and Echihabi (2010) explored document-level quality prediction, but they did not use explicit discourse information, e.g. information to capture text cohesion or coherence.

In this paper we focus on **document-level features** and **document-level prediction**. We believe that judgements on translation quality depend on units longer than just a given sentence, taking into account discourse phenomena for lexical choice, consistency, style and connectives, among others (Carpuat and Simard, 2012). This is particularly important in MT evaluation contexts, since most MT systems, and in particular statistical MT (SMT) systems, process sentences one by one, in isolation. Our hypothesis is that features that capture **discourse phenomena** can improve document-level prediction. We consider two families of features that have been successfully applied in reference-based MT evaluation (Wong and Kit, 2012) and readability assessment (Graesser et al., 2004). In terms of applications, document-level QE is very important in scenarios where the entire text needs to be used/published without post-edition.

Soricut and Echihabi (2010) and Soricut and Narsale (2012) explored a feature based on **pseudo-references** for document-level QE. Pseudo-references are translations produced by one or more external MT systems, which are different from the one producing the translations we want to predict the quality for. These are used as references against which the output of the MT system of interest can be compared using standard metrics such as BLEU. Soricut et al. (2012) and Shah et al. (2013) explored pseudo-references for sentence-level QE. In both cases, features based on pseudo-references led to significant improvements in prediction accuracy. Here we also use pseudo-references for document-level QE, with a number of string similarity metrics to produce document-level scores as features, which are arguably more reliable than sentence-level scores, particularly for metrics like BLEU.

In the remainder of this paper, Section 2 presents related work. Section 3 introduces the document-level QE features we propose. Section 4 describes the experimental setup of this work. Section 5 presents the results.

2 Related work

Work related to this research includes document-level MT evaluation metrics, QE features, and QE prediction, as well as work focusing on other linguistic features, and work using pseudo-references.

Wong and Kit (2012) use lexical cohesion metrics for MT evaluation at document-level. Lexical cohesion relates to word choices, captured in their work by reiteration and collocation. Words and stems were used for reiteration, and synonyms, near-synonyms and superordinates, for collocations. These metrics are integrated with traditional metrics like BLEU, TER (Snover et al., 2006) and METEOR (Banerjee and Lavie, 2005). The highest correlation against human assessments was found for the combination of METEOR and the discursive features.

Rubino et al. (2013) explore topic model features for QE at sentence-level. Latent Dirichlet Allocation is used to model the topics in two ways: a bilingual view, where the bilingual corpus is concatenated at sentence-level to build a single model with two languages; and a polylingual view, where one topic model is built for each language. While the topics models are generated with information from the entire corpus, the features are extracted at sentence-level. These are computed for both source and target languages using vector distance metrics between the words in these sentences and the topic distributions. Topic model features has been achieved promising results.

Soricut and Echihabi (2010) explore document-level QE prediction to rank documents translated by a given MT system. Features included BLEU scores based on pseudo-references from an off-the-shelf MT system, for both the target and the source languages. The use of pseudo-references has been shown to improve state-of-the-art results. Soricut and Narsale (2012) also consider document-level prediction for ranking, proposing the aggregation of sentence-level features for document-level prediction. The authors claim that a pseudo-references-based feature (based in BLEU) is one of the most powerful in the framework. For QE at sentence-level, Soricut et al. (2012) use BLEU based on pseudo-references combined with other features to build the best QE system of the WMT12 QE shared task.[1] Shah et al. (2013) conduct a feature analysis, at sentence-level, on a number of datasets and show that the BLEU-based pseudo-reference feature contributes the most to prediction performance.

In terms of other types of linguistic features for QE, Xiong et al. (2010) and Bach et al. (2011) propose features for word-level QE and show that these improve over the state-of-the-art results. At sentence-level, Avramidis et al. (2011), Hardmeier (2011) and Almaghout and Specia (2013) consider syntactic features, achieving better results compared to competitive feature sets. Pighin and Màrquez (2011) obtain improvements over strong baselines from exploiting semantic role labelling to score MT outputs at sentence-level. Felice and Specia (2012) introduce several linguistic features for QE at sentence-level. These did not show improvement over shallower features, but feature selection analysis showed that linguistic features were among the best performing ones.

3 Features for document-level QE

QE is traditionally done at sentence-level. This happens mainly because the majority of MT systems translate texts at this level. Evaluating sentences instead of documents can be useful for many scenarios, e.g., post-editing effort prediction.

[1] `http://www.statmt.org/wmt12/`

However, some linguistic phenomena can only be captured by considering the document as a whole. Moreover, for scenarios in which post-edition is not possible, e.g., gisting, quality predictions for the entire documents are more useful.

Several features have been proposed for QE at sentence-level. Many of them can be directly used at document-level (e.g., number of words in source/target sentences). However, other features that better explore the document as a whole or discourse-related phenomena can bring additional information. In this paper, discourse information is explored in two ways: lexical cohesion (Section 3.1) and LSA cohesion (Section 3.2). The intuition behind using cohesion features for QE is the following: on the source side, documents that have low cohesion are likely to result in bad quality translations. On the target side, documents with low cohesion are likely to have low overall quality.

From the feature set proposed in (Soricut and Echihabi, 2010) for document-level ranking of MT system outputs, text-based and language model-based features are also covered by the baseline features used in this paper. Pseudo-reference-based features are also addressed herein (Section 3.3). The example-based features cannot be easily reproduced since we do not have access to additional documents to use as development set (our parallel corpora are already small). The training data-based features were not considered because we use MT systems that do not have or make their training sets available.

3.1 Lexical cohesion features

Our first set of features is based on lexical cohesion metrics (hereafter, **LC**). Lexical cohesion is related to word choices in a text (Wong and Kit, 2012). Words can be repeated to make the relation among sentences more explicit to the reader. Another phenomenon of lexical cohesion is the use of synonyms, hypernyms, antonyms, etc. In this paper, we only consider word repetitions as features. These are features that can be easily extracted for languages other than English, for which a thesaurus with synonyms, hypernyms, etc., may not be available. Our LC features are as follows:

Average word repetition: for each content word, we count its frequency in all sentences of the document. Then, we sum the repetition counts and divide it by the total number of content words in the document. This is computed for the source and target documents, resulting in two features.

Average lemma repetition: the same as above, but the words are first lemmatised.

Average noun repetition: the same as above, but only nouns are considered as words.

3.2 LSA cohesion features

General textual quality is often connected to the notion of readability of a text. Readability can be measured in many ways, focusing on different aspects such as coherence, cohesion, how accessible a text is to a certain audience, etc. The Coh-Metrix project[2] (Graesser et al., 2004) has proposed a number of text readability metrics. Latent Semantic Analysis (**LSA**) (Landauer et al., 1998) is used in order to extract cohesion-related features. This is a statistical method based on Singular Vector Decomposition (SVD) and is often aimed at dimensionality reduction. In SVD, a given matrix X can be decomposed into the product of three other matrices:

$$X = WSP^T,$$

where W describes the original row entities as vectors of derived orthogonal factor values; S is a diagonal matrix containing scaling values and P (P^T is the transpose of P) is the same as W but for columns. When these three matrices are multiplied, the exact X matrix is recovered. The dimensionality reduction consists in reconstructing the X matrix by only using the highest values of the diagonal matrix S. For example, a dimensionality reduction of order two will consider only the two highest values of S.

The X matrix (rows x columns) can be built from words by sentences, words by documents, sentences by documents, etc. In the case of words by sentences (which we use in our experiments), each cell contains the frequency of a given word in a given sentence. LSA was originally designed to be used with large corpora of multiple documents. In our case, since we are interested in measuring cohesion within documents, we compute LSA for each individual document through a matrix of words by sentences within the document.

LSA was computed using a package for python,[3] which takes word stems and sentences to build the matrix. Usually, before applying SVD in

[2] http://cohmetrix.com/

[3] https://github.com/josephwilk/semanticpy

LSA, the X matrix is transformed wherefore each cell encapsulates information about a word's importance in a sentence or a word's importance in the text in general. Landauer et al. (1998) suggest the use of TF-IDF transformation for that. However, we disregarded the use of TF-IDF as this transformation would smooth out the values of high frequency words across sentences. In our case, the salience of words in sentences is important.

Our LSA features follow from Graesser et al. (2004)'s work on readability assessment:

LSA adjacent sentences: for each sentence in a document, we compute the Spearman rank correlation coefficient of its word vector with the word vectors of its immediate neighbours (sentences which appear immediately before and after the given sentence). For sentences with two neighbours (most cases), we average the correlation values. After that, we average the values for all sentences in order to have a single figure for the entire document.

LSA all sentences: for each sentence in a document, we calculate the Spearman rank correlation coefficient of the word vectors between this sentence and all the others. Again we average the values for all sentences in the document.

Higher correlation scores are expected to correspond to higher text cohesion, since the correlation among the sentences in a document is related to how close the words in the document are (Graesser et al., 2004). Different from lexical cohesion features, LSA features are able to find correlations among different words, which are not repetitions and may not be synonyms, but are instead related (as given by co-occurrence patterns).

3.3 Pseudo-references

Pseudo-references are translations produced by other MT systems than the system we want to predict the quality for. They are used as references to evaluate the output of the MT system of interest. They have also been used for other purposes, e.g., to fulfil the lack of human references available in reference-based MT evaluation (Albrecht and Hwa, 2008) and automatic summary evaluation (Louis and Nenkova, 2013). The application we are interested in, originally proposed in (Soricut and Echihabi, 2010), is to generate features for QE. In this scenario, reference-based evaluation metrics (such as BLEU) are computed between the MT system output and the pseudo-references and used to train quality prediction models.

Soricut and Echihabi (2010) discussed the importance of the pseudo-references being generated by MT system(s) which are as different as possible from the MT system of interest, and preferably of much better quality. This should ensure that string similarity features (like BLEU) indicate more than simple consensus between similar MT systems, which would produce the same (possibly bad quality) translations, e.g., Google Translate[4].

4 Experimental settings

Although QE is traditionally trained on datasets with human labels for quality (such as HTER – Human Translation Error Rate (Snover et al., 2006)), no large enough dataset with human-based quality labels assigned at document-level is available. Therefore, we resort to predicting automatic metrics as quality labels, as in (Soricut and Echihabi, 2010). This requires references (human) translations at training time, when the automatic metrics are computed, but not at test time, when the automatic metrics are predicted.

Corpora Two parallel corpora with reference translations are used in our experiments: FAPESP and WMT13. **FAPESP** contains $2,823$ English-Brazilian Portuguese (EN-BP) documents extracted from a scientific Brazilian news journal (FAPESP)[5] (Aziz and Specia, 2011). Each article covers one particular scientific news topic. The corpus was randomly divided into 60% ($1,694$ documents) for training a baseline **MOSES**[6] statistical MT system (Koehn et al., 2007) (with 20 documents as development set); and 40% ($1,128$ documents) for testing the SMT system, which generated translations for QE training (60%: 677 documents) and test (40%: 451 documents). In addition, two external MT systems were used to translate the test set: **SYSTRAN**[7] – a rule-based system – and Google Translate (**GOOGLE**), a statistical system.

WMT13 contains English-Spanish (**EN-ES**) and Spanish-English (**ES-EN**) translations from

[4] http://translate.google.com.br/

[5] http://revistapesquisa.fapesp.br

[6] http://www.statmt.org/moses/?n=moses.baseline

[7] http://www.systransoft.com/

the test set of the translation shared task of WMT13.[8] In total, 52 source documents were available for each language pair. In order to build the QE systems, the outputs of all MT systems submitted to the shared task were taken: 18 systems for EN-ES (528 documents for QE training, and 356 for QE test), and 17 systems for ES-EN (500 documents for QE training, and 332 documents for QE test). In both cases, the translations from one MT system are used as pseudo-references for translations from the other systems.

Quality labels The automatic metrics selected for quality labelling and prediction are BLEU and TER.[9] **BLEU** (BiLingual Evaluation Understudy) is a precision-oriented metric that compares n-grams (n=1-4 in our case) from reference documents against n-grams of the MT output, measuring how close the output of the system is to one or more references. **TER** (Translation Error Rate) (Snover et al., 2006) measures the minimum number of edits required to transform the MT output in the reference document. The Asiya Toolkit[10](Giménez and Màrquez, 2010) was used to calculate both metrics.

Baselines As baseline, we use 17 competitive features from the QuEst toolkit (Specia et al., 2013) (the so-called **baseline features** or **BL**.[11]) Since the baseline features are sentence-level, we aggregated them by computing the average for each feature across all sentences in a document. As a second baseline (**Mean**), we calculate the average BLEU or TER scores in the QE training set, and apply this value to all entries (documents) in the test set.

Pseudo-reference features BLEU and TER scores are computed between the output of the MT system of interest and alternative MT systems, at document-level, and used as features in QE models. For the FAPESP corpus, translations from Google Translate were selected as pseudo-references, since this system has shown the best average BLEU score in the QE training set. For the WMT13 corpus, translations from *uedin-wmt13-en-es*, for EN-ES, and *uedin-heafield-unconstrained* for ES-EN, were used as pseudo-references, since these systems achieved the best BLEU scores in the WMT13 translation shared task. Regarding the difference between the systems, for the FAPESP corpus, this difference is guaranteed since GOOGLE is considerably different from SYSTRAN, and is trained on a different (much larger) corpus than MOSES. For the WMT13 corpus, it is not possible to make this assumption, as many of the systems participating in the shared task are close variations of Moses.

Feature sets As feature sets, we combine LC and LSA features with BL (**BL+LC**, **BL+LSA** and **BL+LC+LSA**) to create the models with discursive information. The pseudo-reference features are combined with the baseline (**BL+Pseudo**) and with all other features (**BL+LC+LSA+Pseudo**).

Machine learning algorithm We use the Support Vector Machines (SVM) regression algorithm with a radial basis function kernel and hyperparameters optimised via grid search to train the QE models with all feature sets The scikit-learn module available in QuEst was used for that.

Evaluation metrics The QE models with different feature sets are evaluated using **MAE** (Mean Absolute Error): $MAE = \frac{\sum_{i=1}^{n} |H(s_i) - V(s_i)|}{N}$ where $H(s_i)$ is the predicted score, $V(s_i)$ is the true score and N is the number of data points in the test set. To verify the significance of the results, two-tailed pairwise t-test (p<0.05) was performed for different prediction outputs.

Method Two sets of experiments were conduct. First (Section 5.1), we consider the outputs of the FAPESP corpus of MOSES, SYSTRAN and GOOGLE separately, using as training and test sets the outputs of each system individually, with GOOGLE translations used as pseudo-references for the other two systems. The second set of experiments (Section 5.2) considers, for the FAPESP corpus, the combination of the outputs of MOSES and SYSTRAN (MOS+SYS), again with GOOGLE translations used as pseudo-references. For the WMT2013 corpora, we mixed translations from all except the best system, which were used as pseudo-references.

5 Experiments and results

5.1 MT system-specific models

The results for the prediction of BLEU and TER for MOSES, SYSTRAN and GOOGLE systems

[8] `http://www.statmt.org/wmt13/`

[9] METEOR was also used but the results were inconclusive

[10] `http://asiya.lsi.upc.edu/`

[11] `http://www.quest.dcs.shef.ac.uk/quest_files/features_blackbox_baseline_17`

in the FAPESP corpus are shown in Table 1. The best results for MOSES and SYSTRAN were obtained with the inclusion of pseudo-references (BL+Pseudo and BL+LC+LSA+Pseudo), with both BLEU and TER. However, only the improvements for MOSES showed statistically significant difference: with both BLEU and TER, the best results were tied between BL+Pseudo and BL+LC+LSA+Pseudo, but there are still significant differences between their predictions. An interesting finding is that without considering pseudo-reference features for MOSES and SYSTRAN, the best results are achieved with LSA features. In fact, for SYSTRAN the results from using of only BL+LSA are not significantly different from the use of all features (including pseudo-references).

For GOOGLE, the best results (for BLEU and TER) were obtained by BL+LC [12]. However, BLEU predictions showed no significant difference among all feature sets and the best TER figure was not significantly different from BL+LC+LSA.

In order to understand whether the MAE scores obtained are "good enough", it is interesting to compare them against the error of the Mean baseline, but also to analyse the average of the true scores and the range of variation of these true scores in the test set (last two lines in Table 1). For the prediction of BLEU scores, the true scores range from 0 to 0.5 for MOSES and SYSTRAN, and from 0 to 0.8 for GOOGLE. This suggests that the impact of error differences in MOSES and SYSTRAN is higher. A wider range of scores and a relatively higher Mean MAE could indicate a relatively easier prediction task. This is directly connected to the variation in the quality of the translations in the datasets. This seems to be the case with BLEU prediction for GOOGLE translations: the improvements between the Mean baseline and the BL features is much higher than with the other MT systems. The variation in terms of TER is larger, making improvements over the Mean baseline possible with all feature sets.

Given the low MAE scores obtained by the Mean baseline, as well as with simple BL features, one could say that in general the task of predicting BLEU and TER is close to trivial, at least in the FAPESP corpus. This is again due to the low variation in the quality of texts translated by each system. This is to be expected, given the very nature of document-level prediction: major variations in the quality of specific translated segments get smoothed out throughout the document. In addition, the FAPESP corpus consists of texts from the same style and domain. On the other hand, the average quality (as measured by BLEU and TER metrics) of the different MT systems on the same corpus is very different, as shown in the penultimate line of Table 1. This motivates the experiment described next.

5.2 MT system-independent models

To analyse document-level QE in a more challenging scenario, we experiment with mixing different MT system outputs, for both FAPESP and WMT2013 corpora. Results are shown in Table 2.

The ranges of BLEU/TER scores are now wider, and the overall error scores (including for the Mean baseline) are higher in these settings, showing that this is indeed a harder task. Again, the best results are obtained with the use of pseudo-reference features. However, in this case statistically significant differences against other results were only observed with MOS+SYS BLEU prediction and ES-EN TER prediction. For EN-ES BLEU prediction, the best result (0.043 for BL+Pseudo) showed no significant difference against BL+LC+LSA+Pseudo (0.045). For ES-EN BLEU prediction, there is no significant difference among the results of BL+LSA, BL+LC+LSA and BL+Pseudo. For MOS+SYS TER prediction, BL+Pseudo and BL+LC+LSA+Pseudo showed no significant difference. EN-ES TER prediction was the only case were the BL results showed no significant difference against pseudo-reference features. It is worth mentioning that, as in the previous experiments, if we disregard the pseudo-reference features – which may not be available in many real-world scenarios – the LSA feature sets show the best results.

6 Conclusions

In this paper we focused document-level machine translation quality estimation. We presented an attempt to address the problem by considering discourse information in translation quality estimation in terms of novel features, relying on lexical cohesion aspects. LSA cohesion features showed very promising results.

Features based on pseudo-references were also

[12] Pseudo-reference features were not used for GOOGLE, since its outputs was used as pseudo-reference for the other systems.

	BLEU			TER		
	MOSES	SYSTRAN	GOOGLE	MOSES	SYSTRAN	GOOGLE
Mean	0.059	0.047	0.066	0.063	0.062	0.068
BL	0.046	0.047	0.056	0.054	0.059	0.061
BL+LC	0.044	0.043	**0.055**	0.053	0.059	**0.055**
BL+LSA	0.044	0.044	0.058	0.055	0.059	0.060
BL+LC+LSA	0.044	0.043	0.057	0.053	0.058	0.061
BL+Pseudo	**0.042***	0.038	-	**0.052***	**0.051**	-
BL+LC+LSA+Pseudo	**0.042***	**0.036**	-	**0.052***	**0.051**	-
Test-set average	0.365	0.275	0.456	0.427	0.506	0.372
Test-set range	[0.004,0.558]	[0,0.406]	[0.004, 0.79]	[0.245,1.056]	[0,1.071]	[0.12,1.084]

Table 1: MAE scores for document-level prediction of BLEU and TER for the FAPESP corpus. Boldfaced figures indicate the smallest MAE for a given test set; * indicates a statistically significant difference against all other results; underlined values indicate no significant difference against the best system.

	BLEU			TER		
	FAPESP	WMT2013		FAPESP	WMT2013	
	MOS+SYS	EN-ES	ES-EN	MOS+SYS	EN-ES	ES-EN
Mean	0.064	0.061	0.076	0.07	0.066	0.089
BL	0.045	0.056	0.065	0.063	0.059	0.069
BL+LC	0.044	0.058	0.065	0.063	0.066	0.07
BL+LSA	0.044	0.052	0.051	0.062	0.057	0.051
BL+LC+LSA	0.044	0.053	0.052	0.064	0.054	0.062
BL+Pseudo	0.043	**0.043**	**0.038**	**0.053**	**0.034**	**0.038***
BL+LC+LSA+Pseudo	**0.038***	0.045	0.043	0.054	**0.034**	0.04
Test-set average	0.32	0.266	0.261	0.466	0.524	0.55
Test-set range	[0,0.558]	[0.107,0.488]	[0.072,0.635]	[0,1.07]	[0.317,0.72]	[0.216,0.907]

Table 2: MAE scores for document-level prediction of BLEU and TER for the FAPESP corpus (mixing MOSES and SYSTRAN) and for the WMT2013 EN-ES and ES-EN corpora (mixing all but best system).

explored. Confirming the findings in (Soricut and Echihabi, 2010; Shah et al., 2013), these features were found responsible for the most significant improvements over strong baselines. However, in most settings, our proposed LSA cohesion features performed as well as pseudo-reference features.

Predicting automatic metrics at document-level proved a less challenging task than we expected. This was mostly due to the low variance in the quality of translations for the various documents in the corpus by a given MT system. This was confirmed by the low prediction error obtained by a simple baseline that assigns the mean quality score (BLEU or TER) of the training set to all instances of the test set. Outperforming this mean baseline proved particularly difficult for some MT systems when predicting BLEU. Putting MT systems of various quality levels together made the task more complex. As a consequence, our QE models yielded more significant improvements over the baseline.

In future work, we plan to model this problem as predicting post-editing effort scores, as it has been done in the state-of-the-art work for QE at sentence-level. This will require larger datasets with post-edited machine translations and document-level markup.

Acknowledgements: This work was supported by the EXPERT (EU Marie Curie ITN No. 317471) project.

References

Albrecht, Joshua S. and Rebecca Hwa. 2008. The Role of Pseudo References in MT Evaluation. In *Proceedings of WMT 2008*, pages 187–190, Columbus, OH.

Almaghout, Hala and Lucia Specia. 2013. A CCG-based Quality Estimation Metric for Statistical Machine Translation. In *Proceedings of the XIV MT Summit*, pages 223–230, Nice, France.

Avramidis, Eleftherios, Maja Popovic, David Vilar Torres, and Aljoscha Burchardt. 2011. Evaluate with confidence estimation: Machine ranking of translation outputs using grammatical features. In *Proceedings of WMT 2011*, pages 65–70, Edinburgh, UK.

Aziz, Wilker and Lucia Specia. 2011. Fully Automatic Compilation of a Portuguese-English Parallel Corpus for Statistical Machine Translation. In *Proceedings of STIL 2011*, Cuiabá, MT, Brazil.

Bach, Nguyen, Fei Huang, and Yaser Al-Onaizan. 2011. Goodness: A method for measuring machine translation confidence. In *Proceedings of ACL 2011*, pages 211–219, Portland, OR.

Banerjee, Satanjeev and Alon Lavie. 2005. METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments. In *Proceedings of the ACL 2005 Workshop on Intrinsic and Extrinsic Evaluation Measures for MT and/or Summarization*, pages 65–72, Ann Arbor, MI.

Blatz, John, Erin Fitzgerald, George Foster, Simona Gandrabur, Cyril Goutte, Alex Kulesza, Alberto Sanchis, and Nicola Ueffing. 2004. Confidence Estimation for Machine Translation. In *Proceedings of COLING 2004*, pages 315–321, Geneva, Switzerland.

Bojar, Ondřej, Christian Buck, Chris Callison-Burch, Christian Federmann, Barry Haddow, Philipp Koehn, Christof Monz, Matt Post, Radu Soricut, and Lucia Specia. 2013. Findings of the 2013 Workshop on Statistical Machine Translation. In *Proceedings of WMT 2013*, pages 1–44, Sofia, Bulgaria.

Carpuat, Marine and Michel Simard. 2012. The trouble with SMT consistency. In *Proceedings of WMT 2012*, pages 442–449, Montréal, Canada.

Felice, Mariano and Lucia Specia. 2012. Linguistic features for quality estimation. In *Proceedings of WMT 2012*, pages 96–103, Montréal, Canada.

Giménez, Jesús and Lluís Màrquez. 2010. Asiya: An Open Toolkit for Automatic Machine Translation (Meta-)Evaluation. *The Prague Bulletin of Mathematical Linguistics*, 94:77–86.

Graesser, Arthur C., Danielle S. McNamara, Max M. Louwerse, and Zhiqiang Cai. 2004. Coh-Metrix: Analysis of text on cohesion and language. *Behavior Research Methods, Instruments, and Computers*, 36(2):193–202.

Hardmeier, Christian. 2011. Improving machine translation quality prediction with syntatic tree kernels. In *Proceedings of EAMT 2011*, pages 233–240, Leuven, Belgium.

Koehn, Philipp, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Bertoldi Nicola Federico, Marcello, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. MOSES: Open source Toolkit for Statistical Machine Translation. In *Proceedings of ACL 2007, demonstration session*, Prague, Czech Republic.

Landauer, Thomas K., Peter W. Foltz, and Darrell Laham. 1998. An Introduction to Latent Semantic Analysis. *Discourse Processes*, 25(2-3):259–284.

Louis, Annie and Ani Nenkova. 2013. Automatically Assessing Machine Summary Content Without a Gold Standard. *Computational Linguistics*, 39(2):267–300.

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei jing Zhu. 2002. BLEU: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of ACL 2002*, pages 311–318, Philadelphia, PA.

Pighin, D and L Màrquez. 2011. Automatic projection of semantic structures: an application to pairwise translation ranking. In *Proceedings of SSST-5*, pages 1–9, Portland, OR.

Rubino, Raphael, José G. C. de Souza, Jennifer Foster, and Lucia Specia. 2013. Topic Models for Translation Quality Estimation for Gisting Purposes. In *Proceedings of the XIV MT Summit*, pages 295–302, Nice, France.

Shah, Kashif, Trevor Cohn, and Lucia Specia. 2013. An Investigation on the Effectiveness of Features for Translation Quality Estimation. In *Proceedings of the XIV MT Summit*, pages 167–174, Nice, France.

Snover, Matthew, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of Translation Edit Rate with Targeted Human Annotation. In *Proceedings of AMTA 2006*, pages 223–231, Cambridge, MA.

Soricut, Radu and Abdessamad Echihabi. 2010. TrustRank: Inducing Trust in Automatic Translations via Ranking. In *Proceedings of the ACL 2010*, pages 612–621, Uppsala, Sweden.

Soricut, Radu and Sushant Narsale. 2012. Combining Quality Prediction and System Selection for Improved Automatic Translation Output. In *Proceedings of WMT 2012*, pages 163–170, Montréal, Canada.

Soricut, Radu, Nguyen Bach, and Ziyuan Wang. 2012. The SDL Language Weaver Systems in the WMT12 Quality Estimation Shared Task. In *Proceedings of WMT 2012*, pages 145–151, Montréal, Canada.

Specia, Lucia, Marco Turchi, Nicola Cancedda, Marc Dymetman, and Nello Cristianini. 2009. Estimating the Sentence-Level Quality of Machine Translation Systems. In *Proceedings of EAMT 2009*, EAMT-2009, pages 28–37, Barcelona, Spain.

Specia, Lucia, Kashif Shah, Jose G.C. de Souza, and Trevor Cohn. 2013. QuEst - A translation quality estimation framework. In *Proceedings of WMT 2013: System Demonstrations*, ACL-2013, pages 79–84, Sofia, Bulgaria.

Wong, Billy T. M. and Chunyu Kit. 2012. Extending machine translation evaluation metrics with lexical cohesion to document level. In *Proceedings of EMNLP-CONLL 2012*, pages 1060–1068, Jeju Island, Korea.

Xiong, Deyi, Min Zhang, and Haizhou Li. 2010. Error detection of statistical machine translation using linguistic features. In *Proceedings of ACL 2010*, pages 604–611, Uppsala, Sweden.

Quality estimation for translation selection

Kashif Shah and **Lucia Specia**
Department of Computer Science
University of Sheffield
S1 4DP, UK
{kashif.shah,l.specia}@sheffield.ac.uk

Abstract

We describe experiments on quality estimation to select the best translation among multiple options for a given source sentence. We consider a realistic and challenging setting where the translation systems used are unknown, and no relative quality assessments are available for the training of prediction models. Our findings indicate that prediction errors are higher in this blind setting. However, these errors do not have a negative impact in performance when the predictions are used to select the best translation, compared to non-blind settings. This holds even when test conditions (text domains, MT systems) are different from model building conditions. In addition, we experiment with quality prediction for translations produced by both translation systems and human translators. Although the latter are on average of much higher quality, we show that automatically distinguishing the two types of translation is not a trivial problem.

1 Introduction

Quality Estimation (QE) [Blatz et al., 2004, Specia et al., 2009] has several applications in the context of Machine Translation (MT), considering the use of translations for both inbound (e.g. gisting) and outbound (e.g. post-editing) purposes. To date, research on quality estimation has been focusing mostly on predicting absolute single-sentence quality scores. However, for certain applications an absolute score may not be necessary. Our goal is to model quality estimation by contrasting the output of several translation sources for the same input sentence against each other. The outcome of this process is a ranking of alternative translations based on their predicted quality. For our application, we are only interested in the top-ranked translation, which could for example be provided to a human post-editor for revision.

Previous research on this task has focused on ranking translations from multiple MT systems where system identifiers are known beforehand. Based on such identifiers, individual quality prediction models can be trained for each MT system [Specia et al., 2010], and the predicted (absolute) scores for translations of a given source sentence across multiple MT systems used to rank them. Alternatively, quality prediction models can be built to directly output a ranking of alternative translations based on training data annotated with relative quality scores, using for example pairwise ranking algorithms [Avramidis, 2013, Avramidis and Popović, 2013].

In this paper we model translation selection considering a scenario where translations are produced by multiple MT systems, but the identifiers of the MT systems are not given, i.e., we assume a blind setting where the sources of the translations are not known. While ranking approaches to system selection could also be used in this blind setting, they require training data labelled with comparative assessments of translations produced by multiple sources. In our experiments, we assume a more general scenario where the labelling of training data is produced for individual translation segments in absolute terms, independently and regardless of their origin. In addition, we also experiment with predicting the quality for human translations. Although human translations are on average of much higher quality than machine translations, we show that this is not always the case and that automatically distinguishing the two types of translation is not a trivial problem.

We present experiments with four language pairs and various prediction models in blind and non-blind settings, as well as with the use of the resulting predictions for translation selection. We show that while prediction errors are higher in blind settings, this does not have a negative impact in performance when using predictions in the task of translation selection. Our best result in terms of the quality scores of the selected translation sets are obtained with prediction models where all available translations are polled together in a system-agnostic way. Finally, we show that these system-agnostic models have good performance when predicting quality for out-of-domain translations, produced by other MT systems.

2 Related work

A handful of system ranking and selection techniques have been proposed in recent years. For an overview of various related approaches we refer the reader to the WMT13 shared task on QE [Bojar et al., 2013]. This shared task included a system ranking track aimed at predicting 5-way rankings for translations produced by five MT systems and ranked by humans for model bulding. All related work relies on either knowing the system identifiers or having access to relative rankings of translations at training time.

MT system selection was first proposed by Specia et al. [2010]. QE models are trained independently for each MT system, and the translation option with highest prediction score is used. 77% of the sentences with the highest QE score also have the highest score according to humans. In contrast, 54% of accuracy was found when selecting translations from the best MT system on average.

He et al. [2010] focus on the ranking between translations from either an MT system or a translation memory for post-editing. Classifiers showed promising results in selecting the option with the lowest estimated edit distance.

Hildebrand and Vogel [2013] use an classic n-best list re-ranking approach based on predicting BLEU scores. A feature set where all features that are solely based on the source sentence were removed showed the best results.

Biçici [2013] uses language and MT system independent features to predict F1 scores with regression algorithms. A threshold for judging if two translations are equal over the predicted F1 scores was learned from data.

Avramidis [2013] and Avramidis and Popović [2013] decompose rankings into pairwise decisions, with the best translation for each candidate pair predicted using logistic regression. A number of features of the source and target languages, including pseudo-references, are used. A similar pairwise ranking approach was used by Formiga et al. [2013], but with random forest classifiers.

3 Experimental settings

Datasets Our datasets contain news domain texts in four language pairs (Table 1): English-Spanish (**en-es**), Spanish-English (**es-en**), English-German (**en-de**), and German-English (**de-en**). Each contains a different number of source sentences and their human translations, as well as 2-3 versions of machine translations: by a statistical (SMT) system, a rule-based (RBMT) system and, for en-es/de only, a hybrid system. Source sentences were extracted from tests sets of WMT13 and WMT12, and the translations were produced by top MT systems of each type (SMT, RBMT and hybrid - hereafter **system2**, **system3**, **system4**) which participated in the translation shared task, plus the additional professional translation given as reference (**system1**). These are the official datasets used for the WMT14 Task 1.1 on QE.[1]

Languages	# Training Src/Tgt	# Test Src/Tgt
en-es	954/3,816	150/600
en-de	350/1,400	150/600
de-en	350/1,050	150/450
es-en	350/1,050	150/450

Table 1: Number of training and test source (Src) and target (Tgt) sentences.

Each translation in this dataset has been labelled by a professional translator with [1-3] scores for "perceived" post-editing effort, where:

- **1** = perfect translation, no editing needed.
- **2** = near miss translation: maximum of 2-3 errors, and possibly additional errors that can be easily fixed (capitalisation, punctuation).
- **3** = very low quality translation, cannot be easily fixed.

[1] http://www.statmt.org/wmt14/quality-estimation-task.html

The distribution of true scores in both training and test sets is given in Figures 1 and 2, for each language pair, and for each language pair and translation source, respectively.

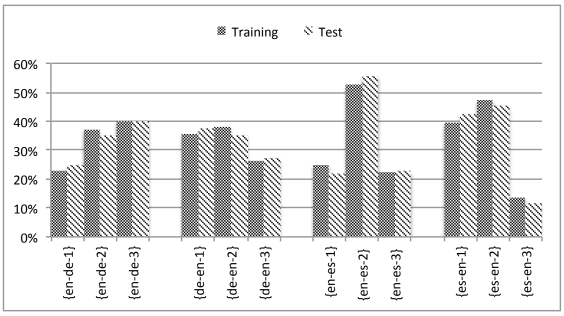


Figure 1: Distribution of true scores by lang. pair.

Out-of-domain test sets For three language pairs, we also experiment with out-of-domain test sets (Table 2) provided by translation companies (also made available by WMT14) and annotated in the same way as above by a translation company (i.e., one professional translator). These were generated using the companies' own source data (different domains than news), and own MT system (different from the three used in our main datasets).

ID	Languages	# Test
LSP$_1$	**en-es**	233
LSP$_2$	**en-es**	738
LSP$_3$	**en-de**	297
LSP$_4$	**es-en**	388
LSP$_5$	**es-en**	677

Table 2: Number of out-of-domain test sentences.

Features We use the `QuEst` toolkit [Specia et al., 2013, Shah et al., 2013] to extract two feature sets for each dataset:

- **BL**: 17 features used as baseline in the WMT shared tasks on QE.
- **AF**: 80 common MT system-independent features (superset of **BL**).

The resources used to extract these features (language models, etc.) are also available as part of the WMT14 shared task on QE.

Learning algorithms We use the Support Vector Machines implementation within `QuEst` to perform either regression (SVR) or classification (SVC) with Radial Basis Function as kernel and parameters optimised using grid search. For SVC, we consider the "one-against-all" approach for multi-class classification with all classes are weighted equally.

Evaluation metrics To evaluate our models, we use standard metrics for regression (MAE: mean absolute error; RMSE: root mean squared error) and classification (precision, recall and F1). For each Table and dataset, bold-faced figures represent results that are significantly better (paired t-test with $p \leq 0.05$) with respect to the baseline.

4 Classification experiments

Our main motivation to use classifiers is the need to distinguish human from machine translations to isolate the former for the system selection task, since in most settings they are not available. We are also interested in measuring the performance of classification-based QE in system selection.

In the experiments to distinguish human translations from machine translations, we pool all MT and human translations together for each language pair, and build binary classifiers where we label all human translations as 1, and all system translations as 0. Results are given in Table 3, where MC stands for "majority class". They show a large variation across language pairs, although MC is outperformed in all cases in terms of F1. The lower performance for **en-es** and **en-de** may be because here translations from three MT systems are put together, while for the remaining datasets, only two MT systems are available. Nevertheless, figures for **en-es** are substantially better than those for **en-de**. This could also be due to the fact that more high quality **human translations** are available for **es-en** and **de-en** (see Figure 2). On the the other hand, for language combination datasets where more low quality human translations or more high quality machine translations are found, distinguishing between these sets becomes a more difficult task. With similar classifiers (albeit different datasets), Gamon et al. [2005] reported as trivial the problem of distinguishing human translations from machine translations. Overall, our results could indicate that this is a harder problem nowadays than some years ago, possibly pointing in the direction that MT systems produce translations that are closer to human translation nowadays.

Results with SVC in the task of classifying instances with 1-3 labels (including human translations) are shown in Table 4. The performance of

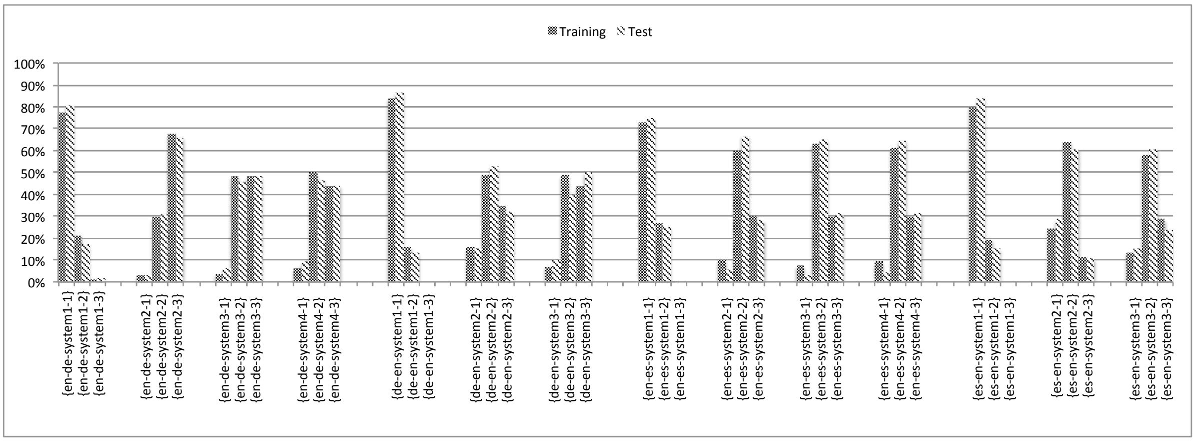


Figure 2: Distribution of true scores for each MT system and language pair.

	System	#feats	Precision	Recall	F1
en-de	MC	-	0.3041	0.1316	0.1566
	BL	17	**0.3272**	0.1200	**0.1756**
	AF	80	**0.3281**	0.1193	**0.1801**
de-en	MC	-	0.5041	0.2416	0.2961
	BL	17	**0.5420**	**0.2321**	**0.3262**
	AF	80	**0.5468**	**0.2333**	**0.3271**
en-es	MC	-	0.6541	0.1521	0.2312
	BL	17	**0.7012**	0.1524	**0.2561**
	AF	80	**0.7188**	0.1533	**0.2527**
es-en	MC	-	0.7311	0.3513	0.4625
	BL	17	**0.7665**	**0.3651**	**0.4942**
	AF	80	**0.7639**	**0.3667**	**0.4954**

Table 3: SVC to distinguish between human translations and machine translations (all MT systems). MC corresponds to always picking machine translation (most frequent) as label.

	System	#feats	Precision	Recall	F1
en-de	MC	-	0.1521	0.4231	0.2072
	BL	17	0.1600	0.4000	0.2285
	AF	80	**0.3401**	**0.4316**	**0.3078**
de-en	MC	-	0.1121	0.3521	0.1672
	BL	17	**0.1248**	0.3533	**0.1844**
	AF	80	**0.1267**	0.3512	**0.1851**
en-es	MC	-	0.2911	0.5561	0.4014
	BL	17	0.3080	0.5550	0.3961
	AF	80	0.3092	0.5542	0.3972
es-en	MC	-	0.1941	0.4516	0.2677
	BL	17	**0.2075**	0.4555	**0.2851**
	AF	80	**0.2071**	0.4541	**0.2855**

Table 4: SVC to predict 1-3 labels for each language pair, with all translations pooled together. MC corresponds to applying the most frequent class of the training set to all test instances.

the classifiers is compared to the standard baseline of the majority class in the training set (MC). The classifiers perform better than MC for all language pairs except **en-es**, particularly in terms of recall and F1. Since this dataset is significantly larger, the majority class is likely to be more representative of the general data distribution. Overall, the classification results are not very positive, and this corroborates the findings of previous work contrasting classification and regression [Specia et al., 2010].

Overall, the use of all features (**AF**) instead of baseline features (**BL**) only leads to slight improvements in some cases.

5 Regression experiments

Here we train models to estimate a continuous score within [1,3], as opposed to discrete 1-3 scores. We compare prediction error for models trained (and tested) on pooled translations from all MT systems (and humans) together (Table 5) – which would be comparable to the settings used to generate Table 4 – against models trained on data from each MT system (or human translation) individually (i.e., system identifier known). For the latter, we consider two settings at test time:

- The system (or human) used to produce the translation is unknown (Table 6 blind setting), and therefore all models are applied, one by one, to predict the quality of this translation and the average prediction is used.
- The system (or human) is known and thus the model for the same translation system/human can be used for prediction (Table 6 non blind setting).

These two variants may be relevant depending on the application scenario. We consider very realistic a scenario where system identifiers are known by developers at model building time, but unknown at test time, e.g. if QE is provided as a web ser-

	System	#feats	MAE	RMSE
en-de	Mean	-	0.6831	0.7911
	BL	17	**0.6416**	**0.7620**
	AF	80	**0.6303**	**0.7616**
de-en	Mean	-	0.6705	0.7979
	BL	17	**0.6524**	**0.7791**
	AF	80	**0.6518**	**0.7682**
en-es	Mean	-	0.4585	0.6678
	BL	17	0.5240	0.6590
	AF	80	0.5092	0.6442
es-en	Mean	-	0.5825	0.6718
	BL	17	0.5736	0.6788
	AF	80	**0.5662**	**0.6663**

Table 5: SVR to build predictions models for each language pair combination, with all translation sources (including human) pooled together.

vice with pre-trained models (Table 6). Users may request predictions using translations produced by any sources, and for out-of-domain data (Table 7). In all tables, **Mean** represents a strong baseline: assigning the mean of the training set labels to all test set instances.

Comparing the two variants of the blind setting (Tables 5 - blind training and test; and Table 6, blind test only), we see that pooling the data from multiple translation systems for blind model training leads to significantly better results than training models for individual translation sources but testing them in blind settings. This is likely to be due to the larger quantities of data available in the pooled models. In fact, the best results are observed with **en-es**, the largest dataset overall.

Comparing scores between blind versus non-blind test setting in Table 6, we observe a substantial difference in the scores for each of the individual translation system. This shows that the task is much more challenging when QE models are trained independently, but the identifiers of the systems producing the test instances are not known.

There is also a considerable difference in the performance of individual models for different translation systems, which can be explained by the different distribution of scores (and also indicated by the performance of the **Mean** baseline). However, in general the prediction performance of the individual models seems less stable, and worse than the baseline in several cases. Interestingly, the individual models trained on human translations only (system1) do even worse than individual models for MT systems. This can be an indication that the features used for quality prediction are not sufficient to model human translations.

In all cases, the use of all features (**AF**) instead of baseline features (**BL**) comparable or better results.

Table 7 shows the results for SVR models trained on pooled models for each language pair (i.e., models in Table 5) when applied to predict the quality of the out-of-domain datasets. This is an extremely challenging task, as the only constant between training and test data is the language pair. The text domain is different, and so are MT systems used to produce the translations. In addition, no human translation is available in the out-of-domain test sets. Surprisingly, the prediction errors are low, even lower than those observed for the in-domain test sets. This is true for all except two out-of-domain test sets: LSP$_5$, which contains unusual texts (such as URLs and markup tags), and LSP$_2$. Manual inspection of these source and translation segments show many extremely short segments (1-2 words), which may render most features useless.

WMT14	System	#features	MAE	RMSE
LSP$_1$ (en-es)	Mean	-	0.2715	0.4311
	BL	17	**0.2524**	**0.4116**
	AF	80	**0.2419**	**0.4076**
LSP$_2$ (en-es)	Mean	-	0.8119	0.9703
	BL	17	0.8094	**0.9470**
	AF	80	**0.8062**	**0.9453**
LSP$_3$ (en-de)	Mean	-	0.4315	0.5914
	BL	17	0.4270	**0.5500**
	AF	80	0.4262	**0.5463**
LSP$_4$ (es-en)	Mean	-	0.5012	0.6711
	BL	17	**0.4847**	**0.6412**
	AF	80	**0.4812**	**0.6392**
LSP$_5$ (es-en)	Mean	-	0.7112	0.8881
	BL	17	**0.6862**	**0.8447**
	AF	80	**0.6828**	**0.8472**

Table 7: Results with SVR pooled models tested on out-of-domain datasets.

6 System selection results

In what follows we turn to using the predictions from SVR and SVC models showed before for system selection. The task consists in selecting, for each source segment, the best *machine* translation among all available: two or three depending on the language pair. For this experiments, we eliminated the human translations – as they do not tend to be represented in settings for system selection. Given the low performance of our classifiers in Table 3, we ruled out human translations based on the meta-data available, without using these classifiers. Another reason to rule out human translations from the selection is that they are used as references to

	System	#feats	blind		non-blind	
			MAE	RMSE	MAE	RMSE
en-de-system1	Mean	-	1.0351	1.2133	0.3552	0.4562
	BL	17	1.0487	1.2348	**0.3350**	0.4540
	AF	80	1.0510	1.2375	**0.3325**	0.4545
en-de-system2	Mean	-	0.7780	0.9339	0.4857	0.5487
	BL	17	**0.7006**	0.9499	**0.3615**	**0.4634**
	AF	80	**0.6924**	**0.9124**	**0.3570**	**0.4644**
en-de-system3	Mean	-	0.7369	0.8426	0.5577	0.6034
	BL	17	**0.6354**	**0.7950**	**0.4535**	**0.5363**
	AF	80	**0.6572**	0.8127	**0.4482**	**0.5245**
en-de-system4	Mean	-	0.7231	0.8215	0.5782	0.6433
	BL	17	**0.6438**	**0.7842**	**0.4912**	**0.5834**
	AF	80	**0.6386**	**0.7905**	**0.4818**	**0.5741**
de-en-system1	Mean	-	0.8594	1.0882	0.2506	0.3409
	BL	17	0.8747	1.1299	**0.2123**	0.3421
	AF	80	0.8747	1.1299	**0.2065**	0.3415
de-en-system2	Mean	-	0.7321	0.8484	0.5412	0.6678
	BL	17	**0.6897**	0.8330	**0.4745**	**0.5931**
	AF	80	0.7122	0.8509	**0.4604**	**0.5850**
de-en-system3	Mean	-	0.8137	0.9253	0.6000	0.6640
	BL	17	**0.7472**	**0.8903**	**0.4965**	**0.6011**
	AF	80	**0.7629**	0.9300	**0.4828**	**0.5901**
en-es-system1	Mean	-	0.8542	0.9923	0.3883	0.4353
	BL	17	0.8956	1.0480	**0.3633**	0.4390
	AF	80	0.8957	1.0480	**0.3519**	0.4381
en-es-system2	Mean	-	0.5567	0.6952	0.4232	0.5314
	BL	17	**0.5275**	0.6827	**0.3812**	**0.4951**
	AF	80	**0.5302**	0.6884	**0.3730**	**0.4893**
en-es-system3	Mean	-	0.5653	0.6998	0.4288	0.5213
	BL	17	**0.5155**	**0.6711**	**0.3821**	**0.4844**
	AF	80	**0.5184**	**0.6704**	**0.3714**	**0.4761**
en-es-system4	Mean	-	0.5573	0.6955	0.4300	0.5321
	BL	17	**0.5103**	**0.6680**	**0.4022**	**0.5162**
	AF	80	**0.5206**	**0.6727**	**0.3902**	**0.5016**
es-en-system1	Mean	-	0.6617	0.8307	0.3026	0.3916
	BL	17	0.6617	0.8307	0.3022	0.3917
	AF	80	0.6617	0.8308	0.3023	0.3915
es-en-system2	Mean	-	0.5637	0.6931	0.4494	0.6027
	BL	17	0.5588	0.7023	**0.4384**	0.6061
	AF	80	0.5567	0.7026	**0.4309**	0.6053
es-en-system3	Mean	-	0.6602	0.8129	0.4720	0.6245
	BL	17	0.7233	0.8621	0.4993	0.6220
	AF	80	0.6973	0.8435	0.4974	0.6198

Table 6: SVR to build individual predictions models for each language pair and translation source.

compute BLEU scores of the selected sets of sentences, as explained below.

To provide an indication of the average quality of each MT system, Table 8 presents the BLEU scores on the test and training sets for individual MT systems. The bold-face figures for each language test set indicate the (BLEU) quality that would be achieved for that test set if the "best" system were selected on the basis of the average (BLEU) quality of the training set (i.e., no system selection). There is a significant variance in terms of quality scores, as measured by BLEU, among the outputs of 2-3 MT systems for each language pair, with training set quality being a good predictor of test set quality for all but **en-es**, once again, the largest dataset.

We measure the performance of the selected sets in two ways: (i) by computing the BLEU scores of the entire sets containing the supposedly best translations, using the human translation available in the datasets as reference, and (ii) by computing the accuracy of the selection process against the human labels, i.e., by computing the proportion of times both system selection and human agree (based on the pre-defined 1-3 human labels) that the sentence selected is the best among the 2-3 options (2-3 MT systems). We compare the results obtained from building pooled (all MT systems)

WMT14	system2		system3		system4	
	Test	Training	Test	Training	Test	Training
en-de	15.39	12.79	13.75	13.83	**17.04**	16.19
de-en	**27.96**	24.03	22.66	20.19	-	-
en-es	**25.89**	34.13	32.68	28.42	29.25	31.97
es-en	**37.83**	40.01	23.55	25.07	-	-

Table 8: BLEU scores of individual MT systems, without system selection. Bold-faced figures indicate scores obtained when selecting best system on average (using BLEU scores for the training set).

against individual prediction models (one per MT system).

Table 9 and 10 show the selection results with various models trained on MT translations only:

- **Best-SVR(I):** Best translation selected with regression model trained on data from individual MT systems, where prediction models are trained per MT system, and the translation selected for each source segment is the one with the highest predicted score among these independent models. This requires knowing the source of the translations for training, but not for testing (blind test).

- **Best-SVR(P):** Best translation selected with regression model trained on pooled data from all MT systems. This assumes a *blind* setting where the source of the translations for both training and test sets is unknown, and thus pooling data is the only option for system selection.

- **Best-SVC(P):** Best translation selected with the classification model trained on pooled data from all MT systems as above. For SVC, only the pooled models were used as predicting exact 1-3 labels with independently trained models leads to an excessively number of ties (i.e., multiple translations with same score), making the decision between them virtually arbitrary.

Table 9 shows that the regression models trained on individual systems – **Best-SVR(I)** – with **AF** as feature set yield the best results, despite the fact that error scores (MAE and RMSE) for these individual systems are worse than for systems trained on pooled data. This is somewhat expected as knowing the system that produced the translation (i.e., training models for each MT system) adds a strong bias to the prediction problem towards the average quality of such a system, which is generally a decent quality predictor. We note however that the **Best-SVR(P)** models are not far behind in terms of performance, with the **Best-SVC(P)** following closely. In all cases, the gains with respect to the MC baseline are substantial. More important, we note the gains in BLEU scores as compared to the bold-face test set figures in Table 8, showing that our system selection approach leads to best translated test sets than simply picking the MT system with best average quality (BLEU).

Results in terms of accuracy in selecting the best translation (Table 10) are similar to those in terms of BLEU scores, with models trained independently performing best.

7 Remarks

We have presented a number of experiments showing the potential of a system selection techniques in scenarios where translations are given by multiple MT systems and system identifiers are unknown. System selection was performed based on predictions from classification and regression models. Results in terms of BLEU and accuracy of selected sets with an MT system-agnostic approach show improvements for system selection over strong baselines.

Overall – in bind test settings – although the prediction error of models trained on individual MT systems are worse than models trained on pooled data, when used for system selection, models trained on individual systems generally perform better.

Acknowledgments

This work was supported by funding from the from European Union's Seventh Framework Programme for research, technological development and demonstration under grant agreement no. 296347 (QTLaunchPad).

References

E. Avramidis. Sentence-level ranking with quality estimation. *Machine Translation*, 28:1–20,

	System	#feats	Best-SVR(I)	Best-SVR(P)	Best-SVC(P)
en-de	MC	-	16.14	15.55	16.12
	BL	17	17.20	17.05	17.33
	AF	80	**18.10**	17.55	17.32
de-en	MC	-	25.81	25.17	25.07
	BL	17	28.39	28.13	28.19
	AF	80	**28.75**	28.43	28.21
en-es	MC	-	30.88	30.29	30.23
	BL	17	32.92	32.81	32.74
	AF	80	**33.45**	33.25	33.10
es-en	MC	-	30.13	29.70	29.53
	BL	17	38.10	38.11	38.14
	AF	80	**38.73**	38.41	38.15

Table 9: BLEU scores on best selected translations (I = Individual, P = Pooled).

	System	#feats	Best-SVR(I)	Best-SVR(P)	Best-SVC(P)
en-de	MC	-	0.1823	0.1787	0.1793
	BL	17	0.2017	0.2001	0.2033
	AF	80	**0.2155**	0.2055	0.2065
de-en	MC	-	0.3511	0.3527	0.3559
	BL	17	0.3733	0.3711	0.3713
	AF	80	0.3915	**0.3923**	0.3821
en-es	MC	-	0.3698	0.3643	0.3659
	BL	17	0.3912	0.3901	0.3892
	AF	80	**0.4102**	0.4087	0.4051
es-en	MC	-	0.4073	0.4043	0.4059
	BL	17	0.4321	0.4301	0.4290
	AF	80	**0.4513**	0.4421	0.4423

Table 10: Accuracy in selecting the best translation for each dataset (I = Individual, P = Pooled).

2013.

E. Avramidis and M. Popović. Machine learning methods for comparative and time-oriented Quality Estimation of Machine Translation output. In *8th WMT*, pages 329–336, Sofia, 2013.

E. Biçici. Referential translation machines for quality estimation. In *8th WMT*, Sofia, 2013.

J. Blatz, E. Fitzgerald, G. Foster, S. Gandrabur, C. Goutte, A. Kulesza, A. Sanchis, and N. Ueffing. Confidence Estimation for Machine Translation. In *Coling*, pages 315–321, Geneva, 2004.

O. Bojar, C. Buck, C. Callison-Burch, C. Federmann, B. Haddow, P. Koehn, C. Monz, M. Post, R. Soricut, and L. Specia. Findings of the 2013 WMT. In *8th WMT*, pages 1–44, Sofia, 2013.

L. Formiga, M. González, A. Barrón-Cedeno, J. A. Fonollosa, and L. Màrquez. The TALP-UPC approach to system selection: Asiya features and pairwise classification using random forests. In *8th WMT*, pages 359–364, Sofia, 2013.

M. Gamon, A. Aue, and M. Smets. Sentence-level MT evaluation without reference translations: beyond language modeling. In *EAMT-2005*, Budapest, 2005.

Y. He, Y. Ma, J. van Genabith, and A. Way. Bridging smt and tm with translation recommendation. In *ACL-2010*, pages 622–630, Uppsala, Sweden, 2010.

S. Hildebrand and S. Vogel. MT quality estimation: The CMU system for WMT'13. In *8th WMT*, pages 373–379, Sofia, 2013.

K. Shah, E. Avramidis, E. Biçici, and L. Specia. Quest - design, implementation and extensions of a framework for machine translation quality estimation. *Prague Bull. Math. Linguistics*, 100: 19–30, 2013.

L. Specia, M. Turchi, N. Cancedda, M. Dymetman, and N. Cristianini. Estimating the Sentence-Level Quality of Machine Translation Systems. In *EAMT-2009*, pages 28–37, Barcelona, 2009.

L. Specia, D. Raj, and M. Turchi. Machine translation evaluation versus quality estimation. *Machine Translation*, pages 39–50, 2010.

L. Specia, K. Shah, J. G. C. d. Souza, and T. Cohn. Quest - a translation quality estimation framework. In *ACL-2013 Demo Session*, pages 79–84, Sofia, 2013.

An Efficient Two-Pass Decoder for SMT Using Word Confidence Estimation

Ngoc-Quang Luong **Laurent Besacier** **Benjamin Lecouteux**
LIG, Campus de Grenoble
41, Rue des Mathématiques,
UJF - BP53, F-38041 Grenoble Cedex 9, France
{ngoc-quang.luong,laurent.besacier,benjamin.lecouteux}@imag.fr

Abstract

During decoding, the Statistical Machine Translation (SMT) decoder travels over all complete paths on the Search Graph (SG), seeks those with cheapest costs and backtracks to read off the best translations. Although these winners beat the rest in model scores, there is no certain guarantee that they have the highest quality with respect to the human references. This paper exploits Word Confidence Estimation (WCE) scores in the second pass of decoding to enhance the Machine Translation (MT) quality. By using the confidence score of each word in the *N*-best list to update the cost of SG hypotheses containing it, we hope to "reinforce" or "weaken" them relied on word quality. After the update, new best translations are re-determined using updated costs. In the experiments on our *real WCE scores* and *ideal (oracle) ones*, the latter significantly boosts one-pass decoder by 7.87 BLEU points, meanwhile the former yields an improvement of 1.49 points for the same metric.

1 Introduction

Beside plenty of commendable achievements, the conventional one-pass SMT decoders are still not sufficient yet in yielding human-acceptable translations (Zhang et al., 2006; Venugopal et al., 2007). Therefore, a number of methods to enhance them are proposed, such as: post-editing, re-ranking or re-decoding, etc. Post-editing (Parton et al., 2012) is in fact known to be a human-inspired task where the machine post edits translations in a second automatic pass. In re-ranking (Zhang et al., 2006; Duh and Kirchhoff, 2008; Bach et al., 2011), more features are integrated with the existing multiple model scores for re-selecting the best candidate among *N*-best list. Meanwhile, the re-decoding process intervenes directly into the decoder's search graph (SG), driving it to the optimal path (cheapest hypothesis).

The two-pass decoder has been built by several discrepant ways in the past. Kirchhoff and Yang (2005); Zhang et al. (2006) train additional Language Models (LM) and combine LM scores with existing model scores to re-rank the *N*-best list. Also focusing on the idea of re-ranking, yet Bach et al. (2011); Luong et al. (2014) employ sentence and word confidence scores in the second pass. Meanwhile, Venugopal et al. (2007) do a first pass translation without LM, but use it to score the pruned search hyper-graph in the second pass.

This work concentrates on a second automatic pass where the costs of all hypotheses in the decoder's SG containing words of the *N*-best list will be updated regarding the word quality predicted by Word Confidence Estimation (Ueffing and Ney, 2005) (WCE) system. In single-pass decoding, the decoder searches among complete paths (i.e. those cover all source words) for obtaining the optimal-cost ones. Essentially, the hypothesis cost is a composite score, synthesized from various SMT models (reordering, translation, LMs etc.). Although the *N*-bests beat other SG hypotheses in term of model scores, there is no certain clue that they will be the closest to the human references. As the reference closeness is the users' most pivotal concern on SMT decoder, this work establishes one second pass where model-independent scores related to word confidence prediction are integrated into the first-pass SG to re-determine the best hypothesis. Inheriting the first pass's N-best list, the second one involves three additional steps:

- Firstly, apply a WCE classifier on the N-best list to assign the quality labels ("Good" or "Bad") along with the confidence probabilities for each word.

- Secondly, for each word in the N-best list, update the cost of all SG's hypotheses containing it by adding the update score (see Section 3.2 for detailed definitions).

- Thirdly, search again on the updated SG for the cheapest-cost hypothesis and trace backward to form the new best translation.

Basically, this initiative originates from an intuition that all parts of hypotheses corresponding to correct (predicted) words should be appreciated while those containing wrong ones must be weakened. The use of novel decoder-independent and objective features like WCE scores is expected to raise up the better candidate, rather than accepting the current sub-optimal one. The new decoder can therefore use both *real* and *oracle* word confidence estimates. In the next section, we introduce the SG's structure. Section 3 depicts our approach about using WCE scores to modify the first-step SG. The experimental settings and results, followed by in-depth analysis and comparison to other approaches are discussed in Section 4 and Section 5. The last section concludes the paper and opens some outlooks.

2 Search Graph Structure

The SMT decoder's Search Graph (SG) can be roughly considered as a "vast warehouse" storing all possible hypotheses generated by the SMT system during decoding for a given source sentence. In this large directed acyclic graph, each hypothesis is represented by a path, carrying all nodes between its begin and end ones, along with the edges connecting adjacent nodes. One hypothesis is called *complete* when all the source words are covered and *incomplete* otherwise. Starting from the empty initial node, the SG is gradually enlarged by expanding hypotheses during decoding. To avoid the explosion of search space, some weak hypotheses can be pruned or recombined. In order to facilitate the access and the cost calculation, each hypothesis **H** is further characterized by the following fields (we can access the value of the field f of hypothesis **H** by using the notion $f(H)$):

- **hyp**: hypothesis ID
- **stack**: the stack (ID) where the hypothesis is placed, also the number of foreign (source) words translated so far.
- **back**: the backpointer pointing to its previous cheapest path.
- **transition** : the cost to expand from the previous hypothesis (denoted by **pre(H)**) to this one.
- **score**: the cost of this hypothesis. Apparently, $score(H) = score(pre(H)) + transition.$
- **out**: the last output (target) phrase. It is worth to accentuate that **out** can contain multiple words.
- **covered**: the source coverage of **out**, represented by the start and the end position of the source words translated into **out**.
- **forward**: the forward pointer pointing to the cheapest outgoing path expanded from this one.
- **f-score**: estimated future cost from this partial hypothesis to the complete one (end of the SG).
- **recombined**: the pointer pointing to its recombined[1] hypothesis.

Figure 1 illustrates a simple SG generated for the source sentence: ***"identifier et mesurer les facteurs de mobilization"***. The attributes **"t"** and **"c"** refer to the transition cost and the source coverage, respectively. Hypothesis **175541** is extended from **57552**, when the three words from 3rd to 5th of the source sentence (*"les facteurs de"*) are translated into *"the factors of"* with the transition cost of -8.5746. Hence, its cost is: $score(175541) = score(57552) + transition(175541) = -16.1014 + (-8.5746) = -24.6760$. Three rightmost hypotheses: **204119**, **204109** and **198721** are complete since they cover all source words. Among them, the cheapest-cost

[1] In the SG, sometimes we recombine hypotheses to reduce the search space in a risk-free way. Two hypotheses can be recombined if they agree in (1) the source word covered so far (2) the last two target words generated, and (3) the end of the last source phrase covered.

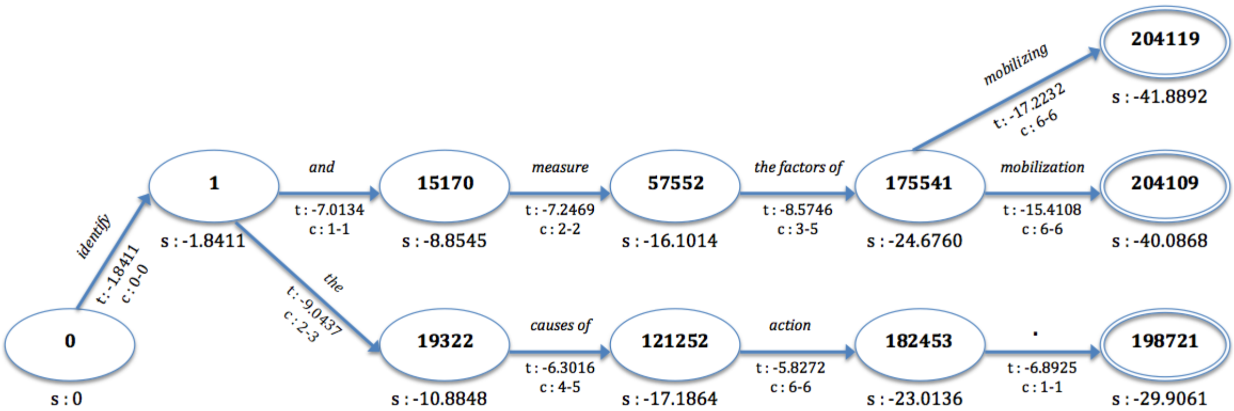


Figure 1: An example of search graph representation

one[2] is **198721**, from which the model-best translation is read off by following the track back to the initial node **0**: ***"identify the causes of action ."***.

3 Our Approach: Integrating WCE Scores into SG

In this section, we present the idea of using additional scores computed from WCE output (labels and confidence probabilities) to update the SG. We also depict the way that update scores are defined. Finally, the detailed algorithm followed by an example illustrates the approach.

3.1 Principal Idea

We assume that the decoder generates N best hypotheses $T = \{T_1, T_2, ..., T_N\}$ at the end of the first pass. Using the WCE system (which can only be applied to sequences of words - and not directly to the search graph - that is why N best hypotheses are used), we are able to assign the *j-th* word in the hypothesis T_i, denoted by t_{ij}, with one appropriate quality label, c_{ij} (e.g. *"G"* (Good: no translation error), *"B"* (Bad: need to be edited)), followed by the confidence probabilities ($P_{ij}(G), P_{ij}(B)$ or $P(G), P(B)$ for short). Then, the second pass is carried out by considering every word t_{ij} and its labels and scores $c_{ij}, P(G), P(B)$. Our principal idea is that, if t_{ij} is a *positive* (good) translation, i.e. c_{ij} = "G'' or $P(G) \approx 1$, all hypotheses $H_k \in SG$ containing it in the SG should be "rewarded" by reducing their cost. On the contrary, those containing *negative* (bad) translation will be "penalized". Let $reward(t_{ij})$ and $penalty(t_{ij})$ denote the reward or penalty score of t_{ij}. The new **transition** cost of H_k after being updated is formally defined by:

$$transition'(H_k) = transition(H_k) +$$
$$\begin{cases} reward(t_{ij}) & \text{if } t_{ij} = good\ translation \\ penalty(t_{ij}) & \text{if } otherwise \end{cases} \quad (1)$$

The update finishes when all words in the *N*-best list have been considered. We then re-compute the new score of complete hypotheses by tracing backward via back-pointers and aggregating the **transition cost** of all their edges. Essentially, the re-decoding pass reorders SG hypotheses in term of the more *"G"* words (predicted by WCE system) they contain, the more cost reduction will be made and consequentially, the more opportunity they get to be admitted in the *N*-best list. The re-decoding performance depends largely on the accurateness of confidence scores, or in other words, the WCE quality.

It is vital to note that, during the update process, we might face a phenomena that the word t_{ij} (corresponds to the same source words) occurs in different sentences of the *N*-best list. In this case, for the sake of simplicity, we process it only at its first occurrence (in the highest rank sentence) instead of updating the hypotheses containing it multiple times. In other words, if we meet the exact t_{ij} once again in the next N-best sentence(s), no further score update will be done in the SG.

3.2 Update Score Definitions

Defining the update scores is obviously a nontrivial task as there is no correlation between WCE labels and the SG costs. Furthermore, we have no clue about how proportional the SMT model and

[2] It is important to note that the concept **cheapest cost hypothesis** means that it has the highest model's score value. In other words, the higher the model score value, the "cheaper" the hypothesis is.

WCE (penalty or reward) scores should share in order to ensure that both of them will be appreciated. In this article, we propose several types of update scores, deriving from the global or local cost.

3.2.1 Definition 1: Global Update Score

In this type, an unique score derived from the cost of the current best hypothesis H^* (by the first pass) is used for all updates. We propose to compute this score by two ways: (a) exploiting WCE labels $\{c_{ij}\}$; *or* (b) only WCE confidence probabilities $\{P(G), P(B)\}$ will matter, WCE labels are left aside.

Definition 1a:

$$penalty(t_{ij}) = -reward(t_{ij}) = \alpha * \frac{score(H^*)}{\#words(H^*)} \tag{2}$$

Where $\#words(H^*)$ is the number of target words in H^*, the positive coefficient α accounts for the impact level of this score on the hypothesis's final cost and can be optimized during experiments. Here, $penalty(t_{ij})$ gets negative sign (since $score(H^*) < 0$) and will be added to the transition cost of all hypotheses containing t_{ij} in case where this word is labelled as *"B"*; whereas $reward(t_{ij})$ (same value, opposite sign) is used in the other case.

Definition 1b:

$$\begin{aligned} update(t_{ij}) &= \alpha * P(B) * \frac{score(H^*)}{\#words(H^*)} \\ &\quad - \beta * P(G) * \frac{score(H^*)}{\#words(H^*)} \\ &= (\alpha * P(B) - \beta * P(G)) * \frac{score(H^*)}{\#words(H^*)} \end{aligned} \tag{3}$$

Where $P(G)$, $P(B)$ ($P(G) + P(B) = 1$) are the probabilities of "Good" and "Bad" class of t_{ij}. The positive coefficient α and β can be tuned in the optimization phase. In this definition, the fact that $update(t_{ij})$ is **a reward** ($reward(t_{ij})$) or **a penalty** ($penalty(t_{ij})$) will depend on t_{ij}'s goodness. Indeed, we have: $update(t_{ij}) = reward(t_{ij})$ if $update(t_{ij}) > 0$, which means: $\alpha*[1-P(G)]-\beta*P(G) < 0$ (since $score(H^*) < 0$), therefore $P(G) > \frac{\alpha}{\alpha+\beta}$. On the contrary, if $P(G)$ is under this threshold, $update(t_{ij})$ takes a negative value and therefore becomes a penalty.

3.2.2 Definition 2: Local Update Score

The update score of each (local) hypothesis H_k depends on its current transition cost, even when they cover the same word t_{ij}. Similarly to **Definition 1**, two sub-types are defined as follows:

Definition 2a:

$$penalty(t_{ij}) = -reward(t_{ij}) = \alpha * transition(H_k) \tag{4}$$

Definition 2b:

$$\begin{aligned} update(t_{ij}) &= \alpha * P(B) * transition(H_k) \\ &\quad - \beta * P(G) * transition(H_k) \\ &= (\alpha * P(B) - \beta * P(G)) * transition(H_k) \end{aligned} \tag{5}$$

Where $transition(H_k)$ denotes the current transition cost of hypothesis H_k, and the meanings of coefficient α (**Definition 2a**) or α, β (**Definition 2b**) are analogous to those of **Definition 1a** (**Definition 1b**), respectively.

3.3 Re-decoding Algorithm

The below pseudo-code depicts our re-decoding algorithm using WCE labels (**Definition 1a** and **Definition 2a**).

Algorithm 1 Using WCE labels in **SG** decoding

```
Input: $SG = \{H_k\}, T = \{T_1, T_2, ..., T_N\}, C = \{c_{ij}\}$
Output: $T' = \{T'_1, T'_2, ..., T'_N\}$
{Step 1: Update the Search Graph}
$Processed \leftarrow \emptyset$
for $T_i$ in $T$ do
    for $t_{ij}$ in $T_i$ do
        $p_{ij} \leftarrow$ position of the source words aligned to $t_{ij}$
        if $(t_{ij}, p_{ij}) \in Processed$ then
            continue; {ignore if $t_{ij}$ appeared in the previous sentences}
        end if
        $Hypos \leftarrow \{H_k \in SG | \ out(H_k) \ni t_{ij}\}$
        if $(c_{ij} = ``Good'')$ then
            for $H_k$ in $Hypos$ do
                $transition(H_k) \leftarrow transition(H_k) + reward(t_{ij})$ {reward hypothesis}
            end for
        else
            for $H_k$ in $Hypos$ do
                $transition(H_k) \leftarrow transition(H_k) + penalty(t_{ij})$ {penalize hypothesis}
            end for
        end if
        $Processed \leftarrow Processed \cup \{(t_{ij}, p_{ij})\}$
    end for
end for
{Step 2: Trace back to re-compute the score for all complete hypotheses}
for $H_k$ in $Final$ (Set of complete hypotheses) do
    $score(H_k) \leftarrow 0$
    while $H_k \neq$ initial hypothesis do
        $score(H_k) \leftarrow score(H_k) + transition(H_k)$
        $H_k \leftarrow pre(H_k)$
    end while
end for
{Step 3: Select N cheapest hypotheses and output the new list $T'$}
```

Rank	Cost	Hypotheses + WCE labels						
1	-29.9061	identify	the	cause	of	action	.	
		G	G	G	G	B	B	
2	-40.0868	identify	and	measure	the	factors	of	mobilization
		G	G	G	G	G	G	G

Table 1: The *N*-best (N=2) list generated by the SG in Figure 1 and WCE labels

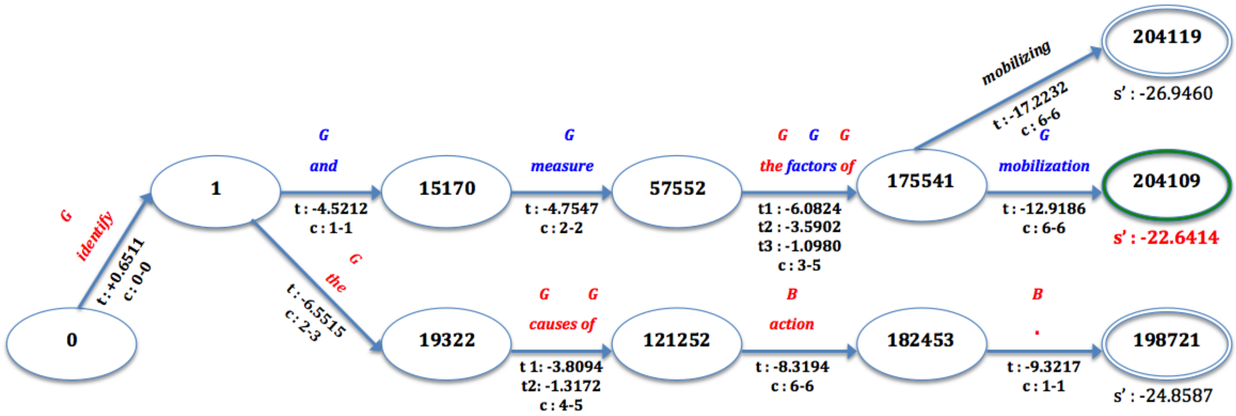


Figure 2: Details of update process for the SG in Figure 1. The first loop (when 1st rank hypothesis is used) is represented in red color, while the second one is in blue. For edges with multiple updates, all transition costs after each update are logged. The winning cost is also emphasized by red color.

The algorithm in case of using WCE confidence probabilities (**Definition 1b** and **Definition 2b**) is essentially similar, except the update step (from line 10 to line 18) is replaced by the following part:

```
for $H_k$ in $Hypos$ do
    $transition(H_k) \leftarrow transition(H_k) + update(t_{ij})$
end for
```

During the update process, the pairs including the visited word t_{ij} and the position of its aligned source words p_{ij} is consequentially admitted to $Processed$, so that all the analogous pairs $(t_{ij}^{'}, p_{ij}^{'})$ occuring in the latter sentences can be discarded. For each t_{ij}, a list of hypotheses in the SG containing it, called $Hypo$, is formed, and its confidence score c_{ij} (or $P(G)$) determines whether all members H_k in $Hypo$ will be rewarded or penalized. Once having all words in the N-best list visited, we obtain a new SG with updated transition costs for all edges containing them. The last step is to travel over all complete hypotheses (stored in $Final$) to re-compute their scores and then backtrack the cheapest-cost hypothesis to output the new best translation.

These above depictions can be clarified by taking another look at the example in Figure 1: from this SG, the N-best list (for the sake of simplicity, we choose $N = 2$) is generated as the single-pass decoder's result. According to our approach, the second pass starts by tagging all words in the list with their confidence labels, as seen in Table 1. Then, the graph update process is performed for each word in the list, sentence by sentence, which details are tracked in Figure 2. In this example, we apply **Definition 1a** to calculate the reward or penalty score, with $\alpha = \frac{1}{2}$, resulting in: $penalty(t_{ij}) = -reward(t_{ij}) = \frac{1}{2} * \frac{-29.9061}{6} = -2.4922$. Firstly, all hypotheses containing words in the 1st ranked sentence are considered. Since the word *"identify"* is labeled as *"G"*, its corresponding edge (connecting two nodes **0** and **1**) is rewarded and updated with a new cost : $t_{new} = t_{old} + reward = -1.8411 + 2.4922 = +0.6511$. On the contrary, the edge between two nodes **121252** and **182453** is penalized and takes new cost: $t_{new} = t_{old} + penalty = -5.8272 + (-2.4922) = -8.3194$, due to the bad quality of the word *"action"*. Obviously, the edges having multiple considered words (e.g. the one between nodes **19322** and **121252**) will be updated multiple times, and the transition costs after each update can be also observed in Figure 2 (e.g. $t1$, $t2$, etc). Next, when the 2nd-best is taken into consideration, all repeated words (e.g. *"iden-*

tify", *"the"* and *"of"*) are waived since they have been visited in the first loop, whereas the remaining ones are identically processed. The only untouched edge in this SG corresponds to the word *"mobilizing"*, as this word does not belong to the list. Once having the update process finished, the remaining job is to recalculate the final cost for every complete path and returns the new best translation: ***"identify and measure the factors of mobilization"*** (new cost = -22.6414).

4 Experimental Setup

4.1 Datasets and SMT Resources

From a dataset of 10,881 French sentences, we applied a Moses-based SMT system to generate their English hypotheses. Next, human translators were invited to correct MT outputs, giving us the post editions. The set of triples (source, hypothesis, post edition) was then divided into the training set (10000 first triples) and test set (881 remaining ones). The WCE model was trained over all **1-best hypotheses** of the training set. More details on our WCE system can be found in next section.

The N-best list ($N = 1000$) with involved alignment information is also obtained on the test set (1000 * 881 = 881000 sentences) by using Moses (Koehn et al., 2007) options *"-n-best-list"* and *"-print-alignment-info-in-n-best"*. Besides, the SGs are extracted by some parameter settings: *"-output-search-graph"*, *"-search-algorithm 1"* (using cube pruning) and *"-cube-pruning-pop-limit 5000"* (adds 5000 hypotheses to each stack). They are compactly encoded under a plain formatted text file that is convenient to transform into user-defined structures for further processing. We then store the SG for each source sentence in a separated file, and the average size is 43.8 MB.

4.2 WCE scores and Oracle Labels

We employ the Conditional Random Fields (Lafferty et al., 2001) (CRFs) as our machine learning method, with WAPITI toolkit (Lavergne et al., 2010), to train the WCE model. A number of knowledge resources are employed for extracting the system-based, lexical, syntactic and semantic characteristics of word, resulting in the total of 25 major feature types as follows:

- Target Side: target word; bigram (trigram) backward sequences; number of occurrences
- Source Side: source word(s) aligned to the target word
- Alignment Context (Bach et al., 2011): the combinations of the target (source) word and all aligned source (target) words in the window ±2
- Word posterior probability (Ueffing et al., 2003)
- Pseudo-reference (Google Translate): Does the word appear in the pseudo reference?
- Graph topology (Luong et al., 2013): number of alternative paths in the confusion set, maximum and minimum values of posterior probability distribution
- Language model (LM) based: length of the longest sequence of the current word and its previous ones in the target (resp. source) LM. For example, with the target word w_i: if the sequence $w_{i-2}w_{i-1}w_i$ appears in the target LM but the sequence $w_{i-3}w_{i-2}w_{i-1}w_i$ does not, the n-gram value for w_i will be 3.
- Lexical Features: word's Part-Of-Speech (POS); sequence of POS of all its aligned source words; POS bigram (trigram) backward sequences; punctuation; proper name; numerical
- Syntactic Features: null link (Xiong et al., 2010); constituent label; depth in the constituent tree
- Semantic Features: number of word senses in WordNet.

In the next step, the word's reference labels (or so-called **oracle labels**) are initially set by using TERp-A toolkit (Snover et al., 2008) in one of the following classes: "I' (insertions), "S" (substitutions), "T" (stem matches), "Y" (synonym matches), "P" (phrasal substitutions), "E" (exact matches) and are then regrouped into binary class: *"G"* (good word) or *"B"* (bad word). Once having the prediction model, we apply it on the test set (881 x 1000 best = 881000 sentences) and get needed WCE labels along with confidence probabilities. In term of F-score, our WCE system reaches very promising performance in predicting *"G"* label (**87.65%**), and acceptable for *"B"* label (**42.29%**). Both **WCE** and **oracle** labels will be used in experiments.

4.3 Experimental Decoders

We would like to investigate the WCE's contributions in two scenarios: real WCE and ideal WCE (where all predicted labels are totally identical to the oracle ones). Therefore, we experiment with the seven following decoders:

- **BL**: Baseline (1-pass decoder)
- **BL+WCE(1a, 1b, 2a, 2b)**: four 2-pass decoders, using our estimated WCE labels and confidence probabilities to update the SGs, and the update scores are calculated by **Definition (1a, 1b, 2a, 2b)**.
- **BL+OR(1a, 2a)**: two 2-pass decoders, computing the reward or penalty scores by **Definition (1a, 2a)** on the oracle labels

It is important to note that, when using oracle labels, **Definition 1b** becomes **Definition 1a** and **Definition 2b** becomes **Definition 2a**, since if a word t_{ij} is labelled as "G", then $P(G) = 1$ and $P(B) = 0$, and vice versa. In order to tune the coefficients α and β, we carry out a **2-fold cross validation** on the test set. First, the set is split into two equivalent parts: **S1** and **S2**. Playing the role of a development set, **S1** will train the parameter(s) which then be used to compute the update scores on **S2** re-decoding process, and vice versa. The optimization steps are handled by CONDOR toolkit (Berghen, 2004), in which we vary α and β within the interval $[0.00; 5.00]$ (starting point is 1.00), and the maximum number of iterations is fixed as 50. Test set is further divided to launch experiments in parallel on our cluster using an open-source batch scheduler: OAR (Nicolas and Joseph, 2013). This mitigates the overall processing times on such huge SGs. Finally, the re-decoding results for them are properly merged for evaluation.

5 Results

Table 2 shows the translation performances of all experimental decoders and their percentages of sentences which outperform, remain equivalent or degrade the baseline hypotheses (when match against the references, measured by TER). Results suggest that using **oracle labels** to re-direct the graph searching boosts dramatically the baseline quality. **BL+OR(1a)** augments 7.87 points in BLEU, and diminishes 0.0607 (0.0794) point in TER(TERp-A), compared to **BL**. Meanwhile, with **BL+OR(2a)**, these gains are 7.67, 0.0565 and 0.0514 (in that order). Besides, the contribution of our real WCE system scores seems less prominent, yet positive: the best performing **BL+WCE(1a)** increases 1.49 BLEU points of **BL** (0.0029 and 0.0136 gained for TER and TERp-A). More remarkable, tiny ***p*-values** (in the range $[0.00; 0.02]$, seen on Table 2) estimated between BLEU of each **BL+WCE** system and that of **BL** relying on Approximate Method (Clark et al., 2011) indicate that these performance improvements are significant. Results also reveal that the use of WCE labels are slightly more beneficial than that of confidence probabilities: **BL+WCE(1a)** and **BL+WCE(2a)** outperform **BL+WCE(1b)** and **BL+WCE(2b)**. In both scenarios, we observe that the global update score (**Definition 1**) performs more fruitfully compared to the local one (**Definition 2**).

For more insightful understanding about WCE scores' acuteness, we make a comparison with the best achievable hypotheses in the SG (oracles), based on the "LM Oracle" approximation approach presented in (Sokolov et al., 2012). This method allows to simplify the oracle decoding to the problem of searching for the cheapest path on a SG where all transition costs are replaced by the *n*-gram LM scores of the corresponding words. The LM is built for each source sentence using uniquely its target post-edition. We update the SG by assigning all edges with the LM back-off score of the word it contains (instead of using the current transition cost). Finally, we combine the oracles of all sentences yielding BLEU oracle of **66.48**.

To better understand the benefit of SG re-decoding, we compare the obtained performances with those from our previous attempt in using WCE for *N*-best list re-ranking (green zone of Table 2). The idea is to build sentence-level features starting from WCE labels, then integrate them with existing SMT model scores to recalculate the objective function value, thus re-order the *N*-best list (Luong et al., 2014). Both approaches are implemented in analogous settings, e.g. identical SMT system, WCE system, and test set. Results suggest that the contribution of WCE in SG re-decoding outperforms that in N-best re-ranking in both "oracle" or real scenarios. **BL+OR(1a)** overpasses its corresponding oracle re-ranker **BL+OR(Nbest_RR)** in 2.08 points of BLEU, diminishes 0.0253 (0.0280) in TER(TERp-A). Meanwhile, **BL+WCE(1a)** wins real WCE re-ranker **BL+WCE(Nbest_RR)** in 1.03 (BLEU), 0.0015 (TER), 0.0103 (TERp-A). These achievements might originate from the following reasons: (1) In re-ranking, WCE scores are integrated at

Systems	**Performance**			**Comparison to BL**			***p*-**
	BLEU ↑	**TER ↓**	**TERp-A ↓**	**Better (%)**	**Equivalent (%)**	**Worse (%)**	**value**
BL	52.31	0.2905	0.3058	-	-	-	-
BL+WCE(1a)	**53.80**	**0.2876**	**0.2922**	28.72	57.43	13.85	0.00
BL+WCE(1b)	53.24	0.2896	0.2995	26.45	59.26	14.29	0.00
BL+WCE(2a)	53.32	0.2893	0.3018	23.68	60.11	16.21	0.02
BL+WCE(2b)	53.07	0.2900	0.3006	22.27	55.17	22.56	0.01
BL+OR(1a)	**60.18**	**0.2298**	**0.2264**	62.52	24.36	13.12	-
BL+OR(2a)	59.98	0.2340	0.2355	60.18	28.82	11.00	-
BL+OR(Nbest_RR)	58.10	0.2551	0.2544	58.68	29.63	11.69	-
BL+WCE(Nbest_RR)	52.77	0.2891	0.3025	18.04	68.22	13.74	0.01
Oracle BLEU score	**BLEU = 66.48 (from SG)**						

Table 2: Translation quality of the conventional decoder and the 2-pass ones using scores from real or "oracle" WCE, followed by the percentage of better, equivalent or worse sentences compared to **BL**

sentence level, so word translation errors are not fully penalized; and (2) in re-ranking, best translation selection is limited to *N*-best list, whereas we afford the search over the entire updated SG (on which not only N-best list paths but also those contain at least one word in this list are altered) .

6 Conclusion and perspectives

We have presented a novel re-decoding approach for enhancing the SMT quality. Inherited the result from the first pass (*N*-best list), we predict words' labels and confidence probabilities, then employ them to seek a more valuable (cheaper) path over SGs throughout the re-decoding stage. While "oracle" WCE labels extraordinarily lifts the MT quality up (to reach the oracle score), real WCE achieves also the positive and promising gains. The method sharpens WCE increasing contributions in every aspect of SMT. As future work, we focus on estimating in more detail the word quality using MQM[3] metric as error typology, making WCE labels more impactful. Besides, the update scores used in this article would be further considered towards the consistency with SMT graph scores to obtain a better updated SG.

References

Nguyen Bach, Fei Huang, and Yaser Al-Onaizan. Goodness: A method for measuring machine translation confidence. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, pages 211–219, Portland, Oregon, June 19-24 2011.

Frank Vanden Berghen. *CONDOR: a constrained, non-linear, derivative-free parallel optimizer for continuous, high computing load, noisy objective functions*. PhD thesis, University of Brussels (ULB - Université Libre de Bruxelles), Belgium, 2004.

Jonathan Clark, Chris Dyer, Alon Lavie, and Noah Smith. Better hypothesis testing for statistical machine translation: Controlling for optimizer instability. In *Proceedings of the Association for Computational Lingustics*, 2011.

Kevin Duh and Katrin Kirchhoff. Beyond log-linear models: Boosted minimum error rate training for n-best re-ranking. In *Proc. of ACL, Short Papers*, 2008.

Katrin Kirchhoff and Mei Yang. Improved language modeling for statistical machine translation. In *Proceedings of the ACL Workshop on Building and Using Parallel Texts*, pages 125–128, Ann Arbor, Michigan, June 2005.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*, pages 177–180, Prague, Czech Republic, June 2007.

John Lafferty, Andrew McCallum, and Fernando Pereira. Conditional random fields: Probabilistic models for segmenting et labeling sequence data. In *Proceedings of ICML-01*, pages 282–289, 2001.

Thomas Lavergne, Olivier Cappé, and François Yvon. Practical very large scale crfs. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 504–513, 2010.

Ngoc Quang Luong, Laurent Besacier, and Benjamin Lecouteux. Word confidence estimation and its integration in sentence quality estimation for machine translation. In *Proceedings of The Fifth International Conference on Knowledge and Systems Engineering (KSE 2013)*, Hanoi, Vietnam, October 17-19 2013.

Ngoc Quang Luong, Laurent Besacier, and Benjamin Lecouteux. Word confidence estimation for smt n-best list re-ranking. In *Proceedings of the Workshop on Humans and Computer-assisted Translation (HaCaT)*, Gothenburg, Sweden, April 2014.

Capit Nicolas and Emeras Joseph. *OAR Documentation - User Guide*. LIG laboratory, Laboratoire d'Informatique de Grenoble Bat. ENSIMAG - antenne de Montbonnot ZIRST 51, avenue Jean Kuntzmann 38330 MONTBONNOT SAINT MARTIN, 2013.

Kristen Parton, Nizar Habash, Kathleen McKeown, Gonzalo Iglesias, and Adrià de Gispert. Can automatic post-editing make mt more meaningful? In *Proceedings of the 16th EAMT*, pages 111–118, Trento, Italy, 28-30 May 2012.

Matthew Snover, Nitin Madnani, Bonnie Dorr, and Richard Schwartz. Terp system description. In *MetricsMATR workshop at AMTA*, 2008.

Artem Sokolov, Guillaume Wisniewski, and Franc ois Yvon. Computing lattice bleu oracle scores for machine translation. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 120–129, Avignon, France, April 2012.

Nicola Ueffing and Hermann Ney. Word-level confidence estimation for machine translation using phrased-based translation models. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 763–770, Vancouver, 2005.

Nicola Ueffing, Klaus Macherey, and Hermann Ney. Confidence measures for statistical machine translation. In *Proceedings of the MT Summit IX*, pages 394–401, New Orleans, LA, September 2003.

Ashish Venugopal, Andreas Zollmann, and Stephan Vogel. An efficient two-pass approach to synchronous-cfg driven statistical mt. In *Proceedings of Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics*, April 2007.

Deyi Xiong, Min Zhang, and Haizhou Li. Error detection for statistical machine translation using linguistic features. In *Proceedings of the 48th Association for Computational Linguistics*, pages 604–611, Uppsala, Sweden, July 2010.

Ying Zhang, Almut Silja Hildebrand, and Stephan Vogel. Distributed language modeling for n-best list re-ranking. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing (EMNLP 2006)*, pages 216–223, Sydney, July 2006.

[3] http://www.qt21.eu/launchpad/content/training

Poster Session B
User and Product/Project Papers

HBB4ALL: Media Accessibility for HBBTV

European Commission
FP7 CIP-ICT-PSP.2013.5.1
PSP.2013.5.1
621014
http://www.hbb4all.eu

List of partners	
Universitat Autonoma de Barcelona, Spain (Coordinator)	vsonix Gmbh, Germany
Rundfunk Berlin-Brandenburg, Germany	Fundación Centro de Tecnologias de Interac-ción Visual y Comunicaciones, Spain
Institut für Rundfunktechnik GmbH	Screen Subtitling Systems Ltd, UK
Rádio e Televisão de Portugal SA, Portugal	Holken Consultants & Partners, France
Televisió de Catalunya SA, Spain	People's Playground BV, Netherlands
Schweizerische Teletext AG, Switzerland	Universidad Politécnica de Madrid, Spain

Project duration: December 2013 — November 2016

Summary

HBB4ALL builds on HbbTV, as the major European standard, for converged services and looks at both the production and service sides. HbbTV 1.x devices are widely available in the market while HbbTV version 2.0 is currently under development .HbbTV provides a straight-forward specification on how to combine broadcast and broadband content plus interactive applications. TV content can be enhanced with additional synchronised services in a personalised manner. For access services this opens an entirely new opportunity for users who may choose an access service delivered via their IP connection which then seamlessly integrates with the regular broadcast programme.

The project will test access services in various pilot implementations and gather user feedback to assess the acceptance and the achievable quality of service in the various delivery scenarios (broadcasting, hybrid, full IP): (A) Multi-platform subtitle services, (B) alternative audio production and distribution, (C) automatic user interface adaptation, and (D) sign-language translation service.

Pilot A will make available advanced HbbTV automatic multilingual subtitling functionalities, building up on technology currently under development in the European SME-DCL SAVAS AND CIP-PSP SUMAT projects. More specifically, it will test and HbbTV-based news service allowing users to access live content automatically subtitled and translated to multiple languages. Complementary user experience testing of different end user related aspects of subtitling in the hybrid world involving users from the target groups will deliver metrics for Quality of Service.

ALST

Linguistic and sensorial accessibility: technologies for voice-over and audio description

Ministerio de Economía y Competetitividad (Spain)
Proyectos de investigación fundamental no orientada
(Spanish national projects funded by the Ministry of Economy and Competitiviness)
FFI2012-31024
Project duration: January 2013-December 2016

Summary

ALST aims to implement three existing technologies (speech recognition, machine translation and speech synthesis) into two different audiovisual transfer modes (voice-over and audio description) in order to research alternative working flows that may guarantee higher accessibility levels. More specifically, it aims to:

a) research if speech recognition could be used to generate transcripts in a faster and more efficient way. This will be the last step of the project and no results are available at this stage.

b) investigate if machine translation could be efficiently applied to reach high quality audiovisual translations, by analyzing the post-editing effort and comparing it to a standard human creation/translation process. A preliminary analysis has been carried out and first results will be presented.

c) implement TTS technologies instead of human voices and test the end user's reception. A pretest and an experiment with 68 blind and visually impaired participants have been carried out and results will be summarized.

Although limited in scope due to funding restrictions and its national scope, it is an innovation in audiovisual translation because until now research on machine translation in this field has mainly dealt with subtitling. ALST will hopefully be the first step in the application of such technologies in both voice-over and audio description and open new research horizons at international level.

qtleap

quality translation with deep language engineering approaches

European Commission
FP7, Call 10
Strep
#610516
http:/qtleap.eu

	List of partners
	Faculty of Sciences, University of Lisbon, Portugal (coordinator)
	German Research Centre for Artificial Intelligence, Germany
	Charles University in Prague, Czech republic
	Bulgarian Academy of Sciences, Bulgaria
	Humboldt University of Berlin, Germany
	University of Basque Country, Spain
	University of Groningen, The Netherlands
	Higher Functions, Lda, Portugal

Project duration: November 2013 — October 2016

Summary

The incremental advancement of research on Machine Translation (MT) has been obtained by encompassing increasingly sophisticated statistical approaches and fine-grained linguistic features that add to the surface level alignment on which these approaches are ultimately anchored.

The goal of this project is to contribute for the advancement of quality MT by pursuing an approach that further relies on semantics and opens the way to higher quality translation.

We build on the complementarity of the two pillars of language technology — symbolic and probabilistic — and seek to advance their hybridization. We explore combinations of them that amplify their strengths and mitigate their drawbacks, along the development of three MT pilot systems that progressively seek to integrate deep language engineering approaches.

The construction of deep treebanks has progressed to be delivering now the first significant Parallel DeepBanks, where pairs of synonymous sentences from different languages are annotated with their fully-fledged grammatical representations, up to the level of their semantic representation.

The construction of Linked Open Data and other semantic resources, in turn, has progressed now to support impactful application of lexical semantic processing that handles and resolves referential and conceptual ambiguity.

These cutting edge advances permit for the cross-lingual alignment supporting translation to be established at the level of deeper semantic representation. The deeper the level the less language-specific differences remain among source and target sentences and new chances of success become available for the statistically based transduction.

The ASMAT project - Arabic Social Media Analysis Tools

Fatiha Sadat
University of Quebec in Montreal
201 President Kennedy
Montreal, QC, H2X 3Y7, Canada
sadat.fatiha@uqam.ca

List of partners
Atefeh Farzindar, NLP Technologies Inc. http://www.nlptechnologies.ca 52, Le Royer Street W., Montréal, Québec, Canada, H2Y 1W7 farzindar@nlptechnologies.ca

Summary

The main objective of the ASMAT project – *Arabic Social Media Analysis Tools*, is to make available a comprehensive set of language resources and tools covering Arabic dialects in social media context.

Current Arabic NLP tools are capable of analysing large part of standard Arabic, but fail short of handling the dialects and the social media domain. To this end, the project aims to create tools for Arabic language and its varieties following certain tasks: (1) language and dialect identification; (2) dialect to standard (MSA) mapping and vice versa; (3) automatic machine translation from any Arabic dialect to English and French. More specifically, the ASMAT project deals with the *Maghrebi* (*North African*) Arabic dialects for machine translation with very scarce resources.

Parts of the ASMAT project, such as dialect identification for all varieties of Arabic language and a systematic rule-based mapping of the Tunisian dialect to MSA were achieved on December 2013, with an industrial collaboration with NLP Technologies, under NSERC Engage[1] grant. Our latest evaluations showed that Naive Bayes classifiers based on character bi-gram model and trained on data extracted from forums and blogs on 18 Arabic dialects could identify the 18 different Arabic dialects with a considerable overall accuracy of 98% on social media texts. A successful identification of which sentence in written in which dialect could guide the system in using the specific pre-processing tools for the respective dialectal portions.

We have already achieved a rule-based system that converts any text of social media in Tunisian dialect to MSA. We are working on the construction of more linguistic resources for the Tunisian dialect and MSA that will help build a hybrid statistical and rule-based MT system integrated in the ASMAT project. Finally, the translation from the Tunisian dialect to French and/or English will be completed through MSA as a pivot language.

Future works of the ASMAT project are concerned by all varieties of Arabic dialects for machine translation, starting from the Maghrebi.

The ASMAT project will be funded from late 2014 by additional research grants for a longer period.

[1] http://www.nserc-crsng.gc.ca/Professors-Professeurs/RPP-PP/Engage-engagement_eng.asp

XLike: Cross-lingual Knowledge Extraction

European Commission
FP7 Language Technologies (ICT-2011.4.2)
Small and medium scale focused research project (STREP)
288342
http://www.xlike.org

List of partners
Jožef Stefan Institute (JSI), Slovenia (coordinator)
Karlsruhe Institute of Technology (KIT), Germany
Technical University of Catalonia (UPC), Spain
University of Zagreb (UZG), Croatia
Tsinghua University (THU), China
Intelligent Software Components (ISOCO), Spain
Bloomberg (BLO), USA
Slovenian Press Agency (STA), Slovenia
New York Times (NYT), USA (associated partner)
Indian Institute of Technology (IIT), India (associated partner)

Project duration: Januray 2012 — December 2014

Summary

The goal of the XLike project is to develop technology to monitor and aggregate knowledge that is currently spread across mainstream and social media, and to enable cross-lingual services for publishers, media monitoring and business intelligence. The aim is to combine scientific insights from several scientific areas to contribute in the area of cross-lingual text understanding. By combining modern computational linguistics, machine translation, machine learning, text mining and semantic technologies we plan to deal with the following two key open research problems: (1) to extract and integrate formal knowledge from multilingual texts with cross-lingual knowledge bases, and; (2) to adapt linguistic techniques and crowdsourcing to deal with irregularities in informal language used primarily in social media. The developed technology will be language-agnostic, while within the project we specifically address English, German, Spanish, Chinese as major world languages and Catalan, Slovenian and Croatian as minority languages. Knowledge resources from Linked Open Data cloud (e.g. Wikipedia, DBpedia, Wordnets etc.) will be used with special focus on general common sense knowledge base CycKB, that will be used as Interlingua. A number of different methods to translate from natural language to the selected formal language that serves as our Interlingua are being explored, among others also SMT. For languages where no required linguistic resources are available, we use SMT systems trained from parallel or comparable corpora (e.g. drawn from the Wikipedia) to come up with the Interlingua representation.

АБУМАТРАН
ABUMATRAN

Abu-MaTran: Automatic building of Machine Translation

FP7-PEOPLE-2012-IAPP
http://www.abumatran.eu

List of partners	
DCU	Dublin City University, Ireland (coordinator)
prompsit	Prompsit Language Engineering SL, Spain
Universitat d'Alacant Universidad de Alicante	Universitat d'Alacant, Spain
	University of Zagreb, Faculty of Humanities and Social Sciences, Croatia
IEA ILSP	Athena Research and Innovation Center in Information Communication & Knowledge Technologies, Greece

Project duration: January 2013 — December 2016

Summary

Abu-MaTran seeks to enhance industry–academia cooperation as a key aspect to tackle one of Europe's biggest challenges: multilingualism. We aim to increase the hitherto low industrial adoption of machine translation by identifying crucial cutting-edge research techniques (automatic acquisition of corpora and linguistic resources, pivot-language techniques, linguistically augmented statistical translation and diagnostic evaluation), making them suitable for commercial exploitation. We also aim to transfer back to academia the know-how of industry to make research results more robust. We work on a case study of strategic interest for Europe: machine translation for the language of a new member state (Croatian) and related languages. All the resources produced will be released as free/open-source software, resulting in effective knowledge transfer beyond the consortium. The project has a strong emphasis on dissemination, through the organisation of workshops that focus on inter-sectoral knowledge transfer. Finally, we have a comprehensive outreach plan, including the establishment of a Linguistic Olympiad in Spain, open-day activities and the participation in the Google Summer of Code.

At EAMT 2014 we will present the results of the first milestone of the project (July 2013), a general-domain MT system for English–Croatian based on free/open-source software and publicly available data, released on July 1st 2013 to mark Croatia's accession to the EU. We will also present ongoing work towards the second milestone (December 2014) including (i) a domain-specific MT system for English–Croatian in the domain of tourism, (ii) generation of synthetic English–Croatian data via Slovene using quality estimation, (iii) outcomes of the first edition of the Linguistic Olympiad of Spain (September 2013 - March 2014), etc.

QTLaunchPad: Preparation and Launch of a Large-Scale Action for Quality Translation Technology

European Commission
Seventh Framework
Coordination and Support Action
296347
http://www.qt21.eu

List of partners
German Research Center for Artificial Intelligence, Germany (coordinator)
Dublin City University, Ireland
Athena Research and Innovation Center in Information Communication & Knowledge Technologies
The University of Sheffield

Project duration: July 2012 — June 2014

Summary

QTLaunchPad is dedicated to overcoming quality barriers in machine and human translation and in language technologies. It is preparing for a large-scale translation quality initiative for Europe. One of the key contributions of QTLaunchPad is the Multidimensional Quality Metrics (MQM), a customizable system that provides analytic methods to assess machine translation output. This system has been used to assess results from top-performing WMT systems and customer data provided by language service providers. Analysis in the project has focused on “almost good” translations, those segments where MT systems produce results that can be easily fixed, to understand the barriers that impact the best MT systems. It has also worked on the development of quality estimation and linguistic evaluation techniques to assist MT processes to identify those segments that are good enough to use as is, those that can be easily repaired by human post-editors, and those that should be discarded and translated from scratch.

Key findings include the identification of linguistic structures in English that are particularly likely to trigger problems for MT systems of different types (e.g., use of *-ing* verb forms, non-genitive uses of *of*, and differences across languages in permissible positions within sentences). These findings were only possible by combining the insights of human evaluators and the output of computational tools. The insights gained from this analysis will be of use to developers seeking to improve MT systems and to implementers seeking to integrate MT into “real world” production chains that include MT, human translators or posteditors, and other technologies.

Able-to-Include: Improving Accessibility for people with Intellectual Disabilities

EU-CIP 621055

List of consortium partners
Ariadna Servicios Informáticos (Spain)
Fundación Prodis (Spain)
Universitat Pompeu Fabra (Spain)
Building Bridges Training Community Interest Company (UK)
Leuven University (Belgium)
Thomas More Kempen (Belgium)
TeamNet International (Romania)
Microelectronics Applications Centre – MAC (Ireland)
Inclusion Europe (Belgium)

While this project has the wider goal to improve the accessibility of the information society for people with intellectual disabilities, one of the important means for achieving this is the automated **text-to-pictogram** translator which has been developed for Dutch (webservices.ccl.kuleuven.be/picto/). In the Able-to-include project we localize the text-to-pictogram translator in order to make it work for Spanish and English, besides Dutch.

Pictograms have been linked to Wordnet synsets from the Dutch lexical-semantic database Cornetto (Vandeghinste & Schuurman @ LREC2014). We will establish links between Princeton Wordnet and the pictograms by using the equivalence relations which link Cornetto synsets to Princeton synsets. In a second stage we will establish the link between the pictograms and the Spanish Wordnet by using the equivalence relations that are provided between the Spanish Wordnet and Princeton Wordnet. In order to obtain a full localisation of the **Text2Picto** translator, we will also have to adapt the linguistic components to English and Spanish.

We will also provide **Picto2Text** which translates a sequence of pictograms into natural language, and which will be used in combination with a pictogram-selection mechanism that serves as input method for people with writing difficulties.

The tools work with two different pictogram sets, Beta and Sclera.

These tools will be used in several pilot projects involving actual user organisations and users with intellectual disabilities, allowing to measure their impact on the daily lives of the target group.

Smart Computer Aided Translation Environment

IWT-SBO 130041

http://www.ccl.kuleuven.be/scate

List of consortium partners
University of Leuven (CCL - Centre for Computational Linguistics + ESAT/PSI - Centre for Processing of Speech and Images + LIIR - Language Intelligence & Information Retrieval + Research Unit Linguistics @ Thomas More Antwerp
University of Ghent (LT3 Language and Translation Technology Team)
Hasselt University (Expertise Centre for Digital Media)

In the SCATE project we aim at improving the translators' efficiency. Commercial translation tools are faced with ever higher productivity requirements imposed by the globalisation of business activities and the increasing information flow.

The SCATE project intends to improve translators' efficiency along the following axes:

- **Exploitation of already translated data –** We will exploit data more exhaustively through the use of syntactic models for fuzzy matching, and detect syntactically similar constructions in the translation memory. We will investigate complex types of translation grammar induction and tree alignment that allow to *transduce* source syntax trees into target trees (i.e. accepting one tree and producing another). We will investigate how to seamlessly integrate MT into a translation memory, by automatically resolving the syntactic fuzziness of the match through MT techniques.
- **Translation evaluation –** We will automatically judge whether MT output is worth post-editing, or whether the suggested translation can be applied to resolve the fuzzy match in the translation memory. We will build an annotated data set and a taxonomy of typical translation errors and combine this with loggings and analysis of human-machine interaction during post-editing, which targets improvements in automatic confidence estimation of machine translation output.
- **Terminology extraction –** We will automatically extract terminology from comparable corpora in order to speed up the translation process and make translations more consistent. Therefore we will study translator's methods in acquiring domain terminology. We will also research methods to determine which texts in different languages contain comparable information, and we will improve current methods of terminology extraction from comparable corpora through techniques such as cross-lingual topic modelling.
- **Speech recognition –** We will integrate the language model of the MT engine with the language model of the speech recogniser. We will study the adaptation of the recogniser as an input method for the post-editor, and investigate the improvement of speech transcription for translation purposes. Furthermore we will study how to perform automatic domain-adaptation for speech recognition, in order to automatically adapt the language models of the recogniser to the domain.
- **Workflows and personalised user interfaces –** We aim at a higher comfort and productivity for the translators, by analysing and modelling current translation systems and translator's workflows and practices, investigating new visualisations of translation features, and developing and testing new interfaces for translation work.

Kanjingo – A Mobile App for Post-Editing

Sharon O'Brien
CNGL/SALIS
Dublin City University
Ireland
sharon.obrien@dcu.ie

Joss Moorkens
CNGL/SALIS
Dublin City University
Ireland
joss.moorkens@dcu.ie

Joris Vreeke
CNGL
Dublin City University
Ireland
joris.vreeke@dcu.ie

Abstract

This paper describes the Kanjingo post-editing application for smartphones. The application was developed using an agile methodology at the Centre for Global Intelligent Content (CNGL) at DCU and a first stage of user testing was conducted using content from Translators Without Borders.[1] Initial feedback on this app was quite positive. Users identified some particular challenges, e.g. input and sensitivity limitations, insufficient Help, lack of automatic punctuation and capitalization. Development and further testing are ongoing and may include interactive MT, speech as input and focus on Asian languages as target languages in the future.

1 Introduction

Kanjingo is a mobile app for translating a source text and post-editing machine translated target text on a mobile interface. It was developed in the CNGL (Centre for Global Intelligent Content) at Dublin City University.[2] This paper describes the first round of user testing where the objective was to obtain feedback and improve the application.

2 User Testing

2.1 Motivation

The objective of the first stage of user testing focused on Kanjingo's suitability for post-editing machine translated output in a mobile scenario. The motivation for doing so is based on the increasing evidence that volunteers are willing to translate or post-edit for causes they wish to support (Munro, 2010; Petras, 2011)

Our use case scenario for this first round of testing is volunteers for an organization such as "Translators Without Borders" (TWB). The volunteers wish to contribute to the translation effort of this organization, but possibly only have time to translate or post-edit a few segments of text per day on their way to and from work. Our assumption is that volunteers may not wish to sit at a desk to do this work and might like to post-edit a few segments of text while waiting at a bus stop, for example.

The Kanjingo App is not intended to replace a desktop CAT environment. However, since MT suggestions sometimes need to be deleted outright due to poor quality and retranslated by a human, we decided to also test the App's potential to support the human translation task in addition to the post-editing task.

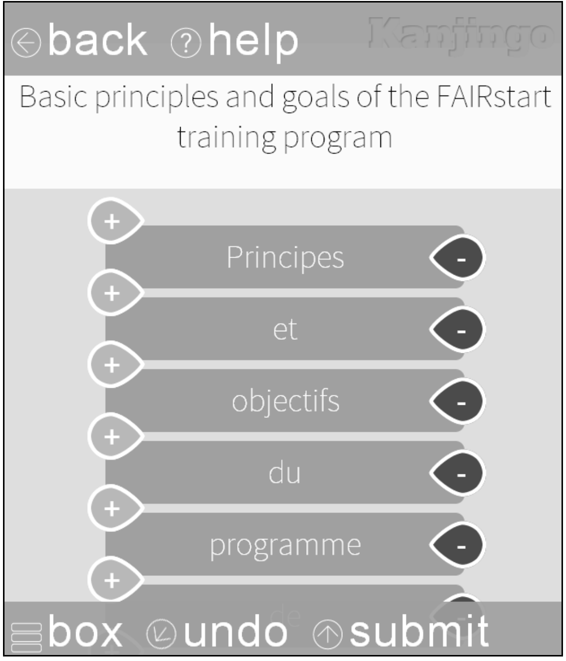

Basic principles and goals of the FAIRstart training program

Figure 1. The Kanjingo post-editing screen

[1] We are very grateful to Translators Without Borders for their collaboration in this project.

[2] This research is supported by the Science Foundation Ireland (Grant 12/CE/I2267) as part of CNGL (www.cngl.ie) at Dublin City University.

2.2 UI Description

When the App is first accessed, the user selects a language pair, e.g. English-French. Source segments are listed in the initial screen presented to users. The user selects a source segment at which point a machine translated segment is presented on the screen in a vertical tiled format (see Figure 1).

If at first users do not know how to interact with the UI, they can click on a Help link which presents them with a screen shot explaining the basic features of the UI (see Figure 2).

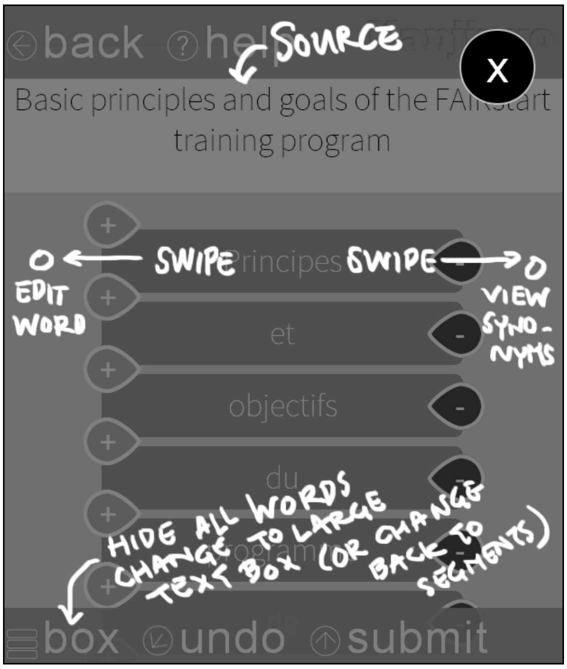

Basic principles and goals of the FAI tart training program

Figure 2. Basic help

As can be seen in Figure 1, each word tile has a "+" symbol on the left which, when tapped, inserts a new empty tile above that word, enabling the post-editor to insert a new word. The '-' sign on the right side of the word tile deletes that tile completely. Tiles can be reordered by dragging the tile up or down on the screen. Users can scroll down through the MT segment by dragging elsewhere on the screen, as with the regular scrolling feature on a smartphone. To edit a word, the user taps on the word, which appears in an edit box. A second tap in the box enables the appearance of the smartphone keyboard.

Once the user has post-edited the segment to his/her satisfaction, the segment can be submitted by clicking on the Submit button, located at the bottom of the screen.

As mentioned above, we also tested the App for translation functionality. When users selected a segment in the translation mode, an empty text box was presented into which they had to type their translation using the phone's keyboard (see Fig. 3).

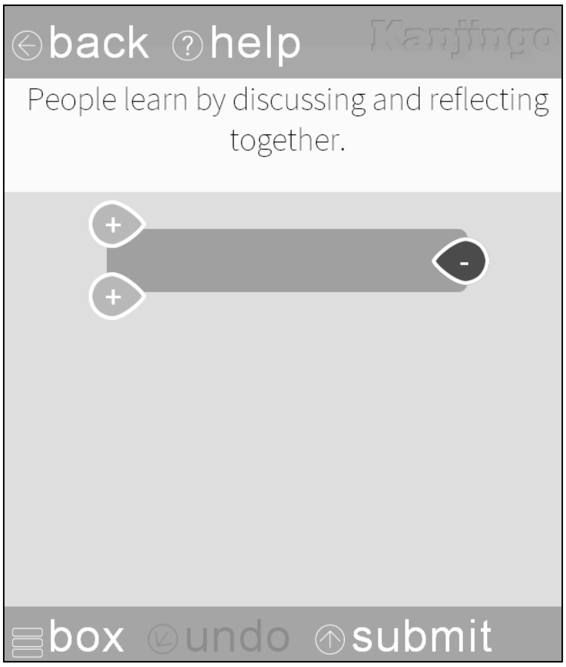

People learn by discussing and reflecting together.

Figure 3. The Kanjingo translation screen

2.3 Setup

Translators Without Borders provided us with sample content in source languages English and French, which were machine translated using Microsoft Bing Translator into French, Spanish, and English.

We recruited five users (2=male) with different backgrounds with the objective of including users of different profiles who were likely to fit the profiles of those who might volunteer to post-edit MT output. Their profiles are listed below with the language pair and direction they worked with listed in brackets.

- Professional translator who also has experience of post-editing in desktop scenarios (Fr-En);
- Research engineer who works with machine translation (En-Fr);
- PhD candidate who is currently researching audio-visual translation/fansubbing (En-Sp)
- Lecturer in language studies who has a Master's qualification in translation and interpreting (En-Sp)
- Master's student of translation, with undergraduate studies of translation (En-Sp).

Although this is a small group of users, this was an adequate number for initial user feedback. Nielsen has written that several iterations of usability testing at this scale maximizes the cost-benefit: “The best results come from testing no more than 5 users and running as many small tests as you can afford” (2000). The variation in profiles is also in keeping with best practice in UX testing.

Participants were requested to use concurrent Think Aloud Protocol (TAP), that is, to speak their thoughts about the task or the user interface during the task (Ericsson and Simon, 1980; 1999). Nielsen wrote that “thinking aloud may be the single most valuable usability engineering method” with some caveats, in that it may bias user behavior, and decrease productivity (1994, p195). This study, however, focuses on usability rather than productivity, so TAP was considered worthwhile, although in practice it transpired that “some test users have great difficulties in keeping up a steady stream of utterances as they use a system” (ibid., p196). Whatever TAP content was produced was transcribed and analyzed for comments that allowed us to identify the strong and weak points of the UI in both the post-editing and translation modes. Following the user interface test, participants were asked ten questions as part of a structured debriefing interview to help elucidate their evaluation of the App.

3 Results

The four users most familiar with smartphones were quickly able to edit the machine translated segments and had fairly positive attitudes towards the App in general, e.g. User 4 saw it as “ideal for short messages or perhaps emails with two or three sentences.” Several participants said that they found the App intuitive, with user 5 commenting that “I think it’s quite friendly, usable as well - easy-to-use.” Most participants were pleased with the drag-and-drop functionality. On the other hand, the user with least experience of smartphones struggled to use the App and disliked it more than the others. This user did not appear to understand the drag-and-drop functionality, despite having “accidentally reorganized the sentence without wanting to”, and found the App largely frustrating. She was one of several users who hit or touched buttons by mistake.

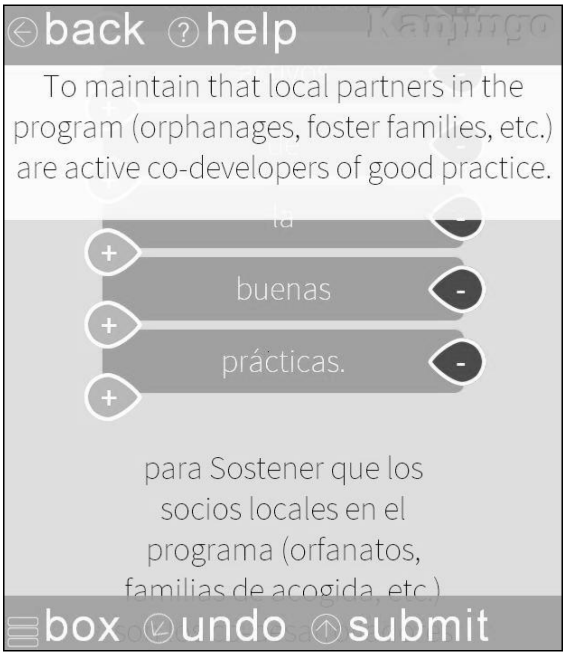
Accidental manipulation of the UI was one of several problems or frustrations encountered repeatedly during the tests. In summary, these were:

- The lack of automatic punctuation and capitalization
- Problems with sensitivity
- Loss of unsubmitted work if the user leaves the UI to check the Internet or dictionaries/glossaries
- Insufficient Help
- Input functionality challenges

Each of these issues is discussed in more detail below.

3.1 Punctuation and capitalization

Four of five participants voiced frustration at having to manually add capitals at the beginning of a segment and having to append a full stop at the end. When the capitalised word from the MT output was moved, this exacerbated the problem. This can be seen in Figure 4. In the next version of the App, any word moved to the top of the tiled list will be automatically capitalized and the full-stop will be attached only at the end of the segment, even if the last word is moved up.



To maintain that local partners in the program (orphanages, foster families, etc.) are active co-developers of good practice.

la

buenas

prácticas.

para Sostener que los socios locales en el programa (orfanatos, familias de acogida, etc.)

Figure 4. Incorrect capitalization.

3.2 Sensitivity

One of the main challenges in designing a smartphone App for text editing purposes is the limited space available to display the text. This problem is exacerbated when two segments have to be displayed within the UI. The challenge increases if the segments are “long”. The shortest segment in our content contained 3 words while the longest contained 20 words. The limited display led to issues regarding over- and under-sensitivity in the App. One user in particular had trouble hitting the plus and minus signs (see Figure 1 above). Some users accidentally tapped the ‘undo’ button when intending to ‘submit’ the segment that had been processed. Users also dragged word tiles accidentally when they simply meant to scroll up or down. Users mentioned that they wanted to group words and drag and drop them together, but this facility was unavailable. Our development team is investigating providing more space on either side of the segment display for scrolling and means of grouping words for combined drag and drop functionality.

3.3 Retention of unsubmitted work

Once a segment is edited, the users could submit it using the “Submit” button. One comment in relation to this was that they were unsure what had happened to the submitted segment because there was no confirmation message. A confirmation message (or other form of visual feedback) will therefore be added to the next version.

It may occur that a user is half-way through a segment and needs to abandon it for a period of time (the bus has arrived or an interesting message has popped up on Twitter!). The half-edited segment was then lost if the user toggled to another app. The development team is investigating ways of using the phone’s local storage to save edits in progress. However, we also need to consider what impact this has on progress if the user decides never to come back to the segments and they cannot be picked up by an alternative user.

3.4 Insufficient Help

We wished to investigate how intuitive the App was with only limited Help available to users. The level of Help available is depicted in Figure 2 above. It became obvious that the Help function in the App was inadequate. The user who was least familiar with smartphone apps, tried to avoid clicking ‘Help’ but eventually relented. Other users commented that they would like to see walk-though instructions before using the App for the first time.

3.5 Input Problems

Due to the limited area available, input was challenging. Particularly for the human translation scenario, input was a frustrating bottleneck between the users and the App. One participant commented that the speed for typing was so much slower than for a desktop scenario. The keyboard sometimes got in the way of the text box for translation. Another user complained because no synonyms or auto-completions were offered. For the translation scenario, one possible solution might be speech as input, though of course this is limited by the environment in which the translation takes place (e.g. if it is noisy, speech recognition may be compromised). We will also look at connecting the App to resources that would allow for synonym suggestion and auto-completion.

We were aware that the content we selected for this initial stage of testing was particularly challenging because (i) it was continuous text for which context was important and (2) some segments were relatively long. Limiting the length of segments would help solve the input problems, but this would also mean ruling out the use of the App for content that is typical to an organization like Translators Without Borders. Moreover, shorter segments bring their own challenges in respect of machine translation output quality and Tweets, or other forms of user-generated content, can also be difficult to decipher. Just limiting to short segments or Tweets is, therefore, not very desirable.

4 Conclusions

In embarking on this small-scale user testing of the mobile post-editing and translation App, Kanjingo, we expected a rather negative response from users given the severe space limitations of the mobile text-editing environment. However, although they were critical of certain aspects, the users were fairly positive about the App and gave some highly useful feedback. This feedback has been taken on board by the development team who are now in the process of developing a new version, for which we intend to do larger-scale user testing.

Future development work could potentially focus on interactive machine translation, speech input and Asian languages as target languages.

References

Ericsson, Anders K. and Herbert A. Simon. 1980. Verbal reports as data. *Psychological Review* 87, 215-251.

Ericsson, Anders K. and Herbert A. Simon. 1999. *Protocol Analysis: Verbal reports as data*. Cambridge, MA: MIT Press, 3rd edition.

Munro, Robert. 2010. Crowdsourced translation for emergency response in Haiti: the global collaboration of local knowledge. *AMTA 2010: The Ninth Conference of the Association for Machine Translation in the Americas*, October 31 - November 4, 2010, Denver, Colorado.

Nielsen, Jakob. 1994. *Usability Engineering.* Academic Press, Cambridge, MA.

Nielsen, Jakob. 2000. Why You Only Need to Test with 5 Users. Nielsen Norman Group. [Accessed from http://www.nngroup.com/articles/why-you-only-need-to-test-with-5-users/]

Petras, Rebecca. 2011. Localizing with community translation. *Multilingual 22(7),* 40-41.

Standard Language Variety Conversion for Content Localisation Via SMT

Federico Fancellu, Andy Way
Centre for Global Intelligent Content
School of Computing
Dublin City University
Dublin, Ireland.
ffancellu@cngl.ie,away@computing.dcu.ie

Morgan O'Brien
McAfee Ireland Inc
Citygate
Mahon
Cork, Ireland.
Morgan_O'Brien@mcafee.com

Abstract

Translation between varieties of the same language is a widespread reality in the localisation industry. However, monolingual statistical machine translation (SMT) is still a solution that has not yet been adequately explored; to the best of our knowledge, previous work in this area has never directly applied SMT to varieties of the same language for the precise purpose of reducing the time and cost of human translation and editing of content that needs to be localised.

In this paper, we start exploring the problem by deploying SMT to translate Brazilian Portuguese into European Portuguese. Our exploration mainly takes into consideration the use of bilingual dictionaries to guide the decoder and modify the translation output. We also consider the option of mining a bilingual dictionary from word alignments obtained after standard SMT training.

On good-quality data provided by Intel, we show that the SMT baseline already constitutes a strong system which in a number of experiments we fail to improve upon. We conjecture that bilingual dictionaries mined from client data would help if more heterogeneous training data were to be added.

1 Introduction

Localising content does not only involve translating across different languages, but often also translating between varieties of the same language. These varieties might differ in different repects, including spelling (e.g. British English *colour* vs. American English *colour*), lexicon (e.g. British English *autumn* vs. American English *fall*), word usage (e.g. British English *I'm pissed off* vs. American English *I'm pissed*), grammar (e.g. Irish English *You're after spilling my pint* vs. British English *You've just spilt my pint*), etc. Considering that often such translation tasks are carried out by humans, monolingual translation becomes costly and time-consuming, especially when one takes into account how much the two languages have in common.

Deploying Statistical Machine Translation (SMT, e.g. Koehn et al. (2003)) would appear to offer a solution to the problem. Given that the two varieties are essentially the same language except for some minor differences, we expect most of the translation variants to be captured by an SMT system. Moreover, we rely on the SMT system to be able to capture those structures that are not only acceptable in a language variety but are also *preferable*; in a rule-based system (RBMT), these could only be handled by complex hand-written rules.

The present study was run as a short-term (3∼4-week) innovation project between CNGL and Intel. The main goal of the present paper is to assess to what extent automatic methods can deliver a good translation between language varieties for localised content. For this reason, in the present study we refrain as much as possible from using any hand-crafted rules. After an initial SMT baseline was generated, we then explored (i) to what extent the system needed to be improved, and (ii) which techniques lead to the biggest improvement in translation quality and hence, decrease in human post-editing cost.

The language pair considered here is Brazilian Portuguese (*BP* henceforth) → European Portuguese (*EP*). Although Portuguese orthography was standardised in 1990, considerable differences remain between the two varieties in a number of linguistic respects, including pronoun (e.g. 2nd pers. pronoun → BP *você* vs. EP *você/tu*) and verb usage (e.g. BP loss of the pluperfect tense) and other lexical differences (e.g. PB *autocarro* vs. BP *ônibus*).

The remainder of this paper is organised as follows. In Section 2, we demonstrate that while same language translation can be of real benefit in a number of use-cases, at the same time, very little previous work appears to have been carried out. In Section 3, we describe the data used to build the various systems, and provide the results using a variety of techniques in Section 4. In Section 5, we discuss some of the pertinent findings, and conclude in Section 6 with some avenues for future work.

2 Same Language Translation

Despite the number of potential applications for same language translation, there are only a few works which address the problem. To the best of our knowledge, there is no published research which has directly applied SMT to translate from one language variety to another.

However, SMT has been applied for two related tasks in two of our own papers. In patented work described in Cahill et al. (2009), we built an English-to-English system using our in-house MaTrEx system (Tinsley et al. (2008)) to generate an N-best list of outputs that could be used for improved target-language speech synthesis. In Penkale and Way (2012), we addressed the problem of translating a bad version of a language into a 'less poor' one. This was in the context of translating in-game text, where incorrect English – usually written by a non-native game developer – needs to be improved prior to localisation *per se*; translating the poor original English 'as is' would produce completely unintelligible output. Using post-edited data as the target-side of the training data, our SMT system was able to learn how to automatically post-edit some of the errors made by the source authors, in much the same way as Dugast et al. (2007) and Simard et al. (2007) have shown previously.

While we are unaware of any published work on the subject, it is clear that Microsoft have done something similar, albeit for a different purpose. They describe their 'Contextual Thesaurus'[1] as "an English-to-English machine translation system that employs the same architecture that the Microsoft Translator uses when translating different languages". They list a number of applications for this "large-scale paraphrasing system", including document simplification, language learning, plagiarism detection, summarization and question answering, to name but a few.

As to non-statistical approaches, only Zhang (1998) appears to have applied RBMT to translate from Mandarin Chinese to Cantonese. Murakami et al. (2012) adopted instead a two-stage translation pipeline where Japanese is first rendered in English through pattern-based translation, which is in turn translated into more correct English. Formiga et al. (2012) focused on improving the output of an English-to-Spanish SMT system, where correct morphology is generated in a post-translation morphological generalisation stage.

As well as the use-cases presented already, the current paper addresses a number of real-world problems, which are as yet unsolved in the translation and localisation industries. Notwithstanding the need to come up with a proper treatment of terminology, we believe that some of the techniques utilised in our work can be brought to bear in addressing two other crucial problems, namely outdated legacy Translation Memories (TMs) and the introduction of new company terminology. As far as the first of these is concerned, companies typically prune data according to its age; clearly this is a very arbitrary solution. With respect to the second, new terminology presented in company glossaries may not tally with legacy (but still useful) TM data.

3 Data and System Building

The data were provided in the form of Intel TMs – BP-to-EN and EP-to-EN, where the English side was common to both – in the area of software documentation and customer support. As it was translated and validated by human experts, the data provided by Intel was of very good quality. However, before training the engines, any punctuation and markup 'noise' still left in the data was removed via regular expressions.

Two phrase-based SMT systems were built using Moses (Koehn et al. (2007)). The first (referred

[1] `http://labs.microsofttranslator.com/thesaurus/`

Approaches	BLEU	TER	METEOR
Baseline	**.589**	**0.292**	**0.704**
+ DNT (exclusive)	.588	**0.292**	**0.704**
+ DNT (constraint)	**.589**	**0.292**	**0.704**
+ LEX SUB1	.577	0.301	0.697
+ RUL1 (all)	.260	0.504	0.445
+ RUL1 (freq>5)	.524	0.327	0.658
+ RUL1 (freq>10)	.529	0.324	0.661
+ LEX SUB & dict from aligned data (constraint)	.578	0.30	0.70
+ post-decoding LEX SUB	.588	**0.292**	**0.704**

Table 1: System A: automatic evaluation scores for the different approaches.

to below as *System A*) was trained using 63,137 length-ratio filtered sentences (approx. 687,410 tokens). A devset of 1,498 sentences (approx. 20,286 tokens) was used to tune the weights for the features in the log-linear model using MERT (Och (2003)). In comparison, the second system (*System B*) was trained on a larger set of 75,324 sentences (approx. 828,532 tokens) using a different devset containing 1,499 sentences (approx. 20,174 tokens). For both systems we used a single test set comprising 1,500 sentences.

4 Methodology and Results

The main goal of the present paper is to show which approach (or combination of approaches) leads to the biggest improvement in translation quality. In more detail, we explored the following options:

1. Guiding decoding to ensure technical terms are translated correctly via supplied dictionaries,
2. Using lexical substitution to replace Brazilian Portuguese words remaining in the output,
3. Using data-driven spelling rules to correct the translation output,
4. Using company-internal and data-driven bilingual dictionaries to both guide decoding and correct the translation output.

The results for System A are shown in Table 1, while those for System B are shown in Table 2. Column 1 shows each of the different system variants built, with columns 2–4 showing the BLEU (Papineni et al. (2002)), Translation Edit Rate (TER: Snover et al. (2006)) and METEOR (Lavie and Denkowski (2009)) scores, respectively. Note that for BLEU and METEOR, the higher the score the better, while for TER, a lower score is indicative of better quality.

In the next sections, we describe each experiment conducted with the results achieved.

4.1 Translation of technical terms

Intel provided us with a list of technical and product names that the system should not mistranslate or lose during decoding. In order to adhere to their requirements, we wrapped those terms in xml tags (i.e. ⟨DNT⟩ . . . ⟨/DNT⟩) and used both the *exclusive* and *constraint* options implemented in Moses to guide decoding; *exclusive* forces the decoder to use a word input by the user as translation, while *constraint* allows the decoder to use *only* those phrases containing that word.

As seen from both Table 1 and Table 2, neither of the two options (*DNT (exclusive)* and *DNT (constraint)*) outperforms the Baseline; however, in Table 1, we see a small deterioration only for *DNT (exclusive)* in terms of BLEU, although more significant differences are seen in Table 2 for both options.[2] Accordingly, it might be said that these options do not appear to be too harmful, either. Forcing the decoder to select a specific word or phrase is likely to adversely impact the fluency of the translation which is otherwise ensured during phrase-based decoding (i.e. in the Baseline). Of course, in the majority of cases the baseline is able to translate these technical terms, merely by dint of these appearing in the TM from which the (correct) translations are learned; to us, this is not too surprising considering that the human translations on which the systems are trained are produced following strict guidelines. However, for many companies, correct rendering of terminology is of paramount importance and they are willing to sacrifice a small drop in (say) BLEU score as a trade-off; in practice, this deterioration in transla-

[2] Note that while it is surprising that the results in Table 2 are consistently lower, despite being trained on a larget data set, the results in Tables 1 and 2 are not directly comparable given that parameter estimation was performed on different devsets.

Approaches	BLEU	TER	METEOR
Baseline (w/ Intel content)	.583	0.295	0.695
+ DNT (exclusive)	.582	0.295	0.694
+ DNT (constraint)	.583	0.295	0.695
+ LEX SUB & dict from aligned data (constraint)	.571	0.305	0.685
+ post-decoding LEX SUB	.583	0.295	0.695

Table 2: System B: automatic evaluation scores for the different approaches.

tion quality is small enough to be of no real consequence to post-editors.

4.2 Lexical substitution

Here we used lexical substution as an attempt to replace words in the hypothesis translations that are still in Brazilian Portuguese. Here we assumed that the reference contains the correct *EP* variant, being human-translated material. We used an initial list of 982 item pairs provided by Intel. However, as shown in Table 1, this simple lexical substitution does not help translation, as words in the human-provided reference sentences do not tally with words described as 'European Portuguese' in the Intel lexicon. As an example, consider the dictionary items in (1):

(1) a. *mais→maior*
b. *confiança→considerar como fidedigno*

Now consult the behaviour in (2):

(2) a. EP reference: pode fazer compras com *mais confiança* em sites que passam os testes diários do serviço SECURE
b. EP translation baseline: pode efectuar compras com *maior confiança* em sites que passem os testes diários de Serviço SECURE
c. EP translation with lexical substitution: pode efectuar compras com *maior considerar como fidedigno* em sites que passem os testes diários de Serviço SECURE

As we can see, while the Baseline produces the correct form *maior* in (2b), it is penalised when compared to the reference in (2a). Furthermore, when we exercise the rule in (1b) to produce (2c) – as required by the Intel dictionary – we generate a translation which differs still further from (2a). Given this, it is perhaps surprising that this approach does not show large deteriorations in translation quality as measured by the automatic metrics in Table 1 (see line 5 'LEX SUB1'). However, we were convinced enough that relying only on such scores would not bring about translation improvements even on the larger set, so we omitted this experiment for System B.

4.3 Correcting the output using data-driven spelling rules

Another method to improve the quality of translation is to automatically extract spelling rules from the bilingual dictionary provided by Intel. These rules are then transformed into regular expression and applied to the test output *post hoc*. The algorithm takes into consideration each pair in the bilingual dictionary and sees which blocks differ and which operation has to apply in order to transform the source block into the target block. For instance, a *delete* type difference is detected between the pair in (3):

(3) BP:*detecção*→ EP: *deteção*

Consequently, we can extract a rule such as c→ø.

In order to exclude lexical differences (e.g. *assinatura→subscrição*) where block matching would yield rules that are not systematic (because they are not related to spelling differences), string-based similarity Levenshtein (1966) is calculated prior to rule extraction. If the pair has a similarity score greater than .6 (empirically determined), the rule is extracted.

At first we just extracted shallow rules resembling phonological rules which consider whether (i) the preceding or following letter is a vowel, (ii) the preceding or following letter is a consonant, and if so which consonant it is, and (iii) whether it is in sentence-initial or final position. For instance, a rule for *c*-deletion when preceded by a vowel and followed by ç is shown in (4):

(4) Vcç→Vøç.

To calculate improvement we then consider three different conditions: (i) *all*: all rules found are considered (RUL1 (all) in Table 1); (ii) *(freq.>5)*: all rules that were found more than 5 times are considered (RUL1 (freq>5)); and (iii) *(freq.>10)*: all rules that were found more than 10 times are considered (RUL1 (freq>10)).

Again, the results in Table 1 do not show any improvement across all metrics. What is especially clear (cf. RUL1 (all)) is that it makes sense to limit the application of the rules to those that were found many times if extremely low performance is to be avoided. One problem we detected with this approach was that some rules were *over-generalised* and could have been grouped more wisely.

Given the poor results of the current rule extraction algorithm, we considered a refinement whereby the context is first over-specified and then generalised if a lot of different contexts for the same target block are found. Consider the two rules in (5):

(5) a. (? <=s)ão$ → ø (lit. delete ão when preceded by s)

b. (? <=ç)ão$ → ø (lit. delete ão when preceded by ç)

We found some preceding context in common and so were able to merge both rules in (5) into the rule in (6):

(6) (? <=[sç])ão$ → ø

However, yet again this method did not lead to any further improvement. One of the reasons why poor-quality rules are extracted is because the input comprised misaligned data. For example, the rule in (7) tells us to delete word-final 's' if it is preceded by either a, p, e or o:

(7) (? <=[apeo])s$ → ø

This works correctly for strings such as (8a), but not for (8b), where the form is the same in both EP and BP:

(8) a. *relatório de atividades → relatório de atividade*

b. *log de atividades → log de atividades*

Furthermore, it applies to strings that it shouldn't: *dos→do*).

4.4 Company-internal vs. data-driven bilingual dictionary

As we showed in Section 4.2, using the glossary supplied by Intel didn't help improve translation performance owing to mismatches with the reference translations. While the results in the previous section were disapppointing, we considered it to have some potential. Accordingly, we extracted instead a bilingual dictionary (omitting function words) using alignment information computed during training. This alignment information was filtered *post hoc* using fine-grained POS-tagging and morphological analysis using Freeling (Padró and Stanilovsky (2012)) for Portuguese.

However, again we were again unable to improve over the Baseline. Nonetheless, this approach (LEX SUB & dict from aligned data (constraint)) produces slightly better quality translations according to all three automatic evaluation metrics than the original LEX SUB1 method. Performing this model in a post-decoding phase causes results to improve still further, with results matching the Baseline in Table 1 for both TER and METEOR, although the BLEU score lags behind a little. In Table 2, with the larger training set, we see exactly the same thing: TER and METEOR scores match the Baseline, with BLEU just a little lower.

5 Observations

The fact that no method implemented leads to two different hypotheses.

Firstly, the baseline models are already able to learn very strong translation patterns (i.e. words and phrases), such that there is little need for modifications to be made. All other methods we tried lead either to errors, or to paraphrases that are still correct but which are sufficiently different from the reference translation to be unfairly penalised. For instance, the sentences in (9) are grammatical and almost identical in meaning to the reference, but an *n*-gram overlap-based metric such as BLEU fails to reward the two sentences appropriately.

(9) a. *Reference*: contacto do suporte (online ou telefone)

b. *Baseline*: contacte o suporte (Online ou Telefonico).

c. *Lex sub w/ aligned data (constraint)*: entre em contacto com o suporte (online ou por telefone)

That the baseline already is able to recognise some inter-language patterns can be seen in (10) and (11), where the baseline system is able to translate the *bp* construction *estar + gerundive* vs. *ep estar a + infinitive*:

(10) a. *Source*: [...] descobrimos que ele pode **estar tentando** vender algo que normalmente [...]

b. *Baseline*: [...] verificamos que pode **estar a tentar** vender algo que , [...]

c. *Lex sub w/ aligned data (constraint)*: *same as Baseline*

(11) a. *Source*: [...] este arquivo **esteja sendo** usado [...]

b. *Baseline*: [...] este ficheiro **está a ser utilizado** [...]

c. *Lex sub w/ aligned data (constraint)*: *same as Baseline*

Secondly, the reason why the basic model is able to learn translation patterns to a consistently high level is because all the material is from the same domain and of good quality, seeing as it is human translated and validated. We hypothesise here that if more heterogenous material were to be used for training (out-of-domain, possibly containing some errors, e.g. emanating from a 'light' post-edit), then lexical substitution based on the aligned data is likely to lead to an improvement over the baseline.

6 Conclusion and Future Work

In this paper, we bootstrapped a Brazilian Portuguese-to-European Portuguese SMT system from Intel TMs where the English side was common to both. We demonstrated that the performance of the Baseline engines was so strong that an array of techniques could not bring about any improvement as measured by three mainstream automatic evaluation metrics. Accordingly, what is essential is that a human evaluation be carried out, to see which translations are actually preferred by users. Given that the SMT system is producing a score of nearly 0.6 BLEU points on a large test set, our experience tells us that this may be immediately deployed in Intel with productivity gains for post-editors likely to be of the order of double their human translation throughput. Of course, this too needs to be verified, and the cost savings calculated once the engine is deployed in Intel's translation workflow.

Given that the methods used are language-independent, it can also be extended to other language variety pairs; those of immediate interest to Intel include ES-to-ES-xx and FR-to-FR-CA. Moreover, we have shown that applying language variety conversion can go far beyond simple content localisation, although for a large player like Intel, already helping just this use-case is likely to lead to significant savings.

As well as these topics, we aim to investigate whether deploying similar pre-processing techniques on the training data itself *before* engine building can lead to improved translation output. If successful, this will have important consequences for companies owning large amounts of legacy TM data, who will subsequently be able to curate their data sets in a more informed manner than is currently the case.

Acknowledgements

This work was partially supported by the Science Foundation Ireland (Grant 07/CE/I1142) as part of CNGL at Dublin City University. Thanks also to Loic Dufresne de Virel at Intel, and Steve Gotz at CNGL, for their support on this project.

References

Cahill, P., Du, J., Berndsen, J., and Way, A. (2009). Using same-language machine translation to create alternative target sequences for text-to-speech synthesis. In *Proceedings of Interspeech 2009, the 10th Annual Conference of the International Speech Communication Association*, pages 1307–1310, Brighton, UK.

Dugast, L., Senellart, J., and Koehn, P. (2007). Statistical post-editing on SYSTRANs rule-based translation system. In *ACL 2007: Proceedings of the Second Workshop on Statistical Machine Translation*, pages 220–223, Prague, Czech Republic.

Formiga, L., Hernández, A., Mariño, J., and Monte, E. (2012). Improving English to Spanish out-of-domain translations by morphology generalization and generation. In *Proceedings of the Monolingual Machine Translation Workshop – AMTA 2012*, pages 6–16, San Diego, CA.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., and Bojar, O. (2007). Moses: Open source toolkit for statistical machine translation. In *ACL 2007: proceedings of demo and poster sessions*, pages 177–180, Prague, Czech Republic.

Koehn, P., Och, F. J., and Marcu, D. (2003). Statistical phrase-based translation. In *HLT-NAACL 2003: conference combining Human Language Technology conference series and the North American Chapter of the Association for Computational Linguistics conference series*, pages 48–54, Edmonton, Canada.

Lavie, A. and Denkowski, M. (2009). The meteor metric for automatic evaluation of machine translation. *Machine Translation*, 23(2-3):105–115.

Levenshtein, V. (1966). Binary Codes Capable of Correcting Deletions, Insertions and Reversals. *Soviet Physics Doklady*, 10:707–710.

Murakami, J., Nishimura, T., and Tokuhisa, M. (2012). Two stage machine translation system using pattern-based MT and phrase-based SMT. In *Proceedings of the Monolingual Machine Translation Workshop – AMTA 2012*, pages 31–40, San Diego, CA.

Och, F. J. (2003). Minimum error rate training in statistical machine translation. In *ACL-2003: 41st Annual meeting of the Association for Computational Linguistics*, pages 60–167, Sapporo, Japan.

Padró, L. and Stanilovsky, E. (2012). Freeling 3.0: Towards wider multilinguality. In *Proceedings of the Language Resources and Evaluation Conference (LREC 2012)*, Istanbul, Turkey. ELRA.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 311–318, Philadelphia, Pennsylvania.

Penkale, S. and Way, A. (2012). SmartMATE: An online end-to-end MT post-editing framework. In *Proceedings of AMTA 2012 Workshop on Post-editing Technology and Practice*, 10pp., San Diego, CA.

Simard, M., Ueffing, N., Isabelle, P. and Kuhn, R. (2007). Rule-based translation with statistical phrase-based post-editing. In *ACL 2007: Proceedings of the Second Workshop on Statistical Machine Translation*, pages 203–206, Prague, Czech Republic.

Snover, M., Dorr, B., Schwartz, R., Micciulla, L., and Makhoul, J. (2006). A study of translation edit rate with targeted human annotation. In *AMTA 2006: Proceedings of the 7th Conference of the Association for Machine Translation in the Americas, Visions for the Future of Machine Translation*, pages 223–231, Cambridge, Massachusetts, USA.

Tinsley, J., Ma, Y., Ozdowska, S., and Way, A. (2008). MaTrEx: the DCU MT system for WMT 2008. In *ACL-08: HLT. Third Workshop on Statistical Machine Translation, Proceedings (ACL WMT-08)*, pages 171-174, Columbus, Ohio, USA.

Zhang, X. (1998). Dialect MT: a case study between Cantonese and Mandarin. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics-Volume 2*, pages 1460–1464, Montreal, Quebec, Canada.

Collaborative Web UI Localization, or How to Build Feature-rich Multilingual Datasets

Vicent Alabau
PRHLT Research Center
Universitat Politècnica de València
valabau@prhlt.upv.es

Luis A. Leiva
PRHLT Research Center
Universitat Politècnica de València
luileito@prhlt.upv.es

Abstract

We present a method to generate feature-rich multilingual parallel datasets for machine translation systems, including e.g. type of widget, user's locale, or geolocation. To support this argument, we have developed a bookmarklet that instruments arbitrary websites so that casual end users can modify their texts on demand. After surveying 52 users, we conclude that people is leaned toward using this method in lieu of other comparable alternatives. We validate our prototype in a controlled study with 10 users, showing that language resources can be easily generated.

1 Introduction

Today most websites are looking forward to making their contents available in more than one language, mainly to reach a global audience, to gain a competitive advantage, or just because of legal requirements. To this end, adapting user interface (UI) texts through translation—or "localization"—is a central task, since its result affects system usability and acceptability. Actually, translation is just one of the activities of localization yet the most important overall (Keniston, 1997).

Recently there have been significant improvements in machine translation (MT) technology, to the extent that, in particular contexts such as medical prescriptions or knowledge-base articles, machine-translated content is qualitatively comparable to that of human-translated (Dillinger and Laurie, 2009). However, for MT systems to excel at UI localization not only it is needed an important amount of training data, but also the data must be especially tailored to the particularities of UI messages. Indeed, translating the text in an interface is a challenging task, even for trained human translators (Muntés-Mulero et al., 2012).

Parallel data offer a rich source of additional knowledge about language, and a sound basis for both translation and contrastive studies (McEnery and Xiao, 2007). Although there are some valuable tools to build multilingual parallel corpora, they are still limited when it comes to the exploitation of UI-based resources. Thus, we propose a novel approach: delegating the corpus generation to the end users of software applications, as a result of a regular interaction with such applications. To support our approach, we developed a proof-of-concept web-based prototype, motivated by the fact that nowadays people use web browsers more than any other class of software. Moreover, software translation poses two interesting challenges: *1)* user interface (UI) strings appear anywhere in the developer's language of choice whereas content is typically generated and consumed in the user's language; *2)* UI bilingual sentences can be enriched with metadata to handle disambiguation.

2 Related Work

In the past, several methods have been developed to build parallel corpora by automatic means, e.g., by mining Wikipedia (Smith et al., 2010), web pages with a similar structure (Resnik and Smith, 2003), parliament proceedings (Koehn, 2005), or using specialized tools such as OPUS (Tiedemann, 2012). However, in the end, parallel texts are scarce resources, limited in size and language coverage (Munteanu and Marcu, 2005).

In addition, many tools such as Crowdin[1], SmartLing[2], and Launchpad[3] do support collaborative translation. However, for these tools to work properly, applications must be internationalized beforehand. Besides, Google Translator Toolkit[4] allows contributing with translations. However,

[1] http://www.crowdin.net

[2] http://www.smartling.com

[3] http://translations.launchpad.net

[4] http://translate.google.com/toolkit/

the proposed translations are not rendered on the web page unless one uses the Website Translator tool *and* owns the site. Furthermore, it is oriented to translating *content* and not UI elements such as buttons, drop-down lists, etc. that otherwise may carry valuable language information.

Probably, the closest work in soul to ours is Duolingo (von Ahn, 2013), an effort to collaboratively translate the Web while users are learning a language. However, we are interested in providing computer users with a means of editing the text of any website on demand, only when it is needed.

More importantly, current tools force users to switch and use said tools, which may prevent them from contributing. Also, user contributions are not shown until the application owner decides to do so, thus hindering collaboration. Therefore, we feel another collaborative translation method is needed.

3 User Survey

We prepared a 2-question survey in order to identify to what extent would users be motivated to translate or edit translations in a computer application or a website. The first question (**Q1**) asked the preference degree to using 4 different methods:

1. **M1**: Editing the application source code.
2. **M2**: Installing a dedicated tool.
3. **M3**: The application features a menu option.
4. **M4**: Editing text in-place, at runtime.

The second question (**Q2**) asked the willingness to personalize the texts displayed in an application, provided that there were an easy method to do it. We included example images for each instance case, and answers to **Q1** were randomly presented to the users, to avoid possible biases. Both questions were scored in a 1–5 Likert scale (1: strongly disagree, 5: strongly agree). The survey was then released online via Twitter, Facebook, and word-of-mouth communication. Eventually, 52 users (24 females) aged 19–34 from 5 countries (USA, UK, France, Spain, and Germany) participated in the survey. The results are shown in Table 1.

	M1	M2	M3	M4	Q2
M	1.79	2.37	3.27	4.58	4.27
Mdn	2	2	3	5	4
SD	0.98	0.97	0.95	0.82	0.88

Table 1: Detailed survey results.

As observed, a preference for in-place runtime translation (**M4**) is evident over the rest of the considered options. Installing dedicated software (**M2**) is not seen as a likable approach, and even less editing the source code of the application (**M1**). On the other hand, having a translation facility bundled with the application (**M3**) is a significant enhancement. This is somewhat already implemented in most Linux programs, e.g., the official GNOME image viewer,which allows users to seamlessly collaborate worldwide to translate the program. Nevertheless, as previously pointed out, **M4** seems to be the most comfortable option.

Regarding the willingness to personalize texts (**Q2**), as expected, people are favorably predisposed to do so if they were given an easy-to-use method such as the one we are proposing. Together with the previous answers, this survey reveals that our method would allow regular computer users to (indirectly) contribute with translations. This suggests in turn that occasional users of an application or arbitrary visitors of a website are more likely to submit a translation pair, which would dramatically facilitate corpus construction, both in terms of human effort and time.

4 Method Overview

Apparently, users are eager to contribute with translations when they can instantaneously personalize their applications and the collaboration effort has a low entry cost. Thus, we propose a method were translations are carried out *just-in-time* and *in-place*. First, just-in-time implies that a translation takes place at the very same moment that the user needs it. For instance, when a user spots a sentence that has not been translated into her language, or a translation error is bothering her, she is simply able to amend the text on the UI. Second, in-place editing means that translation is performed on the same UI, not in another application, so that the overhead introduced by task switching has minimal impact. This localization strategy has shown some advantages over more traditional methods (Leiva and Alabau, 2014).

The core idea of our method is adapting the behavior of UI widgets so that they can switch to an *edit mode* when some accelerator is used. Note that the application should work as it was originally designed, however the behavior of the widgets would change only on demand (see Figure 1). While in theory this could be incorporated to any major UI library (e.g., Qt, GTK, MFC, Co-

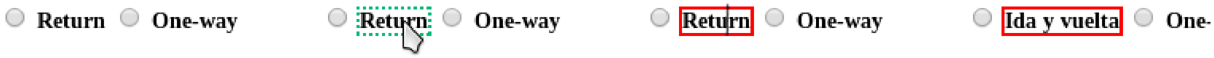


Figure 1: Example of *edit mode*. While `CTRL` is pressed, elements are highlighted as the mouse hovers them. Then, the user clicks on the element, which becomes editable, in order to change its content.

coa), in this paper we test a method that is suitable for web-based UIs. For simplicity, the method is deployed as a bookmarklet (no installation, just drag-and-drop, available for all browsers), which is more compatible than using extensions or plugins. The method can be roughly summarized as follows: *1)* a welcome menu is shown when clicking on the bookmarklet; *2)* resource strings are automatically extracted in the original language from text nodes, `alt` attributes, `form` elements, etc. along with a unique identifier (XPath); *3)* user's previous translations, if any, are loaded and applied to the UI; *4)* event listeners to receive user interaction are attached to UI elements; *5)* when the user activates the *edit mode*, UI elements become *content-editable* items, or a modal window pops up as a fallback mechanism; *6)* user information is collected, such as locale, geolocation by IP, etc. *7)* finally, the user can submit her contributions by clicking again on the bookmarklet.

5 Evaluation

We performed a controlled evaluation to assess if our method was worth being deployed at a larger scale. Thus, we recruited 10 Spanish users with an advanced English level. Participants were told to translate while interacting with a small airline website (5 pages) and one section of the popular Wordpress platform. At the end of the session, users submitted their translations to our server.

In 5 minutes, 159 out of the 205 potentially translatable sentences were identified by the users. On average, each user contributed with 114 (*SD*=4) sentences. Not all sentences were translated because some of them only appear under special circumstances like error messages or hidden options in menus, whereas others have low saliency (e.g., a copyright notice). Figure 2a shows the histogram of sources with different translations. It can be observed that more than a half of the sources received multiple translations, while it was not unusual to have up to 4 different translations for each source. Conversely, Figure 2b shows the histogram of the number of times the most voted translation was indeed produced by the agreement of n users. It turns out that users showed full disagreement only on 24 sentences. For the other sentences, at least two users agreed at any time. In addition, we can see a peak when 9 and 10 users agreed. This is explained in part because some sources were fairly simple to translate (such as navigation links) and thus it was expected that users would submit similar translations.

In general, users reported that they were happy to test our method for translating web pages. They felt the technique was easy to use, and expressed an intention to contribute with translations for their favorite applications. Hence, it seems plausible that a larger scale deployment would be successful.

6 General Discussion

Our method allows users to achieve an immediate benefit, since the website is being adapted to their language needs as they contribute to translating (and personalizing) it. At the same time, researchers also benefit from these contributions, since valuable language resources are being generated in the long run. Further, the method leads to having multiple references for a given source text, coming from different users worldwide, which allows for better training and evaluation of MT systems. More importantly, resources are ultimately supervised by humans—which provides valuable ground truth data—and can be deployed for potentially *any* language. Last but not least, our method enables "contextualized translation", in the sense that additional metadata are coupled to the traditional source-target language pairs, such as the type of widget (e.g., button, label, etc.), geolocation, locale, or the user agent string.

The survey gave us intuition regarding whether regular users would engage to contribute with casual translations. Nevertheless, as in any collaborative tool, the user needs a motivation to carry out any task. We believe that our proposal adds great value to how users experience computer software since, right from the beginning, they can fix translation errors and personalize their favorite applications. In contrast to other approaches where

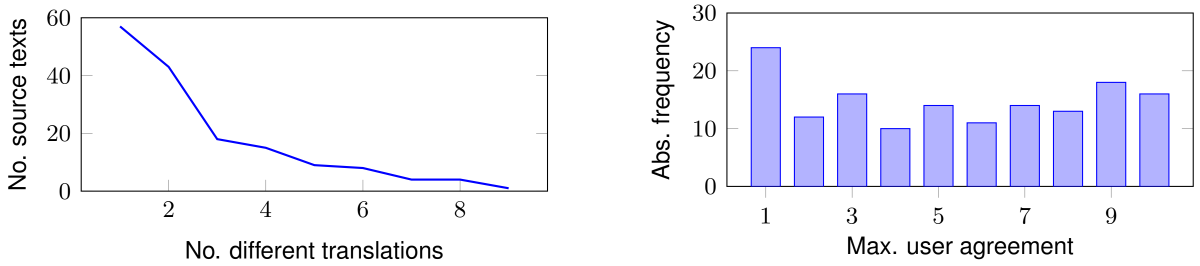


Figure 2: Distribution of different translations per source (2a) and histogram of user agreements (2b).

the user contributions are used to merely collect data, here these contributions are rendered immediately on the UI, so the benefit becomes instantaneous. Besides, as more and more data are collected, they can be used to initially populate a web page or application with the consensus translations from other users. This is especially interesting for minority languages, where a few users with knowledge of said minority language can make the UI accessible to the rest of users. Also, information reported by the browser can provide translations tailored to the user context, e.g., country or operating system. Hopefully, the low entry cost of our approach will reduce the burden on the user and thus foster collaboration.

In addition, the language resources that our method is able to collect provide unprecedented value for the MT community. First, potentially any language with a representative user base can generate parallel data. What is more, sentence pairs are properly aligned, since they come from the very same UI element, and multiple references may be available. Furthermore, translations are performed with a visual context. Thus, not only the chances that translations are appropriate will improve, but also language resources can be tagged with feature-rich metadata. For instance, the type of UI element (e.g., paragraph, button, link) or the text of a header or a label that relates to it, all can be used as additional information to provide better disambiguation in MT (Muntés-Mulero et al., 2012). Even so, personal information—if available and always under the user consent—can provide resources for adaptation of general models to specific dialects, or to target different age groups.

Acknowledgments

Work supported by the FP7 of the European Commission under grant agreements 287576 (CASMACAT) and 600707 (tranScriptorium).

References

Dillinger, M. and G. Laurie. 2009. Success with machine translation: automating knowledge-base translation. ClientSide News.

Keniston, Kenneth. 1997. Software Localization: Notes on Technology and Culture. Working Paper #26, Massachusetts Institute of Technology.

Koehn, Philipp. 2005. Europarl: A Parallel Corpus for Statistical Machine Translation. In *Proc. MT Summit*, pages 79–86.

Leiva, Luis A. and Vicent Alabau. 2014. The impact of visual contextualization on ui localization. In *Proc. CHI*, pages 3739–3742.

McEnery, Anthony and Zhonghua Xiao, 2007. *Incorporating Corpora: Translation and the Linguist*, chapter Parallel and comparable corpora: What are they up to? Multilingual Matters.

Munteanu, Dragos Stefan and Daniel Marcu. 2005. Improving Machine Translation Performance by Exploiting Non-Parallel Corpora. *Computational Linguistics*, 31(4):477–504.

Muntés-Mulero, V., P. Paladini Adell, C. España-Bonet, and L. Màrquez. 2012. Context-Aware Machine Translation for Software Localization. In *Proc. EAMT*, pages 77–80.

Resnik, Philip and Noah A. Smith. 2003. The Web as a parallel corpus. *Computational Linguistics*, 29(3):349–380.

Smith, Jason R., Chris Quirk, and Kristina Toutanova. 2010. Extracting parallel sentences from comparable corpora using document level alignment. In *Proc. NAACL*, pages 403–411.

Tiedemann, Jörg. 2012. Parallel Data, Tools and Interfaces in OPUS. In *Proc. LREC*, pages 2214–2218.

von Ahn, Luis. 2013. Duolingo: learn a language for free while helping to translate the web. In *Proc. IUI*, pages 1–2.

Using the ACCEPT framework to conduct an online community-based translation evaluation study

Linda Mitchell
Symantec Ltd.
Ballycoolin Business Park
Dublin 15, Ireland
linda_mitchell@symantec.com

Johann Roturier
Symantec Ltd.
Ballycoolin Business Park
Dublin 15, Ireland
johann_roturier@symantec.com

David Silva
Symantec Ltd.
Ballycoolin Business Park
Dublin 15, Ireland
david_silva@symantec.com

Abstract

This paper presents how a novel evaluation framework was used to collect translation ratings thanks to users of an online German-speaking support community in the IT domain. Using an innovative data collection approach and mechanism, this paper shows that segment-level ratings can be collected in an effective manner. The collection mechanism leverages the ACCEPT evaluation framework which allows data collection to be triggered from online environments in which community users interact on a regular basis.

1 Introduction

While machine translation is becoming ubiquitous in making Web content accessible to users who do not necessarily understand the language in which this content was first authored, some doubts remain about the ability of machine translation systems to generate content that is sufficiently fluent to be easily understood. Collecting translation evaluation ratings or judgments is often done in dedicated evaluation environments which fail to take into account any ecological validity requirements that are inherent to user-focused settings where translated content is made available. The approach presented in this paper addresses this issue by leveraging the evaluation framework provided by the ACCEPT project and by designing an evaluation task aimed at being as self-contained, self-explanatory, intuitive and engaging as possible for the target population of raters. The paper is organized as follows: Section 2 briefly reviews existing evaluation frameworks Section 3 focuses on the setup of an evaluation task that maximizes user engagement. Section 4 presents some statistics on the evaluation data collected during a 4-week timeframe. Finally, Section 5 contains some preliminary conclusions and suggestions for future work.

2 Related work

Numerous (machine) translation evaluation systems exist. These systems can be grouped into four categories: standalone desktop-based systems, Web-based dedicated systems, Web-based generic systems and Web-based hybrid systems. The first type of system (standalone desktop-based system such as Costa (Chatzitheodorou, 2013) is not relevant for studies involving online community members because it is inconvenient to ask users to install an application to provide ratings. Web-based dedicated systems such as Appraise (Federmann, 2012) or the Dynamic Quality Framework (DQF) Tools[1] may be useful to collect judgments from well-known contributors but they are not suited to collect genuine user feedback (mainly because they require a separate account creation and login process which can be cumbersome for users whose first priority is to consume content rather than evaluate content). Web-based generic systems such as Amazon Mechanical Turk (Callison-Burch, 2009) or Crowdflower[2] suffer from the same problem as dedicated systems. While they can be useful in generating large data volumes in a short period of time using a crowdsourcing approach, they also require a separate login process. Web-based hybrid systems,

[1] https://evaluation.taus.net/tools

[2] http://www.crowdflower.com/

such as Google forms,[3] offer maximum flexibility because the project management and data collection processes are decoupled from the actual data generation process. One such system is made available by the ACCEPT framework, whereby the project management process is completed using the evaluation section of the ACCEPT portal while the data generation process is performed using a JavaScript widget that is injected in an online community environment (an online discussion forum). Since the Evaluation Framework is built under a RESTful API endpoint,[4] this architecture enables the API to be used from any device that can make use of the HTTP protocol, regardless of the technology used for it. The following section provides more detail on the actual implementation of the widget.

3 Experimental setup

A previous study (Mitchell and Roturier, 2012) collected user ratings in an online forum context using a similar approach. However, their approach did not yield a large number of ratings for three main reasons:

- Users were asked to rate translations at the post level (which made the task quite cumbersome),
- The evaluation task was not immediately visible in the online forum environment (i.e. users had to click a number of times to find the content to evaluate),
- The evaluation task was not sufficiently engaging: once a rating was submitted, users were thanked but were not automatically presented with additional content to evaluate.

The present study tried to address these shortcomings by using an improved approach. Ratings would be collected at the segment level instead of the document level; the client-side evaluation widget would be positioned in a prominent place on the online forum (i.e. on the right-handside of the landing page); the widget would offer users a new segment to rate after receiving a rating; the widget would keep track of user ratings in order to avoid asking users to rate the same segment twice.

In order to be able to collect evaluation data on the client side, an evaluation project had to be defined using the Web interface of the ACCEPT portal.

Once the project was created, an API key was automatically generated and associated with this project. The next step was to define a question category, which was used as a container for a specific question. An example of such a category would be *fluency*, where all associated questions would be related to the fluency of text. Once the category was defined, the question that should be answered by users was added. A question has multiple attributes besides the question text itself (which could be "How fluent is this translated content?"):

- A language (e.g. English if the Question text is in English),
- An Action text (which may be used to instruct users how to submit an answer),
- An Action confirmation text (which may be used to show users that their answer has been submitted).

Finally, any number of answers may be added to a question. Each answer has two parts: the actual answer text to present to the user and a value (e.g. *Perfect* and *5*).

Evaluation content was then added to the project. The source content had been extracted from one of Symantec's English forums. The evaluation was based on 50 segments (which were not necessarily complete sentences), 25 of which had been machine-translated from English to German using the ACCEPT SMT system.[5]. The remaining 25 segments were machine-translated segments that had been post-edited by community users. These segments were selected from a pool of 1,700 segments, which had received fluency scores by 3 to 4 authoritative evaluators (e.g. localisation or technical support experts) prior to this evaluation task, as described in (Mitchell et al., 2014). They were selected based on the criterion that all of the evaluators agreed on the score for fluency (on a categorical scale from 1-5.[6] For each category between 1 and 5, 10 segments were selected. The content was uploaded to an evaluation

[3] http://www.google.com/google-d-s/createforms.html

[4] http://en.wikipedia.org/wiki/Representational_state_transfer

[5] http://www.accept.unige.ch/Products/D_4_1_Baseline_MT_systems.pdf

[6] 1=incomprehensible, 2=disfluent, 3=non-native, 4=good, 5=perfect

project on the ACCEPT evaluation portal[7] using a JSON file based on the following format:

```
{ "chunkList": [{
    "chunk":"Alle Resourcen sagen,
    dass diese Infektion nur auf
    PCs zutrifft und bieten
    Lösungen für PCs an.",
    "chunkInfo":"",
    "active":1
    },
]}
```

Listing 1: JSON format used to upload content that should be evaluated

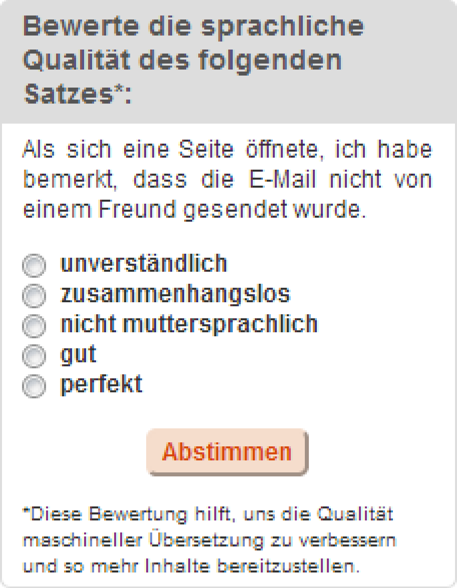
On the client side, the JavaScript client shown in Figure 1 was pre-configured with:

- The public API key generated during the process of creating the evaluation project,
- The ID of the category that should be presented to users (i.e. *fluency*),
- Under the category, the ID identifying the question and answer options that should be displayed.

During the Web page's loading process, the above configuration is used to get the necessary information from the REST API and dynamically build on-the-fly a Web form containing the specified question and the set of possible answers in a format of a radio buttons list, as shown in Figure 1. When the page is fully loaded, a form is displayed and the evaluation process can take place. Once the user chooses and submits an answer by clicking "Abstimmen", the form is serialized and sent over the Web by issuing a POST request against an API method.[8] The payload of this request may contain the ID of the question the user is answering, the API key used to identify the request, the ID of the chosen answer, the content (text) being evaluated and optional meta-data such as the IP address of the client, the ID the user if it can be found on the Web page, etc.[9]

4 Data analysis

During a four-week period, 1470 ratings were collected as shown in Table 1.



Bewerte die sprachliche Qualität des folgenden Satzes*:

Als sich eine Seite öffnete, ich habe bemerkt, dass die E-Mail nicht von einem Freund gesendet wurde.

○ unverständlich
○ zusammenhangslos
○ nicht muttersprachlich
○ gut
○ perfekt

*Diese Bewertung hilft, uns die Qualität maschineller Übersetzung zu verbessern und so mehr Inhalte bereitzustellen.

Figure 1: Client-side widget

Category	Number	%
Incomprehensible	457	31
Disfluent	208	14
Non native	387	26
Good	270	19
Perfect	148	10

Table 1: Evaluation Ratings

The ratings received were submitted by 171 users in total, of which 143 were unregistered users and 28 registered community members. We did not expect these users do have a strong bias towards machine translation since this technology does not pose any apparent threat to their profession. The average number of ratings per user session was eight ratings. The fifty segments received 29 ratings on average (from 8 to 100).

To get an overview of the heterogeneity of the ratings and to identify to what extent evaluators deviated from the average score per segment, a sampling strategy was employed. Segments 23 and 24 were selected - they had received 53 and 54 ratings with a mean of 4.43 and 1.33 as a score, respectively.

Samples from these ratings were selected in 5% increments, from 10% to 95% of all ratings received for a particular segment. For instance, if a segment had received 50 ratings, a 10% sample contained 5 ratings. For each increment, 20 samples were built randomly and the average was calculated based on these, which were then subse-

[7] http://www.accept-portal.eu

[8] http://www.ProjectName-portal.com/AcceptApiStg/Api/v1/Evaluation/ScoreFormPost

[9] http://www.accept.unige.ch/Products/D5.3_Adapted_Evaluation_Portal_Prototype.pdf

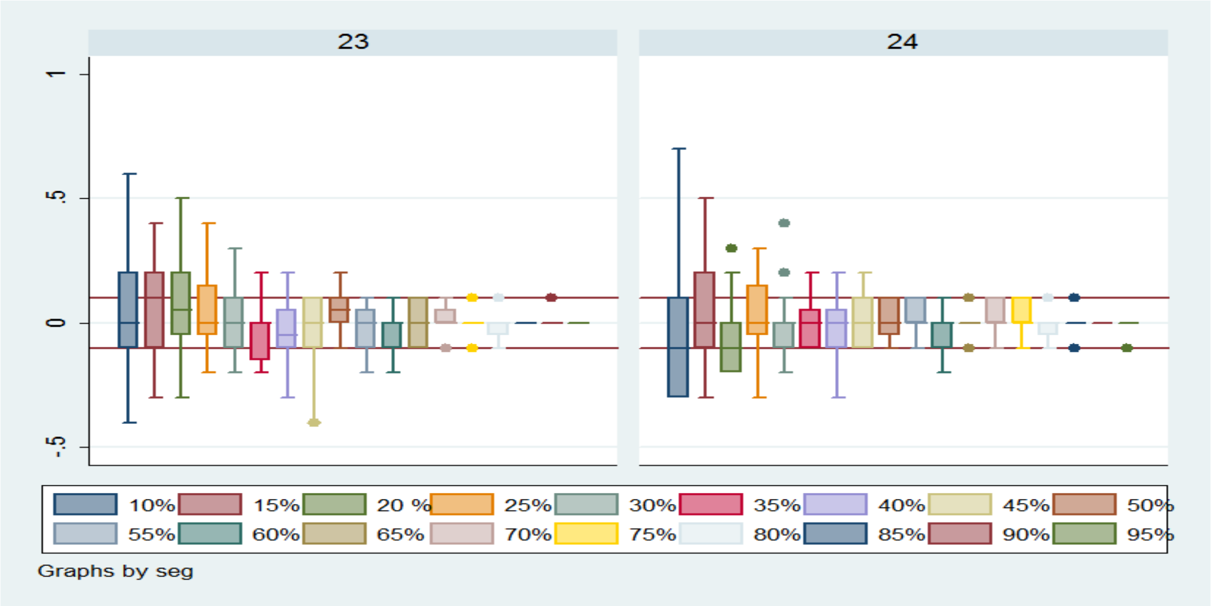


Figure 2: Average score variation for various sample sizes of user ratings

quently compared to the overall average. Figure 2 shows the results from the sampling for two of the segments. The y-axis represents how much the average of the samples per increment differs from the overall average. The box represents one standard deviation in each direction and contains 50% of all data points for each of the increments. The whiskers cover two standard deviations. Outliers are represented as dots. As expected, the larger the samples, the smaller the extent to which the average deviates from the overall average. It can be seen that in both cases to be able to achieve ratings using samples that will always be within 0.1 of the overall average (represented by the horizontal lines), 65% (35 ratings) of the ratings had to be sampled. 15 segments in total had at least one outlier. For 8 of the post-edited segments, outliers deviate from the average on average by 2.62. For 9 of the machine-translated segments, outliers deviate from the average on average by 2.17.

5 Conclusions and future work

This paper has shown that the ACCEPT framework can be used to set up community-based translation evaluation tasks. Such tasks maximize the ecological validity of the ratings obtained because they tend to be provided by users who are used to interacting with the system in which the client-side widget is deployed. While the present study focused on collecting ratings about the fluency of machine-translated and post-edited segments, future work will investigate how adequacy ratings could be obtained in a similar manner. This work will involve targeting users who have some knowledge of the source language. We are also interested in finding out whether the present data collection process can be further optimized by automatically identifying when a sufficient number of ratings has been obtained for a given segment.

References

Callison-Burch, Chris. 2009. Fast, Cheap, and Creative: Evaluating Translation Quality Using Amazon's Mechanical Turk. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, number August, pages 286–295, Singapore.

Chatzitheodorou, Konstantinos. 2013. COSTA MT Evaluation Tool: An Open Toolkit for Human Machine Translation Evaluation. *The Prague Bulletin of Mathematical Linguistics*, 100(1):83–89.

Federmann, Christian. 2012. Appraise: an Open-Source Toolkit for Manual Evaluation of MT Output. *The Prague Bulletin of Mathematical Linguistics*, 98:130–134.

Mitchell, Linda and Johann Roturier. 2012. Evaluation of Machine-Translated User Generated Content: A pilot study based on User Ratings. In *Proceedings of EAMT 2012*, pages 61–64, Trento, Italy.

Mitchell, Linda, Sharon O'Brien, and Johann Roturier. 2014. Quality Assessment in Community Post-Editing. *Machine Translation*. Forthcoming.

Handling Technical OOVs in SMT

Mark Fishel and **Rico Sennrich**
Institute of Computational Linguistics
University of Zurich
Binzmühlestr. 14
CH-8050 Zürich
{fishel,sennrich}@cl.uzh.ch

Abstract

We present a project on machine translation of software help desk tickets, a highly technical text domain. The main source of translation errors were out-of-vocabulary tokens (OOVs), most of which were either in-domain German compounds or technical token sequences that must be preserved verbatim in the output. We describe our efforts on compound splitting and treatment of non-translatable tokens, which lead to a significant translation quality gain.

1 Problem Setting

In this paper we focus on statistical machine translation of a highly technical text domain: software help desk tickets, or put simply – bug reports. The project described here was a collaboration between the University of Zurich and Finnova AG and aimed at developing an in-domain translation system for the company's bug reports from German into English. Here we present a general description of the key project results, the main problems we faced and our solutions to them.

Technical texts like bug reports present an increased challenge for automatic processing. In addition to having a highly specific lexicon, there is often a large amount of source code snippets, form and database field identifiers, URLs and other "technical" tokens that have to be preserved in the output without translation – for example:

Ger:	siehe auch ecl_kd042_de_crm_basis MP-MAR-11, kapitel 9.2.1.1
Eng:	see also ecl_kd042_de_crm_basis MP-MAR-11, chapter 9.2.1.1

While these technical tokens need no translation, our baseline system also suffers from a large number of out-of-vocabulary tokens (OOVs) that should be translated. The concatenative morphology of German compounds is a classical problem for machine translation, as it leads to an increased vocabulary and exacerbates data sparsity (Koehn and Knight, 2003). In our case the problem is inflated due to the domain-specific compound terms like *Tabellenattribute* (table attribute) or *Nachbuchungen* (subsequent postings): many of these are not seen in the smaller in-domain parallel corpus and they are too specific to be present in general-domain corpora.

Technical tokens like URLs and alphanumeric IDs do not require translation and should be transferred into the output verbatim. However, since they are also unknown to the translation system, they still present a number of problems. They are often broken by tokenization and not restored properly by subsequent de-tokenization. Also, splitting a technical token into several parts might result in the internal order of those parts broken. Even tokens that are correctly preserved in their original form can cause problems: if they are unknown to the language model, the model strongly favours permutations of the output in which OOVs are grouped together.

In the following section we give a description of our project and baseline system. We then turn to the problem of OOVs, and focus on handling the technical tokens that require no translation in Section 3, and on compound splitting strategies in Section 4. Experimental results constitute Section 5.

2 Translating Help Desk Tickets

The aim of our project was to develop an in-domain translation system for translating help desk

Token Type	Regular Expression	Examples
DB and form field IDs	`[A-Z0-9][A-Z0-9_/-]*[A-Z0-9]`	BEG_DAT_BUCH
numbers	`-?[0-9]+([.’][0-9]+)?`	-124.30, 1’000
UNIX paths and URLs	`([^ ():]*/){2,}[^ ():]*`	/home/user/readme.txt
code with dots, e.g. java	`[^ :.]{2,}(\.[^ :.]{2,})+`	java.lang.Exception

Table 1: Examples of technical tokens and regular expressions for their detection.

tickets from German to English for use in a post-editing work-flow.

The company had a set of manual translations from the target domain, which enabled us to use statistical machine translation (SMT). The in-domain parallel corpus composed of these translations consisted of 227 000 parallel sentences (2.8 / 3.2 million German/English tokens). Additional monolingual English data for the same domain was also available (141 000 sentences, 1.9 million tokens). As a baseline we used the Moses framework (Koehn et al., 2007) with settings identical to the baseline of WMT shared tasks (Bojar et al., 2013).

To increase the vocabulary of the system we added some publicly available general-domain and out-of-domain parallel corpora: Europarl (Koehn, 2005), OPUS OpenSubtitles (Tiedemann, 2012) and JRC-Acquis (Steinberger et al., 2006). Each of these is at least 10 times bigger than our in-domain corpus. To prefer in-domain translations in case of ambiguity, we combined all the available corpora via instance weighting using TMCombine from the Moses framework (Sennrich, 2012).

Despite the vast amount of general-domain data, the improvement over an in-domain system is relatively small: from 21.9 up to 22.3 BLEU points.[1] This best confirms that our target domain is highly specific. In fact, general-domain data actually hurts translation performance if its size is greater and no domain adaptation is performed: a simple concatenation of the same corpora without weighting causes a drop in translation quality to 21.3 BLEU points.

A post-editing set-up with our translation system resulted in an average efficiency gain of 30% over a pure translation work-flow, raising the number of ticket translations per hour from 4.5 to 5.9. In the next sections, we describe further attempts to improve translation quality by addressing different types of OOVs in the system.

3 Preserving Technical Tokens

The main problems with technical tokens that do not require translation are preserving their orthography and internal order, and placing them at the correct position in a sentence.

Most of these tokens are highly regular, which means that they can be detected with regular expressions and handled separately. We designed a set of regular expressions for that purpose and tagged them with the type of tokens that they detect. Table 1 presents some examples of the regular expressions and detected tokens. 8.8% of the tokens are identified as “technical”, with the largest group being upper-case database and form field IDs (4.0% of the tokens) and numbers (1.6% of the tokens).

We use XML mark-up to mark all technical tokens (consequently referred to as *masking*), and pass masked tokens unchanged through all components of our translation pipeline, i.e. the tokenizer, lowercaser, and the Moses decoder. While masking ensures that the masked tokens themselves are preserved, their position in the output is determined by the decoder. We observed that the n-gram language model that we use for decoding is poor at modelling the position of unknown words, preferring translation hypotheses where unknown words are grouped together, often at the beginning or end of the sentence.

As a solution to this issue, we change the translation pipeline as follows:

- the input text is tokenized and the detected technical tokens are reduced to a single constant token `__TECH__`.
- the translation is done on reduced text; the phrase table, lexical reordering and the language model are trained on corpora with reduced technical expressions.
- after the translation step, the reduced expressions are restored based on the input text and the word alignment between the input and the output, which is reported by the decoder.

[1] Measured on a test set of 1000 randomly held-out sentences, detokenized and re-cased.

This way, the original form of the technical tokens is preserved explicitly, and the feature functions of the translation pipeline do not have to deal with additional unknown input (the approach will be referred to as *1-token reduction*).

An alternative variant we explored is to represent each token sequence with its type (like `JAVA`, `DATE`, `URL`, etc.) instead of a single token `TECH`. A higher level of detail could be useful to model differences in word order between different kinds of technical tokens. Also, in case a sentence contains maskable tokens of different types, this reduces the number of duplicate tokens between which the model cannot discriminate (this alternative will be referred to as *type reduction*).

4 Compound splitting

The German language has a productive compounding system, which increases vocabulary size and exacerbates the data sparsity effect. Many compounds are domain-specific and are unlikely to be learned from larger general-domain corpora. Compound splitting, however, has the potential to also work on our in-domain texts.

We evaluate two methods of compound splitting. Koehn and Knight (2003) describe a purely data-driven approach, in which frequency statistics are collected from the unsplit corpus, and words are split so that the geometric mean of the word frequencies of its parts is maximized. Fritzinger and Fraser (2010) describe a hybrid approach, which uses the same corpus-driven selection method to choose the best split of a word among multiple candidates, but instead of considering all character sequences to be potential parts, they only consider those splits that are validated by a finite-state morphology tool.

The motivation for using the finite-state morphology is to prevent linguistically implausible splittings such as *Testsets* → *Test ETS*. We use the Zmorge morphology (Sennrich and Kunz, 2014), which combines the SMOR grammar (Schmid et al., 2004) with a lexicon extracted from Wiktionary.[2] With this hybrid approach, we only consider nouns for compound splitting; with the data-driven approach on the other hand we have no control over which word classes are split.

Source:	erweiterung tabellen TX_VL und TXTSVL .
Reference:	extension of tables TX_VL and TXTSVL .
Masking:	extension of tables TX_VL TXTSVL and .
Reduction:	extension of tables TX_VL and TXTSVL .

Table 2: An example of the effect of reducing: the correct order of technical tokens is preserved.

5 Experiments and Results

We evaluated our experiments on a held-out in-domain test set. Translation quality is judged using the MultEval package (Clark et al., 2011) and its default automatic metrics (BLEU, TER and METEOR); the package implements the metrics and performs statistical significance testing to account for optimizer instability. We perform three independent tuning runs, and use 95% as the significance threshold. Statistically **non**-significant results are shown in italics. Since tokenization differs between experiments, we compare de-tokenized and re-cased hypothesis and reference translations.

As baseline, we use the weighted combination of in-domain and other corpora, described in Section 2. All modifications to tokenization and compound splitting are done on all included training corpora, both in-domain and others.

Masking the detected technical tokens yields large quality gains over default tokenization:

	BLEU	METEOR	TER
Baseline	22.3	26.1	62.2
Masking	25.1	27.6	56.8

The system with masking better matches the length of the reference translation than the baseline (99.5% vs. 103.7%); this can be attributed to the technical tokens being broken in the baseline and not fixed by the default de-tokenization.

The reduced representation of technical tokens brings a small improvement:

	BLEU	METEOR	TER
Just masking	25.1	27.6	56.8
1-token reduction	25.5	27.7	56.4
Type reduction	25.4	27.7	*56.6*

A manual inspection supports the hypothesis that the reduced representation improves word order for sentences with multiple OOVs; see Table 2 for an example. Representing the expressions with their type, however, does not seem to have any ad-

[2] `http://www.wiktionary.org`

ditional effect: statistically it is indistinguishable from 1-token reduction.

Compound splitting yields gains of 0.8–1 BLEU when evaluated separately from technical token reduction:

	BLEU	METEOR	TER
Just masking	25.1	27.6	56.8
Data-driven split	26.1	28.9	55.1
Hybrid split	25.9	28.6	55.4

In contrast to the results reported by Fritzinger and Fraser (2010), we observe no gains of the hybrid method over the purely data-driven method by Koehn and Knight (2003). We attribute this to the fact that domain-specific anglicisms such as *Eventhandling* (event handling) and *Debugmeldung* (debug message) are unknown to the morphological analyzer, but are correctly split by the data-driven method.

Finally, we obtain the best system by combining masking, 1-token reduction and data-driven segmentation.

	BLEU	METEOR	TER
Just masking	25.1	27.6	56.8
1-token reduction	25.5	27.7	56.4
Data-driven split	26.1	28.9	55.1
Full combination	26.5	29.0	54.1

To conclude, we have shown that the modelling of OOVs has a large impact on translation quality in technical domains with high OOV rates. Overall we observed an improvement of 4.2 BLEU, 2.9 METEOR and 8.1 TER points over the baseline.

In this paper, we focused on two types of OOV tokens: German compounds that can be split into their components, and technical tokens that need no translation. While our modelling of both these types was successful both individually and in combination, in the general case the handling of different types of OOVs are not necessarily independent steps. Also, additional strategies for handling OOVs may be required in other domains and language pairs, e.g. transliteration of named entities. Robustly choosing the right strategy for each OOV token independently of the domain could be the target of future research.

References

Bojar, Ondřej, Christian Buck, Chris Callison-Burch, Christian Federmann, Barry Haddow, Philipp Koehn, Christof Monz, Matt Post, Radu Soricut, and Lucia Specia. 2013. Findings of the 2013 Workshop on Statistical Machine Translation. In *Proceedings of the Eighth Workshop on Statistical Machine Translation*, pages 1–44, Sofia, Bulgaria.

Clark, Jonathan H, Chris Dyer, Alon Lavie, and Noah A Smith. 2011. Better hypothesis testing for statistical machine translation: Controlling for optimizer instability. In *Proceedings of the 49th ACL*, pages 176–181, Portland, Oregon, USA.

Fritzinger, Fabienne and Alexander Fraser. 2010. How to avoid burning ducks: Combining linguistic analysis and corpus statistics for German compound processing. In *Proceedings of the Joint Fifth Workshop on Statistical Machine Translation and MetricsMATR*, pages 224–234, Uppsala, Sweden.

Koehn, Philipp and Kevin Knight. 2003. Empirical methods for compound splitting. In *Proceedings of the 10th EACL*, pages 187–193, Budapest, Hungary.

Koehn, Philipp, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: open source toolkit for statistical machine translation. In *Proceedings of the 45th ACL*, pages 177–180, Prague, Czech Republic.

Koehn, Philipp. 2005. Europarl: A parallel corpus for statistical machine translation. In *Proceedings of MT Summit X*, volume 5, pages 79–86.

Schmid, Helmut, Arne Fitschen, and Ulrich Heid. 2004. A German Computational Morphology Covering Derivation, Composition, and Inflection. In *Proceedings of the 4th LREC*, pages 1263–1266, Lisbon, Portugal.

Sennrich, Rico and Beat Kunz. 2014. Zmorge: A German morphological lexicon extracted from Wiktionary. In *Proceedings of the 9th LREC*, (in print), Reykjavik, Iceland.

Sennrich, Rico. 2012. Perplexity minimization for translation model domain adaptation in statistical machine translation. In *Proceedings of the 13th EACL*, pages 539–549, Avignon, France.

Steinberger, Ralf, Bruno Pouliquen, Anna Widiger, Camelia Ignat, Tomaz Erjavec, Dan Tufis, and Dániel Varga. 2006. The JRC-Acquis: A multilingual aligned parallel corpus with 20+ languages. In *Proceedings of the 5th LREC*, pages 2142–2147, Genoa, Italy.

Tiedemann, Jörg. 2012. Parallel data, tools and interfaces in OPUS. In *Proceedings of the 8th LREC*, pages 2214–2218, Istanbul, Turkey.

Oral Session 4
Research Papers

Using a New Analytic Measure for the Annotation and Analysis of MT Errors on Real Data

Arle Lommel[1], Aljoscha Burchardt[1], Maja Popović[1],
Kim Harris[2], Eleftherios Avramidis[1], Hans Uszkoreit[1]
[1]DFKI / Berlin, Germany
[2] text&form / Berlin, Germany
name.surname@dfki.de[1]
kim_harris@textform.com[2]

Abstract

This work presents the new flexible Multidimensional Quality Metrics (MQM) framework and uses it to analyze the performance of state-of-the-art machine translation systems, focusing on "nearly acceptable" translated sentences. A selection of WMT news data and "customer" data provided by language service providers (LSPs) in four language pairs was annotated using MQM issue types and examined in terms of the types of errors found in it.

Despite criticisms of WMT data by the LSPs, an examination of the resulting errors and patterns for both types of data shows that they are strikingly consistent, with more variation between language pairs and system types than between text types. These results validate the use of WMT data in an analytic approach to assessing quality and show that analytic approaches represent a useful addition to more traditional assessment methodologies such as BLEU or METEOR.

1 Introduction

For a number of years, the Machine Translation (MT) community has used "black-box" measures of translation performance like BLEU (Papineni et al., 2002) or METEOR (Denkowski and Lavie, 2011). These methods have a number of advantages in that they can provide automatic scores for MT output in cases where there are existing reference translations by calculating similarity between the MT output and the references. However, such metrics do not provide insight into the specific nature of problems encountered in the translation output and scores are tied to the particularities of the reference translations.

As a result of these limitations, there has been a recent shift towards the use of more explicit error classification and analysis (see, e.g., Vilar et al. (2006)) in addition to automatic metrics. The error profiles used, however are typically ad hoc categorizations and specific to individual MT research projects, thus limiting their general usability for research or comparability with human translation (HT) results. In this paper, we will report on annotation experiments that use a new, flexible error metric and that showcase a new type of MT research involving collaboration between MT researchers, human translators, and Language Service Providers (LSPs).

When we started to prepare our annotation experiments, we teamed up with LSPs and designed a custom error metric based on the "Multidimensional Quality Metric" MQM designed by the QTLaunchPad project (http://www.qt21.eu/launchpad). The metric was designed to facilitate annotation of MT output by human translators while containing analytic error classes we considered relevant to MT research (see **Section 2**, below). This paper represents the first publication of results from use of MQM for MT quality analysis.

Previous research in this area has used error categories to describe error types. For instance, Farrús et al. (2010) divide errors into five broad classes (orthographic, morphological, lexical, semantic, and syntactic). By contrast, Flanagan (1994) uses 18 more fine-grained error categories with additional language-pair specific features, while Stymne and Ahrenberg (2012) use ten error types of somewhat more intermediate granularity (and specifically addresses combinations of multiple error types). All of these categorization schemes are ad hoc creations that serve a particular analytic goal. MQM, however, provides a *general* mechanism for describing a family of related metrics that share a common vocabulary. This metric was based upon a rigorous examination of major human and machine translation assessment metrics (e.g., LISA QA Model, SAE J2450, TAUS DQF, ATA assessment, and various tool-specific metrics) that served as the basis for a descriptive framework for declaring what a particular metric addresses. While the metric described in this paper is still very much a purpose-driven metric, it is declared in this general framework, which we propose for use to declare specific metrics for general quality assessment and error annotation tasks.

For data, we chose WMT data (Bojar et al., 2013) to represent the state of the art output for MT in research. However, LSPs frequently reported to us that the mostly journalistic WMT data does not represent their business data (mostly technical documentation) or typical applications of MT in business situations. In addition, it turned out that journalistic style often contains literary flourishes, idiosyncratic or mixed styles, and deep embedding (e.g., nested quotations) that sometimes make it very difficult to judge the output.

As a result, we decided to use both WMT data and customer MT data that LSPs contributed from their daily business to see if the text types generate different error profiles. This paper accordingly presents and compares the results we obtained for both types of sources. For practical purposes, we decided to analyze only "near miss" translations, translations which require only a small effort to be converted into acceptable translations. We excluded "perfect" translations and those translations that human evaluators judged to have too many errors to be fixed easily (because these would be too difficult to annotate). We therefore had human evaluators select segments representing this especially business-relevant class of translations prior to annotation.

A total of nine LSPs participated in this task, with each LSP analyzing from one to three language pairs. Participating LSPs were paid up to €1000 per language pair. The following LSPs participated: Beo, Hermes, iDisc, Linguaserve, Logrus, Lucy, Rheinschrift, text&form, and Welocalize.

2 Error classification scheme

The Multidimensional Quality Framework (MQM) system[1] provides a flexible system for declaring translation quality assessment methods, with a focus on analytic quality, i.e., quality assessment that focuses on identifying specific issues/errors in the translated text and categorizing them.[2] MQM defines over 80 issue/error types (the expectation is that any one assessment task will use only a fraction of these), and for this task, we chose a subset of these issues, as defined below.

- **Accuracy**. Issues related to whether the information content of the target is equivalent to the source.
  - **Terminology**. Issues related to the use of domain-specific terms.
  - **Mistranslation**. Issues related to the improper translation of content.
  - **Omission**. Content present in the source is missing in the target.
  - **Addition**. Content not present in the source has been added to the target.
  - **Untranslated**. Text inappropriately appears in the source language.
- **Fluency**. Issues related to the linguistic properties of the target without relation to its status as a translation.
  - **Grammar**. Issues related to the grammatical properties of the text.
    - **Morphology (word form)**. The text uses improper word forms.
    - **Part of speech**. The text uses the wrong part of speech

[1] http://www.qt21.eu/mqm-definition/

[2] This approach stands in contrast to "holistic" methods that look at the text in its entirety and provide a score for the *as a whole* in terms of one or more dimensions, such as overall readability, usefulness, style, or accuracy. BLEU, METEOR, and similar automatic MT evaluation metrics used for research can be considered holistic metrics that evaluate texts on the dimension of similarity to reference translations since they do not identify specific, concrete issues in the translation. In addition, most of the options in the TAUS Dynamic Quality Framework (DQF) (https://evaluation.taus.net/about) are holistic measures.

    - ∗ **Agreement**. Items within the text do not agree for number, person, or gender.
    - ∗ **Word order**. Words appear in the incorrect order.
    - ∗ **Function words**. The text uses function words (such as articles, prepositions, "helper"/auxiliary verbs, or particles) incorrectly.
  - **Style**. The text shows stylistic problems.
  - **Spelling**. The text is spelled incorrectly
    - ∗ **Capitalization**. Words are capitalized that should not be or vice versa.
  - **Typography**. Problems related to typographical conventions.
    - ∗ **Punctuation**. Problems related to the use of punctuation.
  - **Unintelligible**. Text is garbled or otherwise unintelligible. Indicates a major breakdown in fluency.

Note that these items exist in a hierarchy. Annotators were asked to choose the most specific issue possible and to use higher-level categories only when it was not possible to use one deeper in the hierarchy. For example, if an issue could be categorized as *Word order* it could also be categorized as *Grammar*, but annotators were instructed to use *Word order* as it was more specific. Higher-level categories were to be used for cases where more specific ones did not apply (e.g., the sentence *He slept the baby* features a "valency" error, which is not a specific type in this hierarchy, so *Grammar* would be chosen instead).

3 Corpora

The corpus contains Spanish→English, German→English, English→Spanish, and English→German translations. To prepare the corpus, for each translation direction a set of translations were evaluated by expert human evaluators (primarily professional translators) and assigned to one of three classes:

1. **perfect (class 1)**. no apparent errors.
2. **almost perfect or "near miss" (class 2)**. easy to correct, containing up to three errors.
3. **bad (class 3)**. more than three errors.

Both WMT and "customer" data[3] were rated in this manner and pseudo-random selections (selections were constrained to prevent annotation of multiple translations for the same source segment within a given data set in order to maximize the diversity of content from the data sources) taken from the class 2 sentences, as follows:

- **Calibration set**. For each language pair we selected a set of 150 "near miss" (see below) translations from WMT 2013 data (Bojar et al., 2013).
  - For English → German and English → Spanish, we selected 40 sentences from the top-ranked SMT, RbMT, and hybrid systems, plus 30 of the human-generated reference translations.
  - For German → English and Spanish → English, we selected 60 sentences from the top-ranked SMT and RbMT systems (no hybrid systems were available for those language pairs), plus 30 of the human-generated reference translations.

- **Customer data**. Each annotator was provided with 200 segments of "customer" data, i.e., data taken from real production systems.[4] This data was translated by a variety of systems, generally SMT (some of the German data was translated using an RbMT system).

- **Additional WMT data**. Each annotator was also asked to annotate 100 segments of previously unannotated WMT data. In some cases the source segments for this selection overlapped with those of the calibration set, although the specific MT outputs chosen did not (e.g., if the SMT output for a given segment appeared in the calibration set, it would not reappear in this set, although the RbMT, hybrid, or human translation might). Note that the additional WMT data provided was different for each LSP in order to maximize coverage of annotations in line with other research goals; as such, this additional data does not factor into inter-annotator agreement calculations (discussed below).

[3] WMT data was from the top-rated statistical, rule-based, and hybrid systems for 2013; customer data was taken from a variety of in-house systems (both statistical and rule-based) used in production environments.

[4] In all but one case the data was taken from actual projects; in the one exception the LSP was unable to obtain permission to use project data and instead took text from a project that would normally not have been translated via MT and ran it through a domain-trained system.

It should be noted that in all cases we selected only translations for which the source was originally authored in the source language. The WMT shared task used human translations of some segments as source for MT input: for example, a sentence authored in Czech might be translated into English by humans and then used as the source for a translation task into Spanish, a practice known as "relay" or "pivot" translation. As we wished to eliminate any variables introduced by this practice, we eliminated any data translated in this fashion from our task and instead focused only on those with "native" sources.

3.1 Annotation

The annotators were provided the data described above and given access to the open-source translate5[5] annotation environment. Translate5 provides the ability to mark arbitrary spans in segments with issue types and to make other annotations. All annotators were invited to attend an online training session or to view a recording of it and were given written annotation guidelines. They were also encouraged to submit questions concerning the annotation task.

The number of annotators varied for individual segments, depending on whether they were included in the calibration sets or not. The numbers of annotators varied by segment and language pair:

- **German→English**: Calibration: 3; Customer + additional WMT: 1
- **English→German**: Calibration: 5; Customer + additional WMT: 1–3
- **Spanish→English**: Calibration: 4; Customer + additional WMT: 2–4
- **English→Spanish**: Calibration: 4; Customer + additional WMT: 1–3

After annotation was complete some post-processing steps simplified the markup and extracted the issue types found by the annotators to permit comparison.

3.2 Notes on the data

The annotators commented on a number of aspects of the data presented to them. In particular, they noted some issues with the WMT data. WMT is widely used in MT evaluation tasks, and so enjoys some status as the universal data set for tasks such as the one described in this paper. The available translations represent the absolute latest and most state-of-the-art systems available in the industry and are well established in the MT research community.

However, feedback from our evaluators indicated that WMT data has some drawbacks that must be considered when using it. Specifically, the text type (news data) is rather different from the sorts of technical text typically translated in production MT environments. News does not represent a coherent domain (it is, instead, a genre), but rather has more in common with general language. In addition, an examination of the human-generated reference segments revealed that the human translations often exhibited a good deal of "artistry" in their response to difficult passages, opting for fairly "loose" translations that preserved the broad sense, but not the precise details.

The customer data used in this task does not all come from a single domain. Much of the data came from the automotive and IT (software UI) domains, but tourism and financial data were also included. Because we relied on the systems available to LSPs (and provided data in a few cases where they were not able to gain permission to use customer data), we were not able to compare different types of systems in the customer data and instead have grouped all results together.

An additional factor is that the sentences in the calibration sets were much longer (19.4 words, with a mode of 14, a median of 17, and a range of 3 to 77 words) than the customer data (average 14.1 words, with a mode of 11, a median of 13, and a range of 1 to 50 words). We believe that the difference in length may account for some difference between the calibration and customer sets described below.

4 Error analysis

In examining the aggregate results for all language pairs and translation methods, we found that four of the 21 error types constitute the majority (59%) of all issues found:

- Mistranslation: 21%
- Function words: 15%
- Word order: 12%
- Terminology: 11%

None of the remaining issues comprise more than 10% of annotations and some were found so

[5] http://www.translate5.net

infrequently as to offer little insight. We also found that some of the hierarchical distinctions were of little benefit, which led us to revise the list of issues for future research (see **Section 4.2** for more details).

4.1 Inter-Annotator Agreement

Because we had multiple annotators for most of the data, we were able to assess inter-annotator agreement (IAA) for the MQM annotation of the calibration sets. IAA was calculated using Cohen's kappa coefficient. At the word level (i.e., seeing if annotators agreed for each word, we found that the results lie between 0.2 and 0.4 (considered "fair"), with an average of pairwise comparisons of 0.29 (de-en), 0.25 (es-en), 0.32 (en-de), and 0.34 (en-es), with an overall average of 0.30

4.2 Modifications

This section addresses some of the lessons learned from an examination of the MQM annotations described in **Section 4.1**, with a special emphasis on ways to improve inter-annotator agreement (IAA). Although IAA does not appear to be a barrier to the present analytic task, we found a number of areas where the annotation could be improved and superfluous distinctions eliminated. For example, "plain" *Typography* appeared so few times that it offered no value separate from its daughter category *Punctuation*. Other categories appeared to be easily confusible, despite the instructions given to the annotators (e.g., the distinction between "Terminology" and "Mistranslation" seemed to be driven largely by the length of the annotated issue: the average length of spans tagged for "Mistranslation" was 2.13 words (with a standard deviation of 2.43), versus 1.42 (with a standard deviation of 0.82) for "Terminology'.' (Although we had expected the two categories to exhibit a difference in the lengths of spans to which they were applied, a close examination showed that the distinctions were not systematic with respect to whether actual terms were marked or not, indicating that the two categories were likely not clear or relevant to the annotators. In addition, "Terminology" as a category is problematic with respect to the general-domain texts in the WMT data sets since no terminology resources are provided.)

Based on these issues, we have undertaken the following actions to improve the consistency of future annotations and to simplify analysis of the present data.

- The distinction between *Mistranslation* and *Terminology* was eliminated. (For calculation purposes *Terminology* became a daughter of *Mistranslation*.)
- The *Style/Register* category was eliminated since stylistic and register expectations were unclear and simply counted as general *Fluency* for calculation purposes.
- The *Morphology (word form)* category was renamed *Word form* and *Part of Speech*, *Agreement*, and *Tense/mood/aspect* were moved to become its children.
- *Punctuation* was removed, leaving only *Typography,* and all issues contained in either category were counted as *Typography*
- *Capitalization*, which was infrequently encountered, was merged into its parent *Spelling*.

In addition, to address a systematic problem with the *Function words* category, we added additional custom children to this category: *Extraneous* (for function words that should not appear), *Missing* (for function words that are missing from the translation), and *Incorrect* (for cases in which the incorrect function word is used). These were added to provide better insight into the specific problems and to address a tendency for annotators to categorize problems with function words as Accuracy issues when the function words were either missing or added. This revised issue type hierarchy is shown in Figure 1.

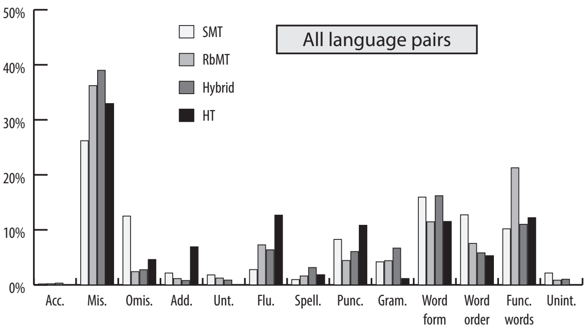


Figure 3: Average Sentence-level error rates [%] for all language pairs.

This revised hierarchy will be used for ongoing annotation in our research tasks. We also realized that the guidelines to annotators did not provide sufficient decision-making tools to help them select the intended issues. To address this problem

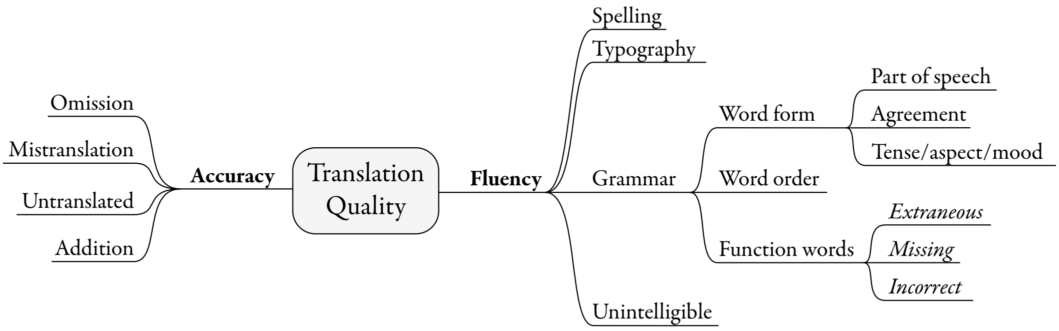


Figure 1: Revised issue-type hierarchy.

we created a decision-tree to guide their annotations. We did not recalculate IAA from the present data set with the change in categories since we have also changed the guidelines and both changes will together impact IAA. We are currently running additional annotation tasks using the updated error types that will result in new scores.

Refactoring the existing annotations according to the above description, gives the results for each translation direction and translation method in the calibration sets, as presented in Figure 2 (with averages across all language pairs as presented in Figure 3). Figure 4 presents the same results for each language pair in the customer data. As previously mentioned, we were not able to break out results for the customer data by system type.

4.3 Differences between MT methods

Despite considerable variation between language pairs, an examination of the annotation revealed a number of differences in the output of different system types. While many of the differences are not unexpected, the detailed analytic approach taken in this experiment has enabled us to provide greater insight into the precise differences rather than relying on isolated examples. The overall results for all language pairs are presented in Figure 3 (which includes the results for the human translated segments as a point of comparison).

The main observations for each translation method include:

- **statistical machine translation**
  - Performs the best in terms of *Mistranslation*
  - Most likely to drop content (*Omission*); otherwise it would be the most accurate translation method considered.
  - Had the lowest number of *Function Words* errors, indicating that SMT gets this aspect substantially better than alternative systems.
  - Weak in *Grammar*, largely due to significant problems in *Word Order*
- **rule-based machine translation**
  - Generated the worst results for *Mistranslation*
  - Was least likely to omit content (*Omission*)
  - Was weak for *Function Words*; statistical enhancements (moving in the direction of hybrid systems) would offer considerable potential for improvement
- **hybrid machine translation** (available only for English→Spanish and English→German)
  - Tends to perform in between SMT and RBMT in most respects
  - Most likely method to produce mistranslated texts (*Mistranslation*)

When compared to the results of human translation assessment, it is apparent that all of the near-miss machine translations are somewhat more accurate than near-miss human translation and significantly less grammatical. Humans are far more likely to make typographic errors, but otherwise are much more fluent. Note as well that humans are more likely to add information to translations than MT systems, perhaps in an effort to render texts more accessible. Thus, despite substantial differences, all of the MT systems are overall more similar to each other than they are to human translation. However, when one considers that a far greater proportion of human translation sentences were in the "perfect" category and a far lower proportion in the "bad" category, and that these comparisons focus only on the "near miss sentences," it

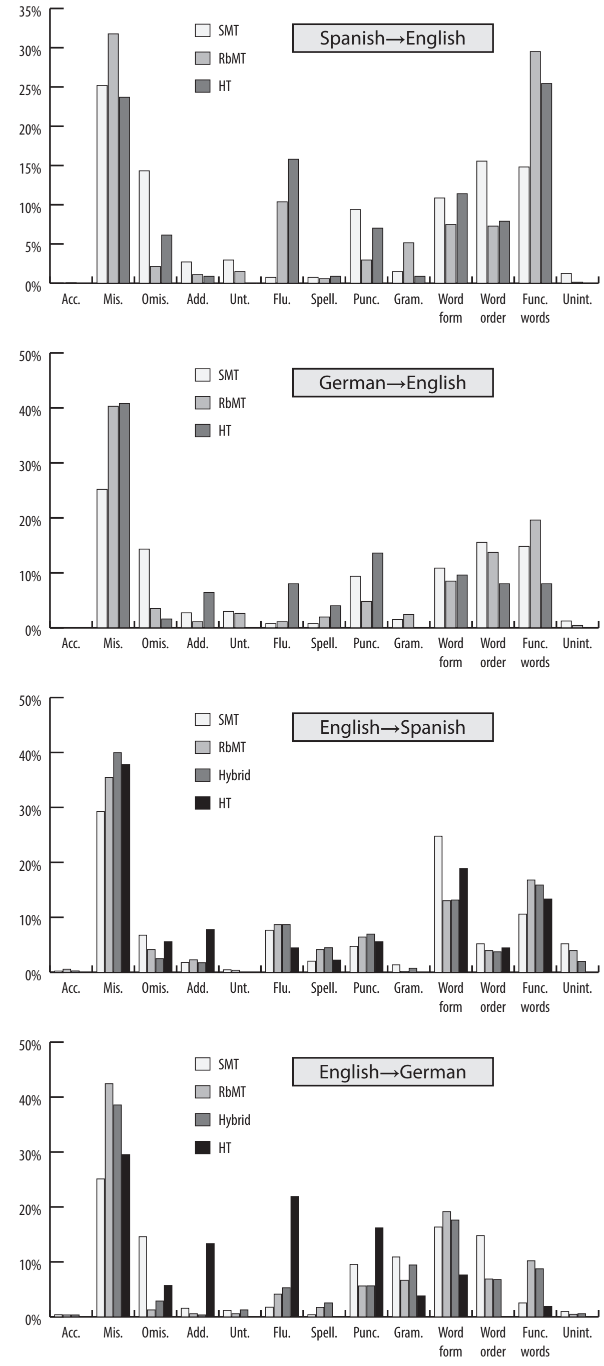


Figure 2: Sentence-level error rates [%] for each translation direction and each translation method for WMT data.

is apparent that outside of the context of this comparison, human translation still maintains a much higher level of Accuracy and Fluency.

In addition, a number of the annotators commented on the poor level of translation evident in the WMT human translations. Despite being professional translations, there were numerous instances of basic mistakes and interpretive translations that resulted in translations that would generally be considered poor references for MT evaluation (since MT cannot make interpretive translations). However, at least in part, these problems with translation may be attributed to the uncontrolled nature of the source texts, which tended to be more literary than is typical for industry uses of MT. In may cases the WMT source sentences presented translation difficulties for the human translators and the meaning of the source texts was not always clear out of context. As a result the WMT texts provide difficulties for both human and machine translators.

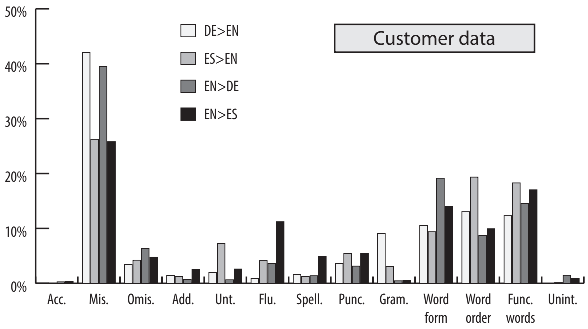


Figure 4: Sentence-level error rates [%] for each translation direction for customer data.

4.4 Comparison of WMT and customer data

By contrast, the customer data was more likely to consist of fragments (such as *Drive vibrates* or section headings) or split segments (i.e., one logical sentence was split with a carriage return, resulting in two fragments) that caused confusion for the MT systems. It also, in principle, should have had advantages over the WMT data because it was translated with domain-trained systems.

Despite these differences, however, the average profiles for all calibration data and all customer data across language pairs look startlingly similar, as seen in Figure 5. There is thus significantly more variation between language pairs and between system types than there is between the WMT data and customer data in terms of the error profiles. (Note, however, that this comparison addresses only the “near-miss” translations and cannot address profiles outside of this category; it also does not address the overall relative distribution into the different quality bands for the text types.)

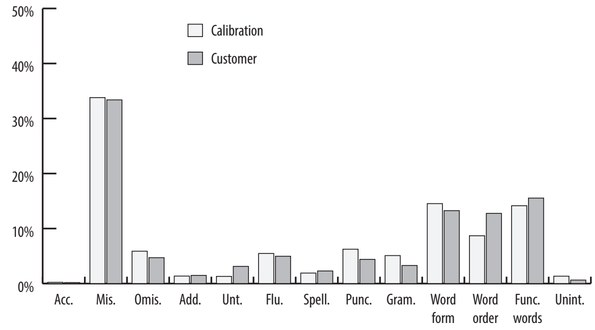


Figure 5: Sentence-level error rates [%] for calibration vs. customer data (average of all systems and language pairs).

5 Conclusions and outlook

The experiment here shows that analytic quality analysis can provide a valuable adjunct to automatic methods like BLEU and METEOR. While more labor-intensive to conduct, they provide insight into the causes of errors and suggest possible solutions. Our research treats the human annotation as the first phase in a two-step approach. In the first step, described in this paper, we use MQM-based human annotation to provide detailed description of the symptoms of MT failure. This annotation also enables us to detect the system type- and language-specific distribution of errors and to understand their relative importance.

In the second step, which is ongoing, linguists and MT experts will use the annotations from the first step to gain insight into the causes for MT failures on the source side or into MT system limitations. For example, our preliminary research into English source-language phenomena indicates that *-ing* verbal forms, certain types of embedding in English (such as relative sentences or quotations), and non-genitive uses of the preposition *of* are particularly contributory to MT failures. Further research into MQM human annotation will undoubtedly reveal additional source factors that can guide MT development or suggest solutions to systematic linguistic problems. Although many of these issues are known to be difficult, it is only with the identification of concrete examples that they can be addressed.

In this paper we have shown that the symptoms of MT failure are the same between WMT and customer data, but it is an open question as to whether the causes will prove to be the same. We therefore advocate for a continuing engagement with language service providers and translators using these different types of data. These approaches will help further the acceptance of MT in commercial settings by allowing them to be compared to HT output and will also help research to go forward in a more principled and requirements-driven fashion.

Acknowledgments

This work has been supported by the QTLaunchPad (EU FP7 CSA No. 296347) project.

References

Bojar, O., Buck, C., Callison-Burch, C., Federmann, C., Haddow, B., Koehn, P., Monz, C., Post, M., Soricut, R., and Specia, L. (2013). Findings of the 2013 workshop on statistical machine translation. In *Proceedings of the Eighth Workshop on Statistical Machine Translation*, pages 1–44.

Denkowski, M. and Lavie, A. (2011). Meteor 1.3: Automatic metric for reliable optimization and evaluation of machine translation systems. In *Proceedings of the EMNLP 2011 Workshop on Statistical Machine Translation*.

Farrús, M., Costa-jussà, M. R., Mariño, J. B., and Fonollosa, J. A. (2010). Linguistic-based evaluation criteria to identify statistical machine translation errors. In *Proceedings of the EAMT 2010*.

Flanagan, M. (1994). Error classification for MT evaluation. In *Proceedings of AMTA*, pages 65–72.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). BLEU: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 311–318.

Stymne, S. and Ahrenberg, L. (2012). On the practice of error analysis of machine translation evaluation. In *Proceedings of LREC 2012*, pages 1785–1790.

Vilar, D., Xu, J., D'Haro, L. F., and Ney, H. (2006). Error analysis of statistical machine translation output. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*, pages 697–702.

Complexity of Spoken Versus Written Language for Machine Translation

Nicholas Ruiz
University of Trento
Fondazione Bruno Kessler
Trento, Italy
nicruiz@fbk.eu

Marcello Federico
Fondazione Bruno Kessler
Trento, Italy
federico@fbk.eu

Abstract

When machine translation researchers participate in evaluation tasks, they typically design their primary submissions using ideas that are not genre-specific. In fact, their systems look much the same from one evaluation campaign to another. In this paper, we analyze two popular genres: spoken language and written news, using publicly available corpora which stem from the popular WMT and IWSLT evaluation campaigns. We show that there is a sufficient amount of difference between the two genres that particular statistical modeling strategies should be applied to each task. We identify translation problems that are unique to each translation task and advise researchers of these phenomena to focus their efforts on the particular task.

1 Introduction

The machine translation community has consistently used the translation of news texts and news commentaries as some of its prime methods of evaluating the progression of MT research. News translation evaluation tasks have existed since the first NIST evaluations in the early 2000s, followed by the Workshop on Machine Translation (WMT) (Bojar et al., 2013).

In recent years, TED talks have attracted the interest of the MT research community for measuring progress. The International Workshop on Spoken Language Translation (IWSLT) is currently in its fifth year of hosting TED talk evaluation campaigns, with a growing number of translation languages and participants (Cettolo et al., 2013). Both the WMT and IWSLT evaluations have enjoyed strong performance results that have progressively improved year after year and are established today as the de-facto evaluation tasks for text and speech translation, respectively. In practice, the top performing MT systems use many of the same training and decoding approaches in these evaluations. But are the WMT and IWSLT translation tasks just different flavors of the same translation problem? Are the strategies used to translate written language directly applicable to the genre of spoken language – in particular, prepared speeches?

This paper investigates the question of what makes MT difficult for speech corpora as opposed to text corpora. We try to understand the differences between the genres of news texts and prepared speeches, both in qualitative and quantitative terms. The ultimate goal is to find information that could explain differences in MT system performance and the types of errors occurring often in MT systems trained on text and speech corpora.

We begin by surveying some of the aspects of language that make MT hard and how they relate to the problem of human understanding of text (Sections 2 and 3). We follow up the discussion with a detailed analysis to determine if these aspects are distinctive of IWSLT or WMT, or are shared in common (Sections 4-7). We contrast WMT News Commentary texts with TED talks due to their similarity to the lecture genre. We analyze their characteristics and compare them both on a monolingual and a bilingual perspective. In the monolingual perspective, we look at the characteristics of the source language that make it difficult to process. In the bilingual perspective, we look at the problem of transferring content and structure from English to German. We follow-up with a small MT experiment, comparing the performance of TED and WMT News Commentaries on similar training conditions in Section 8. In Section 9 we recommend the suitable evaluation task for various research aspects of MT, and we summarize our findings in Section 10.

2 Challenges in human readability

The most commonly researched area of language complexity lies in the field of psycholinguistics. Much of the research focuses on language acquisition and generation by native speakers or second language learners and focus on a single language.

From the reader's end, extralinguistic information such as prior world knowledge and familiarity with a topic provide context that helps her understand a text. A text can activate this information through a variety of linguistic devices, such as anaphoric mentions and grounding. Additionally, the reader must be able to organize the information received from the text into coherent blocks. Readable texts typically have a number of qualities that assist the reader in processing the information, such as redundancy, favoring concrete references over abstract principles, restatements of unfamiliar concepts, and syntactic structures appropriate for the reading level of the intended audience.

Graesser et al. (1994) introduce a *coherence assumption*, which claims that readers routinely attempt to construct coherent meanings and connections among text constituents unless the quality of the text is too poor. This concept forms one the core hypotheses in the constructivist theory of discourse comprehension. As a result, many complexity analysis tools attempt to detect coherence and cohesion through syntax, semantic, and discourse connectives (Graesser et al., 2004; Mitchell et al., 2010; Newbold and Gillam, 2010).

Biber (1988) and follow-up work by researchers investigate the variation in cohesion across text and speech corpora. Louwerse et al. (2004) perform a multi-dimensional analysis to identify a number of linguistic features that divide the corpora along several registers. Their results show variance between speech and writing corpora on a variety of factors, including type frequency, polysemy, pronoun density, abstract noun usage, type-token ratios for nouns, and stem overlap. These features divide the written and spoken genres into subdomains posing unique challenges in comprehension (e.g. prepared speeches versus conversational speech; news broadcasts versus legal documents).

3 Language Complexity in Statistical Machine Translation

Specia et al. (2011) outline three categories for features used in the task of MT quality estimation: *confidence* indicators derived from SMT models, *complexity* indicators that measure the difficulty of translating the source text, and *fluency* indicators that measure the grammaticality of a translation. Likewise, the difficulty of a translation task can be estimated by analyzing source complexity and target language features that indicate the capacity of a statistical system to generate fluent translations.

We attempt to focus on complexity issues that are irrespective of a particular text, speaker, or language pair and focus on issues that are relevant to the MT task. We can categorize these issues into three general areas: the lexicon, syntax, and semantics. When considering the lexicon, we can observe effects of vocabulary size, morphological variations, and both lexical and translation ambiguity as key impacts affecting the ability of the statistical models to cover the words in the language (Carpuat and Wu, 2007). On the syntax level, sentence length, structure complexity, and structural dependencies affect the decoding search space. On the semantic level, phenomena such as idiomatic expressions, figures of speech, anaphora, and elliptical expressions define intrinsic limitations of syntactic models. While we can observe nearly all of these language features on the monolingual level, many of these issues have a greater impact when transferring linguistic information in the process of translation. Between distant language pairs, the effects of these linguistic features cause a cumulative increase in the difficulty of MT.

Although discourse-based machine translation takes into account intersentential factors affecting translation quality (Carpuat, 2009; Foster et al., 2010), the majority of SMT systems treat each sentence independently, ruling out additional context.

4 Research methodology

In this paper, we compare two sources of spoken and written language: TED talk transcripts[1] and News Commentary texts[2]. Both types of texts cover a variety of topics whose content is produced by several authors. Although these types of texts correspond to different genres, they are popular representatives of spoken and written language investigated in MT, while belonging to similar domains. Both genres consist of speakers or authors with similar communication goals: namely, the mass distribution of information and ideas delivered by subject matter experts. At the same time, TED speakers have the additional objective of selling ideas through persuasive speeches. We focus

[1] http://www.ted.com/talks

[2] http://www.statmt.org/wmt09/translation-task.html

Measure	TED-EN	WMT-EN	TED-DE	WMT-DE
Word Count	2000018	2000016	1890106	2046071
Line Count	103588	82256	103588	82256
Surface forms	46001	50129	86787	95922
Stems	34417	36904	62929	66735
Words/Line	19.31	24.31	18.25	24.87
Stem/Surface	0.748	0.736	0.725	0.696

Table 1: Statistics for two million word TED and WMT News Commentary corpora samples.

on the English-German language pair, which belong to the same language family, but have marked differences in levels of inflection, morphological variation, verb ordering, and pronoun cases.

In our experiments, we sample approximately two million words from both the English TED and WMT News Commentary corpora, as well as the German translations of their sentences. Rather than randomly sampling sentences from the corpora, we sequentially read the sample to allow us preserve the underlying discourse. Sentences containing more than 80 words are excluded. We additionally subdivide the sampled corpora into blocks of 100,000 words to measure statistics on vocabulary growth rate.

We use TreeTagger (Schmid, 1994) to lemmatize and assign part-of-speech tags using the Penn Treebank (Marcus et al., 1993) and STTS (Schiller et al., 1995) tagsets for English and German, respectively. Some simple corpora statistics are provided in Table 1.

5 Word statistics

5.1 Sentence length

Since the unconstrained search space in SMT is exponential with respect to the length of the source sentence, we examine the distribution of sentence lengths between the TED and WMT corpora, as shown in Figure 1. On average, TED consists of lines containing around 19 words, while WMT averages five more words per line. Forty percent of the sentences in TED have between six and 15 words, while the majority of the sentences in WMT contain over 20 words. This suggests that TED is less susceptible to length-dependent decoding issues such as long distance reordering.

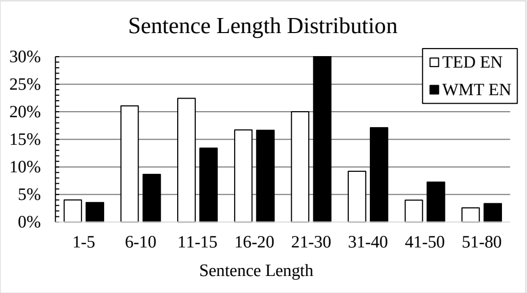


Figure 1: Sentence length statistics for English. TED talks favor shorter sentences.

5.2 Predictability: Perplexity and new words

Perplexity measures the similarity of n-gram distributions between a training set and a test set. Source and target language n-gram distributions govern a SMT system's capacity to adequately translate a sequence of words with its phrase table and language model (LM). Likewise, the out-of-vocabulary (OOV) rate estimates the amount of source words that are impossible to translate with the given training data. We measure these notions of complexity by constructing English and German language models and evaluating their predictive power against in-domain data. Using our 2 million word corpus samples, we incrementally add 100,000 words to each corpus and evaluate its perplexity and OOV rate against a held-out 100,000 word sample from each training corpus. Using IRSTLM (Federico et al., 2008), we construct trigram LMs, using improved Kneser-Ney smoothing, no pruning, and a fixed vocabulary size of 10 million words.

According to the results shown in Figure 2, TED consistently has lower trigram perplexity rates (-46% with the full data for English, -28% for German). We observe no significant differences in OOV between TED and WMT. The results suggest TED is more capable of being modeled than WMT with the same amount of training data and the translation of TED is more regular than the translation of WMT.

6 Lexical ambiguity

Two measurements of lexical ambiguity are word polysemy and translation entropy. We analyze the ambiguity of noun and verb lemmas, which as content words carry the most important information needed to understand a sentence. We only consider

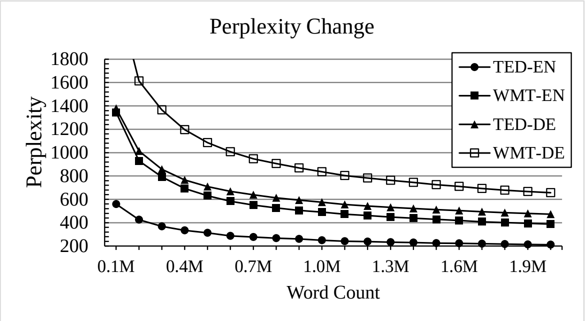


Figure 2: Perplexity change as corpus size increases for English and German.

the types that contain sense information in WordNet (Fellbaum, 1998). We take the top 100 lists of verbs and nouns from each corpus and measure their ambiguity, as described in the sections below. We compare the results against measurements on the full set of nouns or verbs and additionally measure the overlapping lemmas in the corpora.

6.1 Polysemy

As an upper-bound measure of word ambiguity, we measure the number of senses each English word in the corpus can express, as reported by WordNet. While not every sense may be observed in our corpora, this measure estimates how ambiguous a corpus is for a statistical system that considers each sense to be equally likely for a given word. Figure 3a provides a comparison between the top 100 verb and noun lemmas in the two corpora. On a global scale, we do not observe significant differences in the number of senses over the entire set of verbs and nouns in the corpora. By focusing on the top 100 lists, we observe that while the nouns and verbs shared in common between TED and WMT explain the majority of the ambiguity with respect to polysemy, the non-overlapping lemmas demonstrate TED's higher ambiguity through the use of common verbs and nouns. By isolating the lemmas that are unique to each corpus' top 100 list, we see that TED's verbs and nouns exhibit 1.5 and 2 more senses respectively than those of WMT.

In order to measure the overall effects of polysemy on the corpora, we weight the noun and verb senses by their corpora frequencies. Figure 3b shows how the distributional frequency of noun and verb senses varies over TED and WMT. For verbs, we observe that TED exhibits fewer tokens with low ambiguity and a significant increase in tokens with over 11 word senses. The noun senses behave in a similar manner, though the differences are not as pronounced.

These results demonstrate that TED favors the use of common, expressive verbs. Examples are shown in Table 2. Piantadosi et al. (2012) explain this phenomena as a trade-off between the pressures of clarity and ease in communication. We find that this is the case when combining these observations with the perplexity measures in Section 5.2.

Lemma	# Senses	TED	WMT
tell	8	2159	362
learn	6	1102	336
hear	5	875	187
read	11	529	110

Table 2: Common polysemic verbs and their occurrence frequencies in TED and WMT.

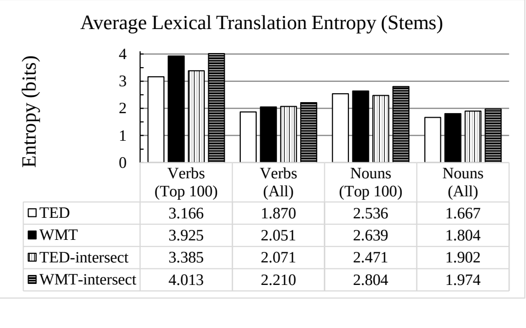
6.2 Lexical translation entropy

If the results in Section 6.1 suggest that TED talks are more ambiguous through the use of common verbs and nouns, does this transfer to the problem of SMT? To address this question, we analyze the lexical translation table provided by Moses and MGIZA through the word alignment process. We again compare TED and WMT both on the top 100 lists and the full sets of noun and verb lemmas. We train a word alignment model using MGIZA on the lemmatized corpora to build an English-German lexical translation table. In order to control the effects of alignment noise, we find the German lexical translations of each English lemma that cover the top 95% of the probability mass. Figure 4 compares TED and WMT in terms of lexical entropy.

Translating the top 100 verbs is much less ambiguous in the TED talk translation task (3.2 bits versus 3.9 bits). Most of the entropy is explained by the set of verbs TED and WMT share in common. WMT suffers from underspecification of these primarily common verbs. For example, the verb "bring", which occurs over 800 times in both corpora, exhibits an entropy of 4.04 bits and 170 translation options in TED, as opposed to 4.39 bits and 210 translation options for WMT. In terms of translation perplexity, the translation difficulty is as hard as deciding between 16 equally likely translations in TED, versus 21 in WMT. As a word with 11 senses in WordNet, this implies that fewer senses are actually being considered during translation in TED. A similar behavior can be observed for the common nouns. These results indicate that while TED has potentially higher English noun and verb polysemy, the common nouns and verbs are used more regularly than in WMT.

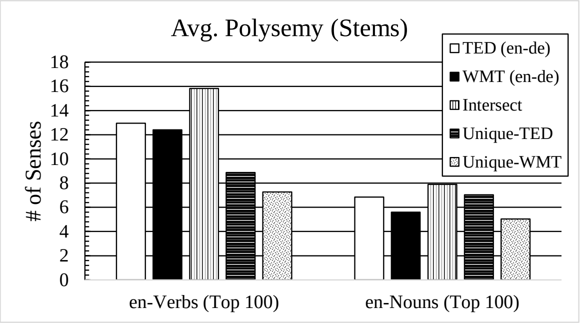
6.3 Pronominal anaphora

Hardmeier and Federico (2010) demonstrate that differences in the pronominal systems of a source

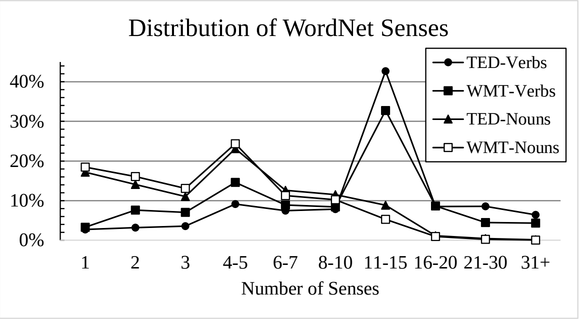


	Verbs (Top 100)	Verbs (All)	Nouns (Top 100)	Nouns (All)
□TED	3.166	1.870	2.536	1.667
■WMT	3.925	2.051	2.639	1.804
▥TED-intersect	3.385	2.071	2.471	1.902
▤WMT-intersect	4.013	2.210	2.804	1.974

Figure 4: Average lexical translation entropy on English noun and verb stems, computed from the top 95% threshold in the lexical translation table generated by MGIZA.



(a) Average number of senses per verb/noun for the 100 most frequent words in each corpus, as well as the types shared in common (Intersect), and those unique to the respective corpus (Unique-TED and Unique-WMT).



(b) Distribution of WordNet senses for all nouns and verbs in TED and WMT, weighted by observation frequency. Frequencies are bucketed to highlight differences between the corpora.

Figure 3: Polysemy statistics on TED and WMT corpora. Statistics are computed on the 100 most frequent verb/noun stems from each corpus as well as the full list of verbs/nouns found in WordNet.

and target language often results in the mistranslation of pronouns. For example, German has four personal pronoun cases, while English only has two. In cases of high ambiguity, it is up to models that depend on local context, such as n-gram LMs to determine the correct pronoun to use in the translation. If the local features of the sentence cannot resolve the ambiguity, the output pronoun is up to chance. We highlight two additional problems outlined by Hardmeier (2012): the difficulty for anaphora resolution systems to resolve pronouns (e.g. expletive pronouns), and translation divergences, such as when a pronoun is replaced with its referent in the translation.

Using the POS tags assigned by TreeTagger, we identify the English and German pronouns for TED and WMT and report statistics in Table 3. TED contains three times as many pronouns than WMT. While WMT contains few first and second person anaphoric mentions, TED consists of talks in which the speaker often refers to himself and to the audience. In particular, TED and WMT share seven pronominal translations for the English pronoun “you”, based on the context of the sentence. At times, “you” may be translated as an indefinite pronoun (“man”, “jemand”, “eine”), or can be replaced with a different grammatical person (“wir”, “sie”). TED contains additional ambiguity which may be attributed to word alignment errors, resulting in high translation entropy (1.53 bits). Likewise the indefinite and ambiguous pronoun “it” occurs twice as often in TED.

Person	Pronouns	TED	WMT	Diff	Rel Diff
1st	10	3.85%	0.48%	3.37%	699.2%
2nd	4	1.68%	0.06%	1.63%	2776.5%
3rd	24	4.06%	2.56%	1.50%	58.6%
Total	38	9.59%	3.10%	6.49%	209.5%

Table 3: Percent of English pronoun tokens in the 2 million word TED and WMT samples. Pronouns are grouped by grammatical person.

Field	TED	WMT	Diff	Rel Diff
Idioms/1K	1.541	2.122	-0.581	-27%
Avg. Length	2.896	2.695	0.201	7.46%
Types	494	556	-62	-11%
Singletons	289	271	18	7%

Table 4: The average rate of idioms per 1,000 words, idiom length, and the number of idiom and singleton types in each corpus sample.

6.4 Idiomatic expressions

Low frequency idiomatic expressions pose challenges for SMT systems. We crawled a list of English idioms generated by an online user community[3]. We manually scanned and pruned a handful of submitted entries that were likely to suggest more false positives than actual idiomatic expressions. In total, we collected 3,720 distinct idiomatic expressions. We perform a greedy idiom search on the surface representation of each corpus, favoring long idioms and ensuring that idioms did not overlap one another. Some statistics are reported in Table 4.

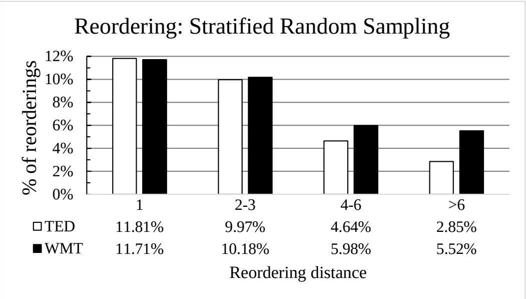
TED and WMT share 237 idioms in common, such as “at the end of the day”, “in the face of”, and “on the table”. These signify expressions that cross genres and are likely to be easily represented with statistical models. Some TED-specific expressions include “beeline for”, “bells and whistles”, “up the wall”, and “warm and fuzzy” – expressions that may be difficult to translate in MT systems trained on news genres. While TED uses fewer idioms overall, nearly 60% of the idiom types appear only once, compared to nearly 50% in WMT.

[3] http://www.usingenglish.com/reference/idioms/

7 Word reordering

One of the most notorious problems in phrase-based statistical machine translation is word reordering (Birch et al., 2009). Expressing the reordering problem as a task of searching through a set of word permutations for a given source sentence **f**, we arrange each source word f_i according to the mean of the target positions $\bar{a}_i$ aligned to it, as suggested by Bisazza and Federico (2013). Unaligned words are assigned the mean of their neighboring words' alignment positions. We then compute a word-after-word distortion length histogram between adjacent source words in their projection to the target language (Brown et al., 1990). To eliminate the effects of sentence length, we randomly sample 100 sentences with replacement for each observable sentence length in each corpus. A histogram is computed for each sentence length, whose results are averaged together.

Figure 5 compares the reordering behaviors of TED and WMT after stratified random sampling. Word permutations are computed from the symmetrized word alignments on English and German stems, using the grow-diag-final-and heuristic in Moses. To visualize the results better, we consider the absolute value of the relative distortion positions. In the figure, Bucket #1 corresponds to discontiguous reordering jumps one position forward (i.e. e_i -- e_{i+1}) or backward (i.e. e_{i+1} e_i), and so on. For example, "we could communicate" is translated once as "wir kommunizieren können" and yields reordering jumps of (+1,-1), which are both binned into Bucket #1. For English-German, monotonic reorderings account for 70.73% and 66.63% for TED and WMT, respectively. This 4% absolute increase in monotonic reorderings is accounted for by the reduction in long distance reorderings of four positions or more.



	1	2-3	4-6	>6
□TED	11.81%	9.97%	4.64%	2.85%
■WMT	11.71%	10.18%	5.98%	5.52%

Figure 5: Discontiguous word reordering percentage by reordering distance for English-German. Statistics are computed on reordering buckets of ± 1, $\pm[2,3]$, $\pm[4,6]$, and $\pm[7,\infty)$.

8 Machine Translation performance

Thus far, we have identified several linguistic factors that distinguish the TED translation task from that of WMT News Commentaries. We continue our analysis with a head-to-head comparison of MT performance. Since we cannot directly compare BLEU scores from the two official evaluation tasks, we create a small scale baseline evaluation that fixes the corpora sizes. Using the same two million word samples, we train separate SMT systems on TED and WMT, and tune two held-out samples of 100,000 words. We average the results of three MERT runs to reduce random effects. Each phrase-based SMT system is trained with the default training parameters of Moses (Koehn et al., 2007). We construct separate 4-gram LMs on the German side of the training data with IRSTLM, using a similar configuration as in Section 5.2. To evaluate, we control the effects of sentence length by focusing on sentences containing between 10 and 20 words (after tokenization). For each unique sentence length, we sample 200 sentences with replacement from 300,000 word segments of the TED and WMT corpora. We evaluate using the Translation Edit Rate (TER) metric (Snover et al., 2006). Results are reported in Figure 6 for SMT systems trained with 500K, 1M, and 2M words.

Due to the limited amount of TED data, we cannot measure the effects of additional training data on translation quality, but we attempt to extrapolate the learning curve by looking at smaller training sets. While we cannot explicitly say that TED translation yields higher translation quality than that of WMT, we do observe a growth in the absolute TER difference from 6.4% to 6.8% with 500K words and 2M words, respectively. Likewise, TED has fewer phrase table entries (3.5M vs. 3.7M) and LM entries (1.68M vs. 1.91M 4-grams) than WMT. These results suggest that the characteristics of TED allow better modeling of the translation task with less training data.

9 Discussion

Both TED and WMT News Commentary are good sandboxes for evaluating specific aspects of MT. Our experimental results identify several distinct linguistic phenomena that distinguish each genre's usefulness on specific areas of MT research.

TED talks enjoy performance advantages due to a SMT system's ability to translate their content reasonably well with a surprisingly small amount of training data. While TED has lower lexical ambiguity than WMT in terms of translation en-

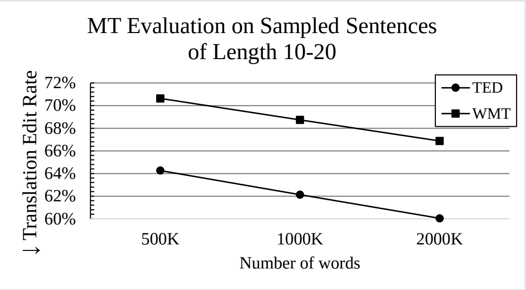


Figure 6: Phrase-based MT results for sampled sentences of length 10-20 in TED and WMT. SMT systems are trained with 500K, 1M, and 2M words.

tropy, it uses significantly more common and thus more ambiguous expressions. Because of this, it is a good candidate for evaluating semantically-informed translation models. The key issue for TED talks is the problem of pronominal anaphora. With over three times as many pronouns than WMT and twice as many third person mentions, the ability for MT systems to handle context is crucial. This makes it is an excellent task for investigating the translation of anaphoric expressions through discourse-aware MT, while at the same time managing the complexity of the system.

As WMT consists of longer sentences with more frequent cases of long distance reordering, it is a better task for measuring differences between hierarchical and linear phrase-based SMT. Additionally, with a lower German-English sentence length ratio, noun and verb compound detection may be a larger issue in WMT. WMT also suffers from higher perplexity scores than TED, suggesting that it can be a good benchmark for evaluating language modeling strategies with large amounts of readily-available in-domain data. Both TED and WMT are good candidates for research on handling idiomatic expressions during translation.

Some linguistic features do not correspond well with the problem of translation difficulty. As shown with our comparison of WordNet polysemy and lexical translation entropy, the challenge of disambiguating between a high number of noun and verb senses lessens during the word alignment process. This could be one of the reasons why previous work on word sense disambiguation in MT has yet to achieve significant improvements in automatic evaluations (Carpuat and Wu, 2007).

It should also be mentioned that while TED appears to be a simpler MT task overall, we have not addressed the larger problem of TED talk translation: the integration with automatic speech recognition. The linguistic features of TED make it a perfect candidate for speech translation, allowing researchers to focus on problems of translating content that may have been corrupted by speech recognition errors.

10 Conclusion

We have shown that the TED spoken language corpus and WMT News Commentary machine translation corpora exhibit differences in several linguistic features that each warrant dedicated research in machine translation. By sampling two million words from TED and WMT, we compared the two corpora on a number of linguistic aspects, including word statistics, such as sentence length and language model perplexity, lexical ambiguity, pronominal anaphora, idiomatic expressions, and word reordering. We observe that while TED consists of shorter sentences with less reordering behavior and stronger predictability through language model perplexity and lexical translation entropy, it has increased occurrences of pronouns that may refer to antecedents in the transcript and a high amount of polysemy through common verbs and nouns. In a small MT experiment, we evaluated a subset of sentence lengths in TED and WMT with MT systems trained on a comparable amount of data and show that TED can be modeled more compactly and accurately.

Finally, we have outlined linguistic features that distinguish the two corpora and propose suggestions to the MT community to focus their attention on TED or WMT, depending on their research goals. While both tasks are interesting for MT research, characteristics of spoken versus written texts provide different challenges to overcome.

Acknowledgments

This work was supported by the EU-BRIDGE project (IST-287658), which is funded by the European Commission under the Seventh Framework Programme for Research and Technological Development. The authors would like to thank Christian Girardi for his assistance with extracting idioms from the web, as well as Arianna Bisazza, Mauro Cettolo, and Carlo Strapparava for other scripts and valuable discussions.

References

D. Biber. 1988. *Variation Across Speech and Writing*. Cambridge University Press, Cambridge.

A. Birch, P. Blunsom, and M. Osborne. 2009. A quantitative analysis of reordering phenomena. In *StatMT '09: Proceedings of the Fourth Workshop on Statistical Machine Translation*, pages 197–205, Morristown, NJ, USA. Association for Computational Linguistics.

A. Bisazza and M. Federico. 2013. Dynamically Shaping the Reordering Search Space of Phrase-Based Statistical Machine Translation. *Transactions of the Association for Computational Linguistics*, 1:327–340.

O. Bojar, C. Buck, C. Callison-Burch, C. Federmann, B. Haddow, P. Koehn, C. Monz, M. Post, R. Soricut, and L. Specia. 2013. Findings of the 2013 Workshop on Statistical Machine Translation. In *Proceedings of the Eighth Workshop on Statistical Machine Translation*, pages 1–44, Sofia, Bulgaria, August. Association for Computational Linguistics.

P. F. Brown, J. Cocke, S. A. Della Pietra, V. J. Della Pietra, F. Jelinek, J. Lafferty, R. L. Mercer, and P. Rossin. 1990. A statistical approach to machine translation. *Computational Linguistics*, 16(2):79–85.

M. Carpuat and D. Wu. 2007. Improving statistical machine translation using word sense disambiguation. In *Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 61–72.

M. Carpuat. 2009. One translation per discourse. In *Proceedings of the Workshop on Semantic Evaluations: Recent Achievements and Future Directions*, DEW '09, pages 19–27, Stroudsburg, PA, USA. Association for Computational Linguistics.

M. Cettolo, J. Niehues, S. Stüker, L. Bentivogli, and M. Federico. 2013. Report on the 10th IWSLT Evaluation Campaign. In *Proc. of the International Workshop on Spoken Language Translation*, December.

M. Federico, N. Bertoldi, and M. Cettolo. 2008. IRSTLM: an Open Source Toolkit for Handling Large Scale Language Models. In *Proceedings of Interspeech*, pages 1618–1621, Brisbane, Australia.

C. Fellbaum, editor. 1998. *WordNet: an electronical Lexical Database*. MIT Press, Cambridge, MA.

G. Foster, P. Isabelle, and R. Kuhn. 2010. Translating structured documents. In *Proceedings of the Ninth Conference of the Association for Machine Translation in the Americas*.

A. C. Graesser, M. Singer, and T. Trabasso. 1994. Constructing inferences during narrative text comprehension. *Psychol Rev*, 101(3):371–395, July.

A. C. Graesser, D. S. McNamara, M. M. Louwerse, and Z. Cai. 2004. Coh-Metrix: Analysis of text on cohesion and language. *Behavior Research Methods, Instruments, & Computers*, 36(2):193–202, May.

C. Hardmeier and M. Federico. 2010. Modelling Pronominal Anaphora in Statistical Machine Translation. In *Proceedings of the Seventh International Workshop on Spoken Language Translation (IWSLT)*, pages 283–289.

C. Hardmeier. 2012. Discourse in Statistical Machine Translation. *Discours*, (11), December.

P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E. Herbst. 2007. Moses: Open Source Toolkit for Statistical Machine Translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic.

M. M. Louwerse, P. M. McCarthy, D. S. McNamara, and A. C. Graesser. 2004. Variation in language and cohesion across written and spoken registers. In *Proceedings of the 26th Annual Meeting of the Cognitive Science Society*, pages 843–848.

M. Marcus, B. Santorini, and M. Marcinkiewicz. 1993. Building a large annotated corpus of english: the penn treebank. *Computational Linguistics*, 19:313–330.

J. Mitchell, M. Lapata, V. Demberg, and F. Keller. 2010. Syntactic and semantic factors in processing difficulty: An integrated measure. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 196–206, Stroudsburg, PA, USA.

N. Newbold and L. Gillam. 2010. The linguistics of readability: The next step for word processing. In *Proceedings of the NAACL HLT 2010 Workshop on Computational Linguistics and Writing: Writing Processes and Authoring Aids*, pages 65–72, Stroudsburg, PA, USA.

S. T. Piantadosi, H. Tily, and E. Gibson. 2012. The communicative function of ambiguity in language. *Cognition*, 122(3):280 – 291.

A. Schiller, S. Teufel, C. Stöckert, and C. Thielen, 1995. *Vorläufige Guidelines für das Tagging deutscher Textcorpora mit STTS*. Universität Stuttgart, Institut für maschinelle Sprachverarbeitung.

H. Schmid. 1994. Probabilistic part-of-speech tagging using decision trees. In *Proceedings of International Conference on New Methods in Language Processing*.

M. Snover, B. Dorr, R. Schwartz, L. Micciulla, and J. Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *5th Conference of the Association for Machine Translation in the Americas (AMTA)*, Boston, Massachusetts, August.

L. Specia, N. Hajlaoui, C. Hallett, and W. Aziz. 2011. Predicting Machine Translation Adequacy. In *Machine Translation Summit XIII*, pages 73–80.

Oral Session 5
Research Papers

Does post-editing increase usability? A study with Brazilian Portuguese as Target Language

Sheila Castilho
CNGL/SALIS
Dublin City University
Ireland
castils3@mail.dcu.ie

Sharon O'Brien
CNGL/SALIS
Dublin City University
Ireland
sharon.obrien@dcu.ie

Fabio Alves
Federal University of Minas Gerais
(UFMG)
Brazil
fabio-alves@ufmg.br

Morgan O'Brien
McAfee
Mahon, Cork
Ireland
morgan_o'brien@mcafee.com

Abstract

It is often assumed that raw MT output requires post-editing if it is to be used for more than gisting purposes. However, we know little about how end users engage with raw machine translated text or post-edited text, or how usable this text is, in particular if users have to follow instructions and act on them. The research project described here measures the usability of raw machine translated text for Brazilian Portuguese as a target language and compares that with a post-edited version of the text. Two groups of 9 users each used either the raw MT or the post-edited version and carried out tasks using a PC-based security product. Usability was measured using an eye tracker and cognitive, temporal and pragmatic measures of usability, and satisfaction was measured using a post-task questionnaire. Results indicate that post-editing significantly increases the usability of machine translated text.

1 Introduction

This paper discusses the measurement of usability for raw machine translated output and post-edited output for instructional text relating to a commercial PC security product machine translated from English into Brazilian Portuguese.

Authentic English source text relating to the software product (anonymised for confidentiality reasons) was identified and machine translated into Brazilian Portuguese using the freely available MT engine, Microsoft Bing.

Eighteen users were recruited to read the instructions and carry out tasks by creating files and folders, changing settings within the product etc. The participants were divided equally into two groups; one group used the raw machine translated instructions and the other used the post-edited instructions. The usability of both sets of instructions was investigated using screen recording, eye tracking and a post-task questionnaire. The main objective of this project was to investigate the extent to which human post-editing of machine translation impacted on the usability of instructional content.[1]

The paper is structured as follows: Section 2 discussed related research, Section 3 explains the methods used, Section 4 provides results and Section 5 the conclusions.

2 Related Work

The task and process of post-editing has received significant attention in the past few years (e.g. Guerberof (2014), De Almeida and O'Brien (2010), Depraetere (2010), Plitt and Masselot (2010), Sousa et al. (2011), Koponen (2012), O'Brien et al. (2012), O'Brien et al. (2013), Specia (2011)). While MT technology has made sig-

[1] This research is supported by the International Strategic Cooperation Award through Science Foundation Ireland and Dublin City University.

nificant strides in the last decade, it is accepted that post-editing is needed in cases where the content is required for more than gisting purposes. Empirical research has demonstrated that post-editing can lead to higher productivity, without having negative effects on quality (e.g. Guerberof, forthcoming), though it might have an impact on perceptions of stylistic quality (Fiederer and O'Brien 2009). Yet little empirical research has focused on the value of post-editing or on its return on investment (ROI). It is generally assumed that post-editing is required to bring content to a publication-ready level, but we know very little about the impact that post-editing has on the usability and, by extension, acceptability of machine translated content.

Related work is at this stage still somewhat limited. Jones et al (2005) present a usability test where participants answer questions from a machine translated version of an Arabic language test. Their results suggest that MT may enable an ILR level 2 (limited working proficiency) but it is not suitable for level 3 (general professional proficiency).

Stymne et al (2012) use eye tracking as a complement to MT error analysis. They found that MT errors have longer gaze time and more fixations than correct passages of text and the average gaze time is dependent on error types, which could indicate that some error types require more cognitive effort than others.

In 2010, Doherty, O'Brien and Carl tested the use of eye tracking as a machine translation evaluation technique, concluding that eye tracking was a reliable method for evaluating the quality of machine translated output. Building on this, Doherty and O'Brien (2014) conducted a study to compare the usability of raw machine translated output for four target languages against the usability of the source content (English). The conclusion of that study was that, although the raw MT output scored lower for usability measurements when compared with the source language content, the raw MT output was deemed to be usable, especially for Spanish as a target language. The target language Japanese, unsurprisingly, scored lowest in terms of usability.

The study by Doherty and O'Brien (2014) used both questionnaires and eye-tracking measurements to record levels of usability. The current study builds on that, but is different in several respects: (1) the content translated differs; (2) the target language in this case is Brazilian Portuguese, which was not included in the 2014 study; (3) the MT system differs and, most importantly, (4) the current study compares the usability of raw MT output against post-edited content, not against the usability of the source language content, which was the case for the previous study.

3 Methods

In this section we discuss the methods deployed to measure usability and the experiment design.

3.1 Measuring Usability

We adopt the ISO/TR 16982 definition for usability: "the extent to which a product can be used by specified users to achieve specified goals with **effectiveness**, **efficiency**, and **satisfaction** in a specified content of use" (ISO 2002).

When this definition is divided into its component parts (in bold above), it allows us to measure different aspects of usability using a variety of methods.

Effectiveness is measured through goal completion, that is, how successful the users were at accomplishing tasks documented in the instructions measured by observing the user interactions as recorded by a Tobii T60XL eye tracker.

Efficiency is measured as the number of successful tasks completed (out of all possible tasks) when total task time is taken into account. A second measure of efficiency is cognitive effort, i.e. how much cognitive effort is evident when users are reading the instructions and trying to complete their tasks? Cognitive effort is measured using typical indicators recorded via the eye tracking apparatus, i.e. mean total fixation time, mean fixation duration, total fixation count, average visit duration and visit count. Such fixation data are well established as indicators of cognitive effort (Rayner 1998, Rayner and Sereno 1994, Radach et al. 2004). For example, the more fixations there are on a set of instructions, the more probable it is that the reader is having difficulties in processing the instructions.

Satisfaction is a measure of user satisfaction with the translated content and, by extension, the product itself. As satisfaction is a multi-faceted concept, we measure it using a questionnaire with a Likert scale ranging from 1 (Strongly Disagree) to 5 (Strongly Agree). In our questionnaire, "satisfaction" is addressed using a number of statements (see section 4.8).

3.2 Content

In collaboration with an industry partner, we selected a security software product that controlled for viruses, allowed for the setting of parental controls and so on and instructional content in English on how to configure features of this product. The total number of words in the source content amounted to 594. This content was machine translated into Brazilian Portuguese using Microsoft's Bing engine.[2] Brazilian Portuguese was selected for this study as it was part of a Brazil/Ireland research collaboration project. The raw machine translated output was post-edited by a native speaker of Brazilian Portuguese who has an undergraduate degree in linguistics and literature and a Master's degree in natural language processing and human language technology. The post-editor had also conducted research previously on post-editing. The guidelines adhered to during post-editing were those of TAUS for the level "fit-for-purpose" (TAUS: online). From a practical perspective, this meant that edits were carried out when terminology did not conform to the client-specific glossary and grammatical errors were fixed. No edits were implemented for purely stylistic reasons and the focus was on accuracy and comprehensibility.

To measure how much post-editing was performed we conducted an automatic evaluation comparing the post-edited version against the MT output. We observed an average HTER score of 0.20 which indicates that post-editing was of a light nature.

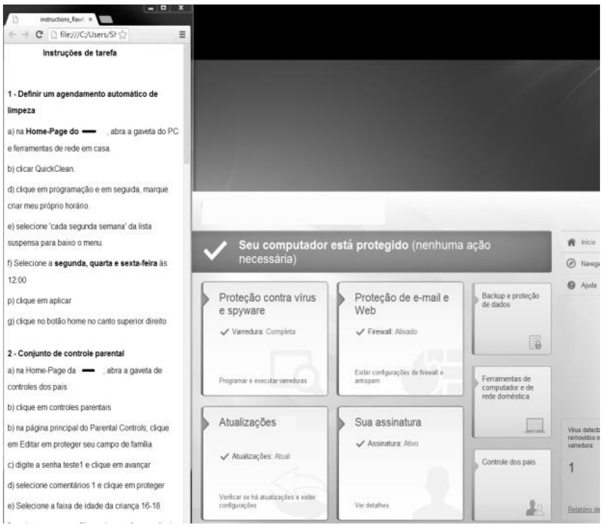
3.3 Participants

18 native speakers of Brazilian Portuguese where recruited from the student body of the Federal University of Minas Gerais, Belo Horizonte, Brazil.[3] It was ensured that participants had no previous experience of this particular security product so that previous knowledge could not be used to compensate for poor quality machine translation output (Moravcsik and Kintsch 1995, Kaakinen et al. 2003).

The participants were randomly assigned to one of two groups: Group 1 used the raw machine translated output and were asked to follow the instructions while Group 2 read and followed the post-edited instructions. Neither group knew that the texts they were reading had been translated. Both groups were given a warm up task where they were asked to read a text in Brazilian Portuguese for comprehension; the text came from Wikipedia and explained the concept of virus checking. Fixation data gathered during this reading exercise were used as a baseline measurement for 'reading for comprehension' in Brazilian Portuguese among participants. Two participants (one from each group) appeared to be outliers in terms of several of the fixation measurements and were removed from each group.

Participants were seated at the eye tracker and were informed that they would be presented with some instructions on the left-hand side of the screen and a software product on the right hand side in which they had to perform five tasks as per the instructions (see Figure 1 for layout – for confidentiality reasons, company-specific information has been removed).

The tasks involved setting up an automatic cleaning schedule, setting parental controls, creating a vault, shredding files and deleting a vault. Participants were instructed not to reposition any of the windows relating to the software product or the instructions, so as to facilitate eye-tracking analysis. Once they had completed their tasks they responded the questionnaire.



Instruções de tarefa

1 - Definir um agendamento automático de limpeza

a) na **Home-Page do** [redacted] , abra a gaveta do PC e ferramentas de rede em casa.

b) clicar QuickClean.

d) clique em programação e em seguida, marque criar meu próprio horário.

e) selecione 'cada segunda semana' da lista suspensa para baixo o menu.

f) Selecione a **segunda, quarta e sexta-feira** às 12:00

p) clique em aplicar

g) clique no botão home no canto superior direito

2 - Conjunto de controle parental

a) na Home-Page da [redacted] , abra a gaveta de controles dos pais

b) clique em controles parentais

b) na página principal do Parental Controls, clique em Editar em proteger seu campo de família

c) digite a senha teste1 e clique em avançar

d) selecione comentários 1 e clique em proteger

e) Selecione a faixa de idade da criança 16-18

Seu computador está protegido (nenhuma ação necessária)

Varredura: Completa

Firewall: Ativado

Atualizações: Atual

Assinatura: Ativo

instructions_Raw
file:///C:/Users/St
Proteção contra vírus e spyware
Programar e executar varreduras
Proteção de e-mail e Web
Exibir configurações de firewall e antispam
Backup e proteção de dados
Ferramentas de computador e de rede doméstica
Atualizações
Verificar se há atualizações e exibir configurações
Sua assinatura
Ver detalhes
Controle dos pais
Início
Navega
Ajuda
Vírus detecta removidos n varredura
1
Relatório de

Figure 1- Set up screenshot

[2] Our intention had been to use the company-specific MT engine trained using the Microsoft Translator Hub. However, at the time of the experiment, technical difficulties prevented this and the company suggested the use of the generic Bing engine as an alternative.

[3] Ethics approval was granted by the relevant university research ethics committee.

4 Results

We first present the results from the eye tracking data, which, as discussed above, we treat as measures of efficiency. For all results “Baseline” refers to the baseline reading task of the Wikipedia text, “Instructions” refers to the eye tracking data for the area of the screen in which the instructions were displayed (the AOI, or Area of Interest) and “Interface” refers to the area on the screen in which the product itself was displayed and where users had to carry out the tasks required. For the eye tracking data, “MT” refers to the raw MT instructions and PE refers to the post-edited version. We first present cognitive indicators of efficiency (fixation measures: 4.1-4.5), then goal completion as a measure of effectiveness (4.6), followed by goal completion as a factor of time (also a measure of efficiency – 4.7) and finally satisfaction measures (4.8).

4.1 Mean Total Time in Fixation

The mean total time in fixation is the time spent in fixations combined for each group within an AOI (in seconds).

Figure 2 shows the mean across both groups for the baseline, MT and PE texts. Data for the baseline text is much shorter, as would be expected, because this was just one short text that had to be read and there was no other task associated with it. The mean total fixation time is higher for the MT group for both the Instructions AOI and the Interface AOI.

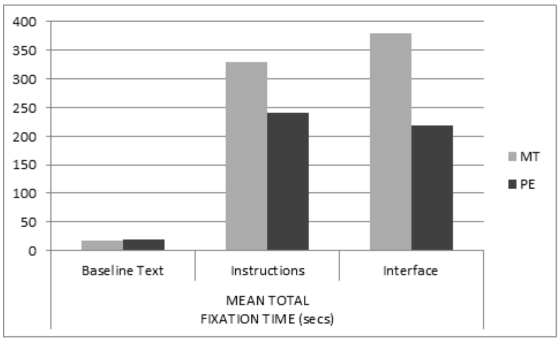


Figure 2- Mean Total Time in Fixation

An independent-samples t-test was conducted to compare both conditions. There was a significant difference in the scores for total fixation time for the Instructions AOI: *t-value* (14) = 2.83, *p-value* = .013 and for the Interface AOI: *t* (14) = 4.58, *p* = .001. There was no significant difference between groups for the Baseline (*p* = .65), which indicates that there was no difference in the baseline reading activity between the two groups. (All significance levels at p > 0.05.)

4.2 Mean Fixation Duration

Mean FD (in seconds) is the average length of fixations for all participants in both groups (Figure 3).

For both groups, the mean value is 0.33 for the baseline, again indicating that there was no difference across both groups for the baseline task. For the Instructions AOI, the mean fixation duration for the PE group is (0.45) and for the MT group (0.43). Both are greater than the baseline, suggesting that reading of the MT output (either in raw or post-edited form) required greater effort than reading the baseline text. Although the value for the MT text is slightly higher than that of the PE text (0.45 vs. 0.43), these are not statistically different. This is also the case for the Interface AOI.

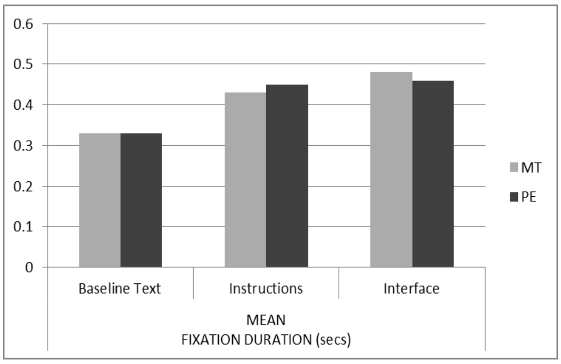


Figure 3- Mean Fixation Duration

4.3 Fixation Count

Fixation count (FC) is the total number of fixations within an AOI. The more there are, the higher the cognitive effort is deemed to be. As can be seen (Figure 4), the total FC is higher for the MT group for both the Instructions and Interface AOIs. Table 1 also shows the mean, median and standard deviations values for the Fixation Count measure. (Note: We do not report data for the baseline reading task here as comparisons of fixation count would be meaningless, given that the task and text differ substantially from the task and text used in the actual experiment. Comparisons for mean total fixation time (Fig. 2), on the other hand, are meaningful as they demonstrate that the groups did not differ radically in their baseline reading activity.)

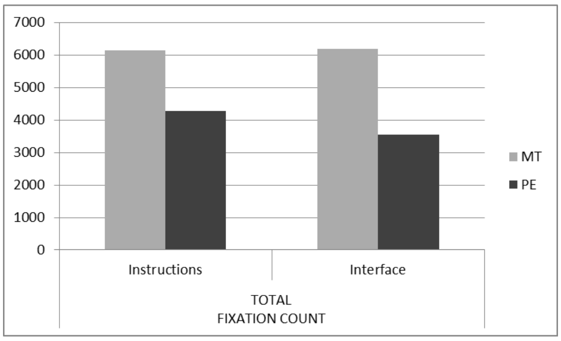
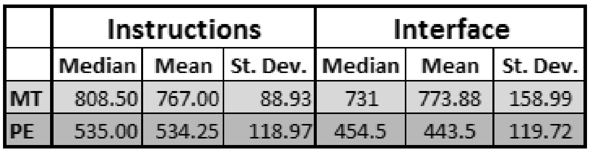


Figure 4- Total Fixation Count

A significant difference was found for Fixation Count on the AOI Instructions: *t* (14) = 4.43, *p* = .001 as well as for the Fixation Count on the AOI Interface, *t* (14) = 4.69, *p* <.001.

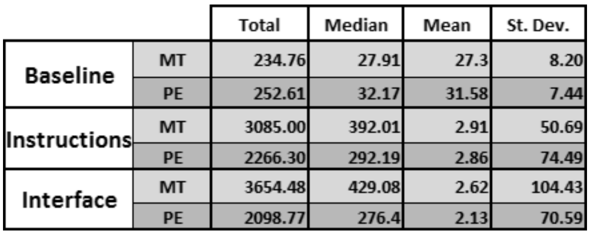


	Instructions			Interface		
	Median	Mean	St. Dev.	Median	Mean	St. Dev.
MT	808.50	767.00	88.93	731	773.88	158.99
PE	535.00	534.25	118.97	454.5	443.5	119.72

Table 1 – Fixation Count & St. Dev

4.4 Visit Duration

Visit duration (VD) is the total time (in seconds) spent looking at an AOI, starting with a fixation within the AOI and ending with a fixation outside this AOI, that is, saccades (or rapid eye movements between fixations) are also counted. Table 2 presents the values for the baseline, instructions and interface for both groups.



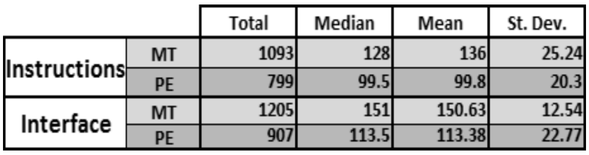
		Total	Median	Mean	St. Dev.
Baseline	MT	234.76	27.91	27.3	8.20
	PE	252.61	32.17	31.58	7.44
Instructions	MT	3085.00	392.01	2.91	50.69
	PE	2266.30	292.19	2.86	74.49
Interface	MT	3654.48	429.08	2.62	104.43
	PE	2098.77	276.4	2.13	70.59

Table 2 - Visit Duration (secs)

As Table 2 demonstrates, the mean VD is higher for the machine translation group for both the Instructions and Interface. A *t*-test found a significant difference between both conditions, where *t*(14) = 3.212, *p* = .006 for the AOI Instructions and *t*(14) = 4.363, *p* = .001 for the AOI Interface. For the baseline, *t*(14) = -.578, *p* = .578, again suggesting that there is no statistically significant difference between the two groups in the baseline task and so the effects we see between the two conditions MT and PE are likely to have been produced by the texts themselves and not by variances in the groups.

4.5 Visit Count

Visit Count is the number of visits (using eye movements as evidence) to an AOI. Table 3 shows the number for VC for both MT and PE groups:



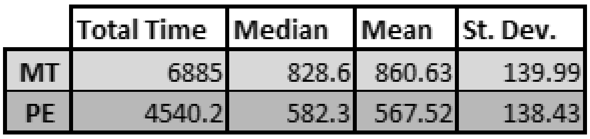
		Total	Median	Mean	St. Dev.
Instructions	MT	1093	128	136	25.24
	PE	799	99.5	99.8	20.3
Interface	MT	1205	151	150.63	12.54
	PE	907	113.5	113.38	22.77

Table 3 - Visit Count

Note that the baseline is not shown here as the number of visits in a static text presented for reading would always be 1. The total VC is higher for the MT group for both AOIs. A *t*-test found a significant difference between both conditions, where t(14)= 3.209, *p* = .006 for the AOI Instructions and *t*(14)= 4.052, *p* = .001 for the AOI Interface.

4.6 Goal Completion - Effectiveness

All participants in the PE group were able to complete all the tasks, with the exception that one participant in the MT group skipped task 1 (Set an Automatic Cleaning Schedule). This demonstrates that, regardless of the type of instructions, participants were still able to complete their tasks. At the same time, it is worth pointing out some confusion among those who read the raw MT instructions: For Task 2 (Set Parental Controls) one of the options to be blocked by the participants had a different translation from the interface. As a result, some participants were not able to select that option and skipped it, but the task as a whole was completed. Also, Tasks 3 and 5 for the MT group resulted in participants erasing and moving incorrect files but, in the end, the task of creating and deleting the vault was completed. Table 4 gives the total task times for both groups.



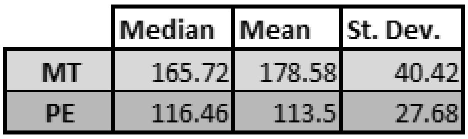
	Total Time	Median	Mean	St. Dev.
MT	6885	828.6	860.63	139.99
PE	4540.2	582.3	567.52	138.43

Table 4 – Total Task Time (secs)

An independent-samples t-test was conducted to compare both conditions. There was a significant difference between the MT and PE groups; *t-value* (14) = 4.21, *p-value* = .001.

4.7 Efficiency

Efficiency is also measured as the number of successful tasks completed divided by the total task time (Table 5). The PE group were found to be more efficient (t (14) = 3.75, p = .002).



	Median	Mean	St. Dev.
MT	165.72	178.58	40.42
PE	116.46	113.5	27.68

Table 5 – Efficiency Scores (secs)

4.8 Satisfaction

As mentioned in Section 3, the participants responded to a post-task questionnaire that measured their level of satisfaction with the instructions through a range of questions. None of the participants knew that the instructions had been machine translated.

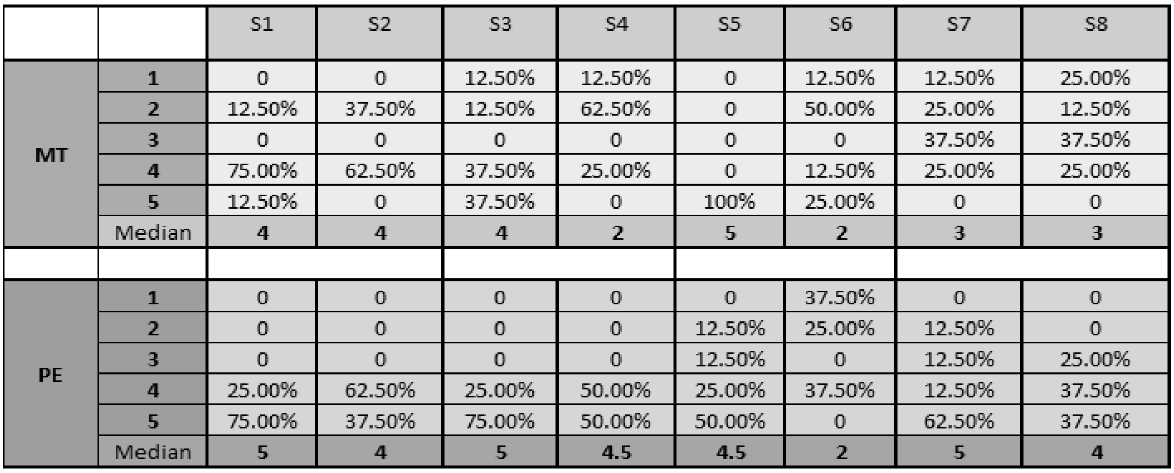
As a reminder, the statements they had to respond to were as follows:

1. The instructions were usable.
2. The instructions were comprehensible.
3. The instructions allowed me to complete all of the necessary tasks.
4. I was satisfied with the instructions provided.[4]
5. The instructions could be improved upon.
6. I would be able to use the software again in the future without re-reading the instructions.
7. I would recommend the software to a friend or a colleague.
8. I would consider buying this product after participating in this experiment.

Table 6 presents the results for each statement and each group. For all statements, except number 5, the higher score (5) indicates higher satisfaction (the opposite is true for statement 5). As can be seen, levels of satisfaction are generally higher for the post-edited instructions. Exceptions include statements 2, 6 and 5. In the case of 5, the lower score means higher satisfaction for the post-edited text. The considerable difference in scores for statements 7 and 8 are worth noting due to the potential commercial implications. Those who read the post-edited text would seem more inclined to recommend or purchase the product.

5 Conclusions and Future Work

We set about measuring and comparing the usability of instructions for a software product that had been machine translated and machine translated and lightly post-edited.



		S1	S2	S3	S4	S5	S6	S7	S8
MT	1	0	0	12.50%	12.50%	0	12.50%	12.50%	25.00%
	2	12.50%	37.50%	12.50%	62.50%	0	50.00%	25.00%	12.50%
	3	0	0	0	0	0	0	37.50%	37.50%
	4	75.00%	62.50%	37.50%	25.00%	0	12.50%	25.00%	25.00%
	5	12.50%	0	37.50%	0	100%	25.00%	0	0
	Median	4	4	4	2	5	2	3	3
PE	1	0	0	0	0	0	37.50%	0	0
	2	0	0	0	0	12.50%	25.00%	12.50%	0
	3	0	0	0	0	12.50%	0	12.50%	25.00%
	4	25.00%	62.50%	25.00%	50.00%	25.00%	37.50%	12.50%	37.50%
	5	75.00%	37.50%	75.00%	50.00%	50.00%	0	62.50%	37.50%
	Median	5	4	5	4.5	4.5	2	5	4

Table 6 – Post-Task Questionnaire

[4] We made sure the participants understood that by ‘instructions’ we meant the written task instructions provided to perform the tasks, not the verbal instructions given by the researcher on how the experiment would be carried out.

Our objective was to see whether the post-edited version was more usable than the raw MT output. The natural hypothesis is to assume that post-editing improves the quality and usability of a text, but this is usually measured using quality evaluation and not via end user eye tracking-based measurements. The empirical investigation we have carried out here is a validation of this hypothesis.Using the ISO/TR 16902 definition of usability, we undertook a suite of measurements to assess different parts of this definition. Measures of effectiveness included the cognitive measurements of mean total fixation time, mean fixation duration, fixation count, visit duration, and visit count. For all of these measures except mean fixation duration a statistically significant difference was found between the MT and PE groups implying that those who read the PE instructions were more effective and that therefore those instructions had a higher level of usability.

The measurement of goal achievement demonstrated that regardless of the type of instructions, both groups were successful in achieving their goals. We put this down to the use of human intelligence and experience in making sense of content that is not optimal. Moreover, a higher level of confusion was evident among the MT group, as discussed above.

Additional measures of effectiveness and efficiency also demonstrated that the PE instructions were more usable. Finally, the responses to a post-task questionnaire on satisfaction indicated a higher level of satisfaction among those who used the post-edited instructions. Noteworthy in particular are the responses regarding recommendation to a friend or the purchase of the product; for both statements those who read the post-edited instructions were more likely to do so, which has important implications for commercial users of MT.

We have shown that post-editing – even to the level of ‘fit-for-purpose’ – adds value to machine translated content because it increases usability and satisfaction levels. While this is perhaps an unsurprising result, the important aspect of this study is the number of measures of usability and the inclusion of end users actually performing tasks with the instructions and a software product. This lends a higher level of credibility to the claim of increased usability.

Obviously the sample size is small and we have included only one language pair so future work could build on the number of participants and language pairs. Another focus in the future will be comparisons between human translation and raw and post-edited MT as well as a focus on different kinds of content.

References

De Almeida, Gisele and Sharon O’Brien. 2010. Analysing Post-Editing Performance: Correlations with Years of Translation Experience. *Proceedings of the 14th Annual Conference of the EAMT*. St. Raphael, May 27-28.

Depraetere, Ilse. 2010. What Counts as Useful Advice in a University Postediting Training Context? Report on a case study. *Proceedings of the 14th Annual EAMT Conference*. St. Raphael, May 27-28.

Doherty, Stephen and Sharon O’Brien. 2013. Assessing the Usability of Raw Machine Translation Output: A User-Centered Study using Eye Tracking. *International Journal of Human-Computer Interaction,* 30: 40-51.

Doherty, Stephen, Sharon O’Brien and Michael Carl. 2010. Eye Tracking as an MT Evaluation Technique. *Machine Translation,* 24(1): 1-13.

Fiederer, Rebecca, and Sharon O’Brien. 2009. Quality and Machine Translation: A Realistic Objective? *Journal of Specialised Translation [online]*.

Guerberof, Ana. Forthcoming. The Role of Professional Experience in Post-Editing from a Quality and Productivity Perspective. In O’Brien, Sharon, Laura, Winther-Balling, Michael Carl, Michel Simard and Lucia Specia (eds) *Post-Editing of Machine Translation: Processes and Applications.* Cambridge Scholars Publishing, 51-76.

International Organization for Standardization. 2002. *ISO/TR 16982: Ergonomics of human-system interaction – Usability methods supporting human centered design*. Available from: http://www.iso.org/iso/catalogue_detail?csnumber =31176.

Jones, Douglas, Wade Shen, Neil Granoien, Martha Herzog and Clifford Weinstein. 2005. Measuring Translation Quality by Testing English Speakers with a New Defense Language Proficiency Test for Arabic. *Proceeding of the International Conference on Intelligence Analysis*. McLean, VA.

Kaakinen, Johann, Jukka Hyönä and Janice Keenan. 2003. How prior knowledge, WMC, and relevance of information affect eye fixations in expository text. *Journal of Experimental Psychology: Learning, Memory and Cognition,* 29(3): 447-457.

Koponen, Maarit. 2012. Comparing Human Perceptions of Post-editing Effort with Post-editing Operations. *Proceedings of the 7th Workshop on Statistical Machine Translation*. Association for Computational Linguistics, Montreal, June 7-8.

Moravcsik, Julia and Walter Kintsch. 1995. Writing quality, reading skills, and domain knowledge as factors in text comprehension. In John Henderson, M. Singer, and F. Ferreira (eds.) *Reading and Language Processing*. New York, London: Psychology Press, 232-246.

O’Brien, Sharon, Michel Simard and Lucia Specia (Eds.) 2012. Workshop on Post-editing Technology and Practice (WPTP 2012). *Conference of the Association for Machine Translation in the Americas (AMTA 2012).* San Diego, October 28.

O’Brien, Sharon, Michel Simard and Lucia Specia (Eds.) 2013. Workshop on Post-editing Technology and Practice (WPTP 2013). *Machine Translation Summit XIV*. Nice, September 2-6.

Plitt, Mirko, and Françoise Masselot. 2010. A Productivity Test of Statistical Machine Translation Post-editing in a Typical Localisation Context. *The Prague Bulletin of Mathematical Linguistics*. Prague: 7-16.

Radach, Ralph, Alan Kennedy and Keith Rayner. 2004. *Eye Movements and Information Processing during Reading*. Hove: Psychology Press.

Rayner, Keith. 1998. Eye Movements in Reading and Information Processing: 20 Years of Research. *Psychological Bulletin*, 124:372-422.

Rayner, Keith and Sarah Sereno. 1994. Eye Movements in Reading: Psycholinguistic Studies. In Gernsbacher M.A. (ed.), *Handbook of Psycholinguistics*. New York: Academic Press, 57-81.

Sousa, Sheila C.M., Wilker Aziz, and Lucia Specia. 2011. Assessing the post-editing effort for automatic and semiautomatic translations of DVD subtitles. *Proceedings of the Recent Advances in Natural Language Processing Conference*. Hissar, Bulgaria.

Specia, Lucia. 2011. Exploiting Objective Annotations for Measuring Translation Post-editing Effort. *Proceedings of the 15th Annual EAMT Conference*. Leuven, May 30-31.

Stymne, Sara, Henrik Danielsson, Sofia Bremin, Hongzhan Hu, Johanna Karlsson, Anna Prytz Lillkull and Martin Wester. 2012. Eye Tracking as a Tool for Machine Translation Error Analysis. *Proceedings of the Language Resources and Evaluation Conference*. 1121-1126. Istanbul. May 21-27.

Translation Automation User Society (TAUS). Online. *Machine Translation Post-Editing Guidelines*. Available at: https://evaluation.taus.net/resources/guidelines/post-editing/machine-translation-post-editing-guidelines

Relations between different types of post-editing operations, cognitive effort and temporal effort

Maja Popović, Arle Lommel, Aljoscha Burchardt,
Eleftherios Avramidis, Hans Uszkoreit
DFKI – Berlin, Germany
name.surname@dfki.de

Abstract

Despite the growing interest in and use of machine translation post-edited outputs, there is little research work exploring different types of post-editing operations, i.e. types of translation errors corrected by post-editing. This work investigates five types of post-edit operations and their relation with cognitive post-editing effort (quality level) and post-editing time. Our results show that for French-to-English and English-to-Spanish translation outputs, lexical and word order edit operations require most cognitive effort, lexical edits require most time, whereas removing additions has a low impact both on quality and on time. It is also shown that the sentence length is an important factor for the post-editing time.

1 Introduction and related work

In machine translation research, ever-increasing amounts of post-edited translation outputs are being collected. These have been used primarily for automatic estimation of translation quality. However, they enable a large number of applications, such as analysis of different aspects of post-editing effort. (Krings, 2001) defines three aspects: temporal, referring to time spent on post-editing, cognitive, referring to identifying the errors and the necessary steps for correction, and technical, referring to edit operations performed in order to produce the post-edited version. These aspects of effort are not necessary equal in various situations.

Since the temporal aspect is important for the practice, post-editing time is widely used for measuring post-editing effort (Krings, 2001; Tatsumi, 2009; Tatsumi et Roturier, 2010; Specia, 2011). Human quality scores based on the needed amount of post-editing are involved as assessment of the cognitive effort in (Specia et al., 2010; Specia, 2011). Using edit distance between the original and the post-edited translation for assessment of the technical effort is reported in (Tatsumi, 2009; Tatsumi et Roturier, 2010; Temnikova, 2010; Specia, 2011; Blain et al., 2011).

More details about the technical effort can be obtained by analysing particular edit operations. (Blain et al., 2011) defined these operations on a linguistic level as post-editing actions and performed comparison between statistical and rule-based systems. (Temnikova, 2010) proposed the analysis of edit operations for controlled language in order to explore cognitive effort for different error types – post-editors assigned one of ten error types to each edit operation which were then ranked by difficulty. In (Koponen, 2012) post-edit operations are analysed in sentences with discrepancy between the assigned quality score and the number of performed post-edits. In one of the experiments described in (Wisniewski et al., 2013) an automatic analysis of post-edits based on Levenshtein distance is carried out considering only the basic level of substitutions, deletions, insertions and TER shifts. These edit operations are analysed on the lexical level in order to determine the most frequent affected words. General user preferences regarding different types of machine translation errors are explored in (Kirchhoff et al., 2012) for English-Spanish translation of texts from publich health domain, however without any relation to post-editing task. (Popović and Ney(, 2011)

number of	quality level				
sentences	ok	edit+	edit	edit-	bad
fr-en 2011	323	1559	0	544	99
en-es 2011	31	399	0	550	20
en-es 2012	200	548	856	576	74

Table 1: Corpus statistics: number of sentences assigned to each of the quality levels.

describe a method for automatic classification of machine translation errors into five categories, but only using independent human reference translations, not post-edited translation outputs.

The aim of this work is to systematically explore the relations of five different types of edit operations with the cognitive and the temporal effort. To the best of our knowledge, such study has not yet been carried out. Classification of edit operations is based on the edit distance and is performed automatically, and human quality level scores are used as a measure of cognitive effort.

2 Method and data

Experiments are carried out on 2525 French-to-English and 1000 English-to-Spanish translated sentences described in (Specia, 2011) as well as 2254 English-to-Spanish sentences used for training in the 2013 Quality Estimation shared task (Callison-Burch et al., 2012). All translation outputs were generated by statistical machine systems. For each sentence in these corpora, a human annotator assigned one of four or five quality levels as a measure for the cognitive effort:

- acceptable (ok)
- almost acceptable, easy to post-edit (edit+)
- possible to edit (edit)
- still possible to edit, better than from scratch (edit-)
- very low quality, better to translate from scratch than try to post-edit (bad)

Numbers of sentences assigned to each quality level are presented in Table 1.

All sentences were post-edited by the same two human translators[1] which were instructed to perform the minimum number of edits necessary to make the translation acceptable. Post-editing time is measured on the sentence level in a controlled way in order to isolate factors such as pauses between sentences.

The technical effort is represented by following five types of edit operations:

- correcting word form
- correcting word order
- adding omission
- deleting addition
- correcting lexical choice

The performed edit operations are classified on the word level using the Hjerson automatic tool (Popović, 2011) for error analysis. The post-edited translation output was used as a reference translation, and the results are available in the form of raw counts and edit rates for each category. Edit rate is defined as the raw count of edited words normalised over the total number of words i.e. sentence length of the given translation output.

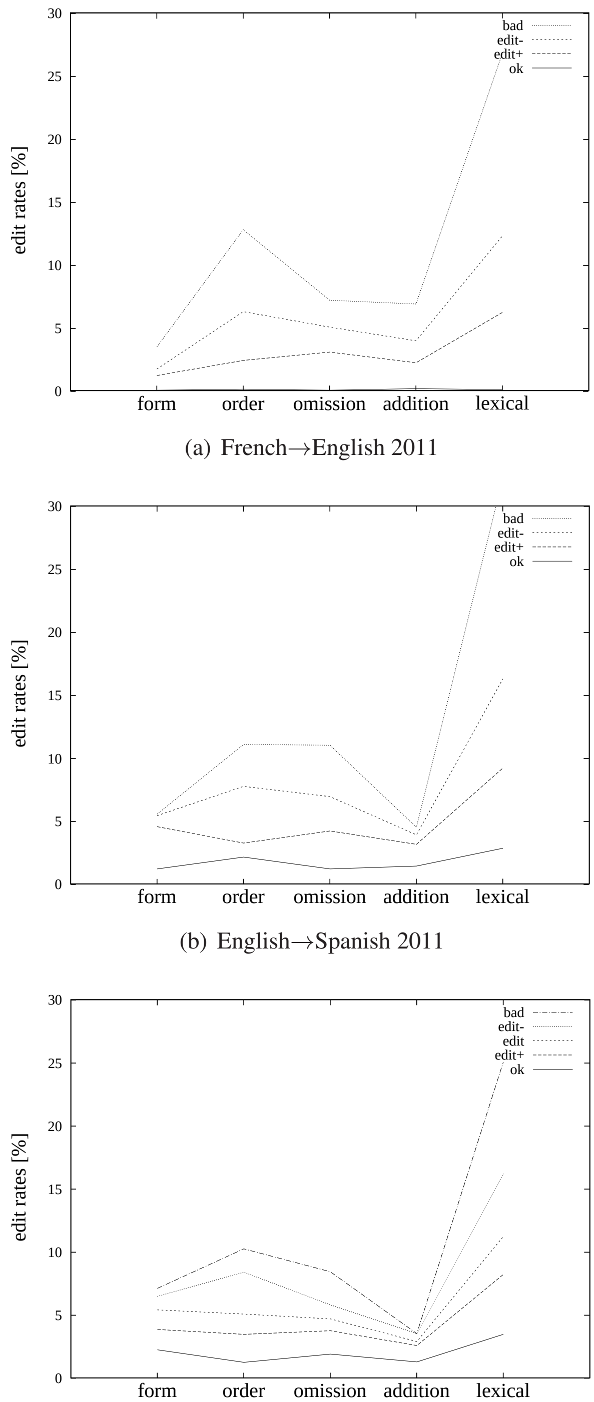
3 Results

3.1 Edit operations and quality level

The distributions of five edit rates for different quality levels are presented in Figure 1. All edit rates increase with the decrease of quality, lexical choice and word order being the most prominent. The main difference between two edit types is that the number of lexical edits increases monotonically whereas the number of reordering edits is relatively low for high quality translations and relatively high for low quality translations.

Impact of reordering distance: In addition to five basic error types, we analysed reordering distances, i.e., the number of word positions by which a particular word is shifted. Reordering distances for different quality levels are presented in Figure 2. It can be seen that the distant reorderings are not an important issue, even for low quality translations, whereas the number of local and longer range reorderings both increase as quality decreases. The increase of longer ones, however, is more prominent for the low-quality translations: this relationship means that the increase of overall reordering errors presented in Figure 1 is primarily due to these reorderings. It should be noted that the experiments were carried out only on the language

[1] One for French-English and one for English-Spanish output.



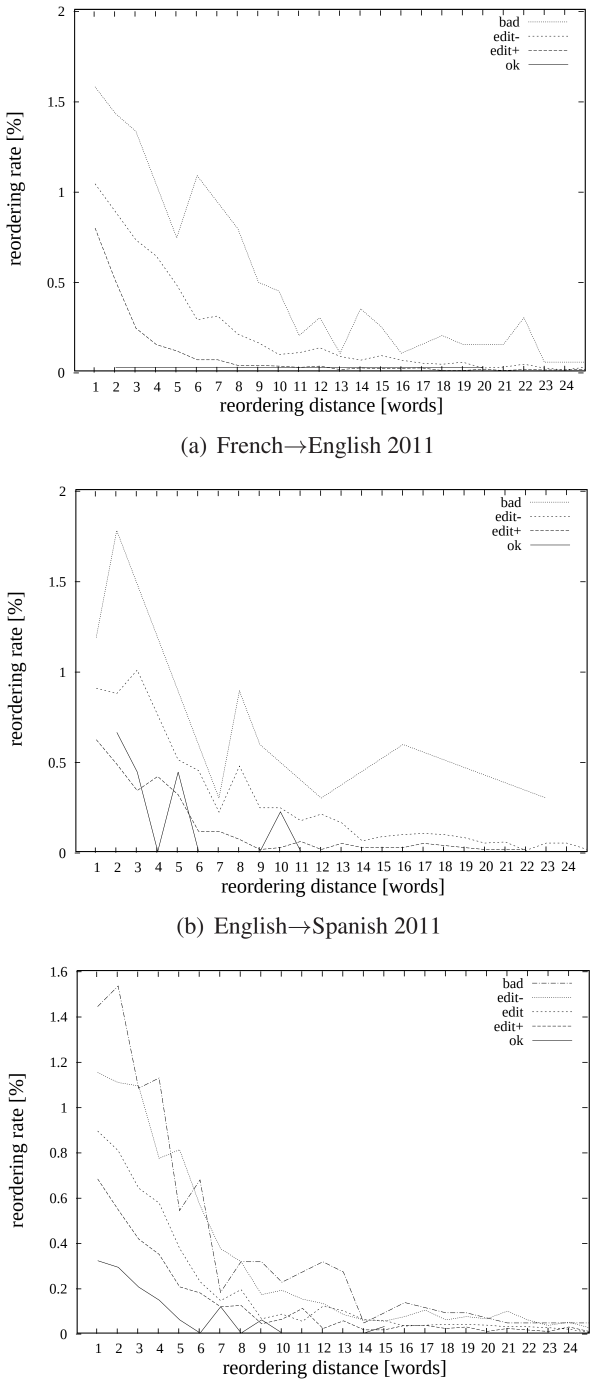
(a) French→English 2011

(b) English→Spanish 2011

(c) English→Spanish 2012

Figure 1: Distribution of five edit types for different quality levels in (a) one French-to-English and (b) two English-to-Spanish translation outputs.

pairs with prevailing local structure differences – future experiments should include languages with different structure, such as German.



(a) French→English 2011

(b) English→Spanish 2011

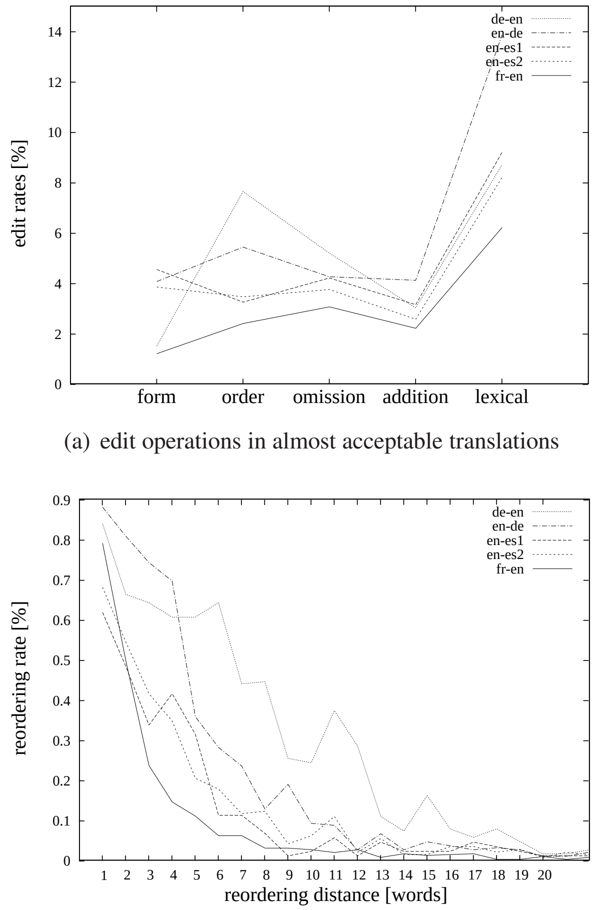
(c) English→Spanish 2012

Figure 2: Distribution of reordering distances for different quality levels in (a) one French-to-English and (b),(c) two English-to-Spanish translation outputs.

3.1.1 Almost acceptable translations

In addition to exploring different quality levels, we carried out an analysis only on almost acceptable translations for different language pairs. Almost acceptable translations are of the special interest for high-quality machine translation – they are namely close to perfect translations and do not require much post-editing effort. The main question is which types of errors are keeping these translations from perfect.

For analysis of almost acceptable translations, apart from the sentences assigned to the “edit+” category in Table 1, an additional corpus was available, namely a portion of the German-to-English (778 sentences) and English-to-German (955 sentences) translations obtained by the best ranked statistical and rule based systems in the framework of the 2011 shared task (Callison-Burch et al., 2011).



(a) edit operations in almost acceptable translations

(b) reordering distances in almost acceptable translations

Figure 3: Distribution of (a) five edit operations and (b) reordering distances in almost acceptable translations: French-to-English, two English-to-Spanish, German-to-English and English-to-German outputs.

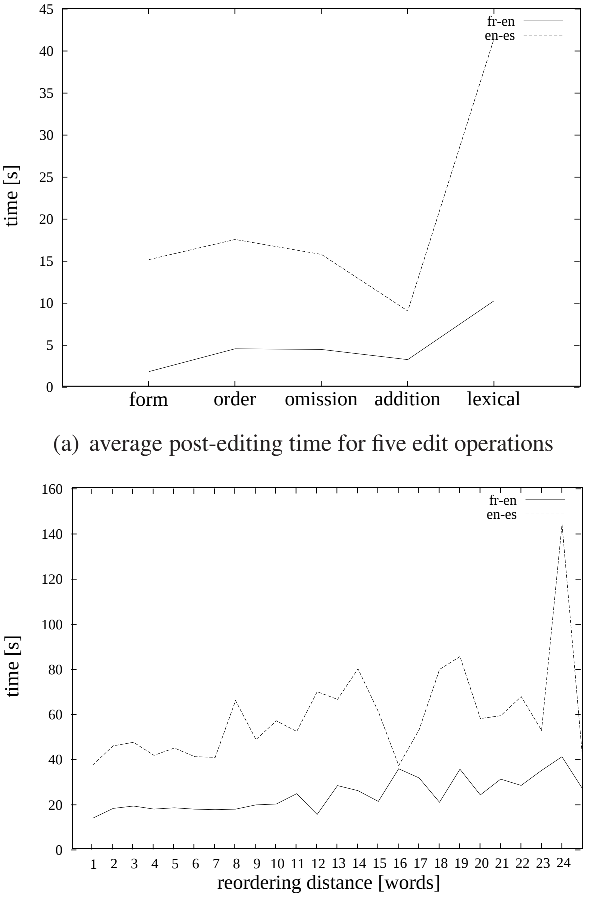
Distributions of five edit types as well as reordering distances in five almost acceptable sets are shown in Figure 3 and it can be seen that they are largely dependent on language pair and translation direction. The lexical edits are the most prominent for all translation directions indicating that even in the high-quality translations, large portions of texts are mistranslated. Inflectional errors are rare in high-quality English outputs, but still relatively high in Spanish and German translations. As for reordering errors, for French-English and English-Spanish translations the reordering edit rates are low, less than 4%, however for German-to-English translations it is almost 8% being not much lower than the lexical edit rate. This high rate indicates that, for this translation direction, even high-quality translations contain a significant number of syntactic errors. English-to-German, conversely, is quite difficult in general and the reordering edit rate is comparable to the rates for other types of operations; since all the edit rates are similar, improving any of them should lead to quality increase. As for reordering distances, short range reorderings are dominant in all high-quality translations, and the main difference for German-to-English outputs is due longer range reordering edits. Further analysis (e.g. based on POS tags) is needed to determine exact nature of reordering problems in the high quality translations.

3.2 Edit operations and post-editing time

Post-editing times are available for the 2011 data (first two rows in Table 1). The post-editing times for the English output are much shorter than for the Spanish output, probably due to language differences and/or to the different annotators. In any case, this difference does not represent an issue for estimating distribution of post-editing time over five edit operation classes. For each edit operation type, average post-editing time is calculated in the following way:

- for each sentence, divide the raw count of each edit type by the total number of edit operations thus obtaining weights;
- for each edit type in the sentence, estimate its post-editing time by multiplicating its weight with the whole sentence post-editing time;
- finally, for each edit type average the post-editing time over all sentences.

It should be noted that using uniform weights might be debatable on the sentence level but is sufficiently reliable on the document level. For example, if one sentence contains two lexical errors and one word order error and the editing took 30 seconds, the estimated time for correcting each error type in this sentence is 10 seconds. However, it is theoretically possible that the reordering error actually took 20s and each of the lexical errors took only 5s. Nevertheless, many other sentences with different error distributions will be able to reflect this correctly. Therefore, averaging over all sentences gives a good estimate of post-editing time distribution over edit types. Distribution of post-editing time over reordering distances is calculated in a similar way, and all the results are presented in Figure 4.



(a) average post-editing time for five edit operations

(b) average post-editing time for different reordering distances

Figure 4: Average post-editing time (a) for five types of edit operations and (b) for different reordering distances: French-to-English and English-to-Spanish translation outputs.

It can be seen that the lexical edits require the largest portion of the time for both outputs. For the English translation output, the shortest time is needed for correction of the word form, and the times for other three edit types are similar. For the Spanish output, the deletion of extra words requires much less time than other edit types. As for reordering distances, as expected, longer reorderings require more time.

3.3 Quality level and post-editing time

In previous sections, we compared five edit operation types with cognitive effort and with temporal effort separately. Nevertheless, the relation between these two aspects in the given context is also important to better understand all effects.

Post-editing times for different quality levels for the 2011 data are presented in Figure 5. Although an overall increase of the post-editing time can be observed when quality level decreases (i.e. cognitive effort increases), there is a discrepance for a significant number of sentences, especially for the sentences with low quality level score. In order to explore the reasons for differences between cognitive and temporal effort, further analysis of edit operations is carried out taking into accout both quality level and post-editing time.

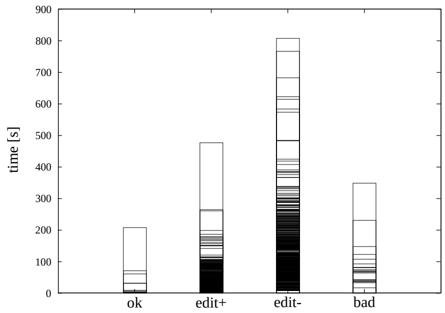


Figure 5: Distribution of post-editing times for different quality levels.

3.4 Analysis of discrepances

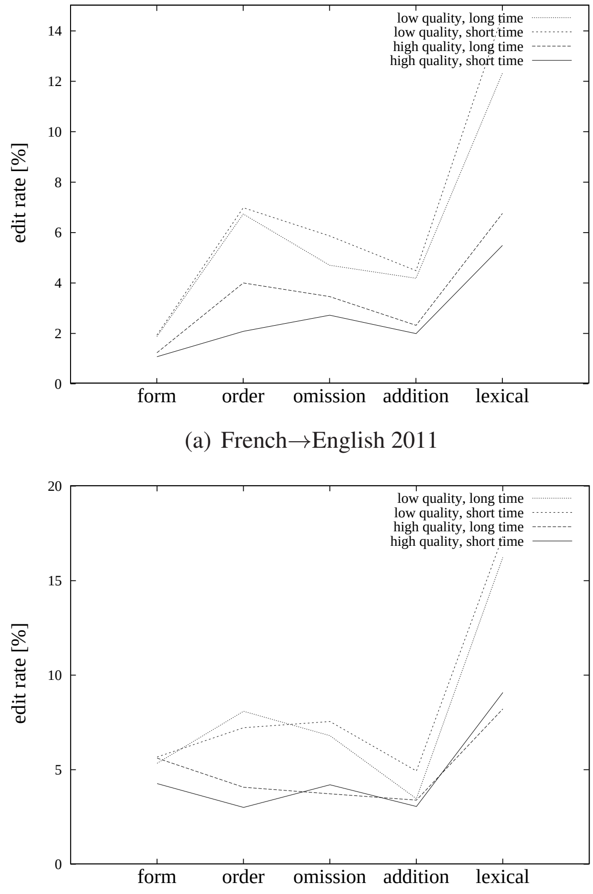
In order to examine differences between the cognitive and the temporal effort, we divided the texts in four parts:

- create two quality subsets: high-quality (edit+ and ok) and low-quality (edit- and bad) sentences

- calculate median post-editing time for low-quality sentences (which is 40 seconds for the English and 100 seconds for the Spanish output) and use it as a threshold

- create two time subsets for both quality subsets according to this threshold: “short-time” and “long-time” sentences.

As a first step, edit rates for each subset are calcuated and the results are shown in Figure 6. The distributions for the same quality are very close – all edit rates are higher for the low-quality sentences regardless of the post-editing time. This indicates that the cognitive effort is tightly related to the amount of particular translation errors, mainly lexical and reordering errors, as already stated in Section 3.1.



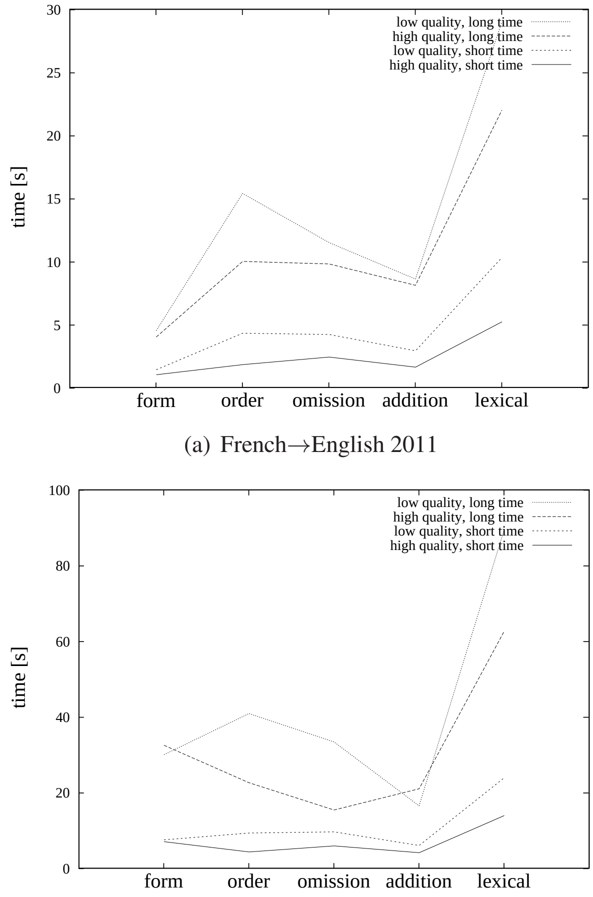
(a) French→English 2011

(b) English→Spanish 2011

Figure 6: Edit rates for five edit operations – analysing discrepances between quality and time; (a) French-to-English and (b) English-to-Spanish output.

The next step was the analysis of post-editing time – what are the causes of long post-editing time for high quality translations and short post-editing time for low quality translations? For each sentence subset, average time distributions over five edit operation types are calculated as described in Section 3.2 and presented in Figure 7. The same tendencies can be observed for both translation outputs:

- all edit types required significantly more time in the long-time sentences than in the short-time sentences regardless of the quality level;
- low-quality translations required more time than high-quality translations in the same time subset;
  - this effect is larger for the long-time sentences,
  - especially for reordering errors, omissions and lexical corrections.



(a) French→English 2011

(b) English→Spanish 2011

Figure 7: Average post-editing times for five edit operations – analysing discrepances between quality and time; (a) French-to-English and (b) English-to-Spanish output.

The results confirm that the lexical and reordering errors require more post-editing effort than the others. In addition, post-editing time for low-quality translations is also affected by omissions, whereas this class has no significant importance in the high-quality translations.

These results also indicate the importance of the sentence length for the post-editing time (which has also been observed in other studies, e.g. (Tatsumi, 2009; Koponen, 2012)). Edit rates are namely raw counts of edit operations normalised over the sentence length: since there is no significant variation of edit rates between the long-time and the short-time subset, the only remaining factor is the sentence length. On the other hand, a number of high-quality sentences require long post-editing time despite of low edit rates: the possible reason is that those sentences are longer.

In order to confirm this assumption, average sentence lengths were calculated for each sentence subset and the results are given in Table 2. As expected, long-time sentences are longer than short-time sentences regardless of the quality level. In addition, the relations of the sentence length with post-editing time and with quality level are presented in Figure 8: the post-editing time increases almost linearly with the increase of the sentence length, whereas the correspondence between the sentence length and the quality level is not straightforward, mainly due to the large number of short low-quality sentences.

quality	time	fr-en	en-es
high	short	22.7	19.6
	long	43.2	31.4
low	short	21.2	19.0
	long	40.6	35.5

Table 2: Average sentence lengths for four sentence subsets based on different quality levels and post-editing times.

4 Summary and outlook

We presented an experiment aiming to explore the relations of five different types of post-edit operations with the cognitive and the temporal post-editing effort. We performed automatic analysis of edit operations for different quality levels and estimated post-editing time for each of the five categories. The results showed that the reordering edits (shifts) and correcting mistranslations correlated most strongly with quality level i.e. cognitive effort, as well as that the lexical errors require the largest portion of post-editing time. Analysis of reordering distances showed that longer range reorderings have more effects both to the quality level and to the post-editing time, however very long ranges do not represent an issue.

In addition, we analysed the edit operations and reordering distances in almost acceptable translations in order to investigate which error types are present in almost perfect high-quality translations preventing them to be completely perfect. It is shown that the error distributions are dependent on the language pair and the translation direction: however, mistranslations are the dominant error type for all translation outputs.

Furthermore, we showed that the edit rates, especially for mistranslations and reorderings, correlate strongly with quality level regardless of the time spent on post-editing. On the other hand, post-editing time strongly depends on the sentence length.

Our experiment offers many directions for future work. First of all, it should be kept in mind that the French-English and English-Spanish language pairs are very similar in the terms of structure and morphology – word order differences are mostly of the local character, and both French and Spanish morphologies are rich mostly due to verbs. In future work, languages with more distinct structural differences (such as German) and richer morphology (such as Czech or Finnish) should be analysed. Furthermore, more details about edit operation types can be obtained by the use of additional knowledge such as POS tags.

Acknowledgments

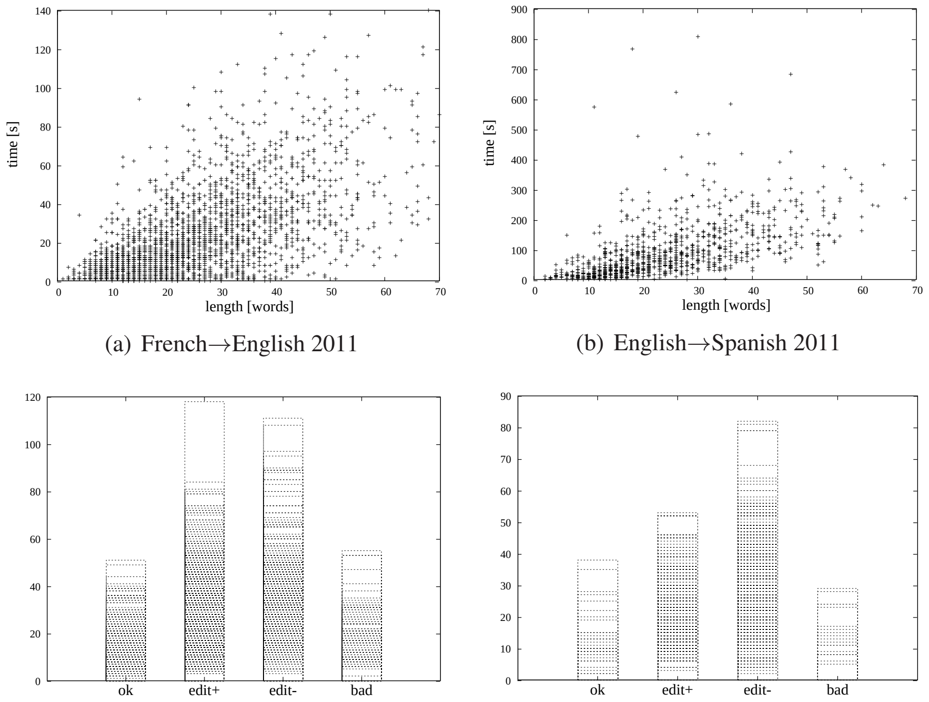
This work has been supported by the project QTLaunchPad (EU FP7 CSA No. 296347).

References

Blain, Frédéric, Jean Senellart, Holger Schwenk, Mirko Plitt, and Johann Roturier. 2011. Qualitative analysis of post-editing for high quality machine translation. In *Machine Translation Summit XIII*, Xiamen, China, September.

Callison-Burch, Chris, Philipp Koehn, Christof Monz, and Omar Zaidan. 2011. Findings of the 2011 Workshop on Statistical Machine Translation. In *Proceedings of the Sixth Workshop on Statistical Machine Translation (WMT 2011)*, pages 22–64, Edinburgh, Scotland, July.

Callison-Burch, Chris, Philipp Koehn, Christof Monz, Matt Post, Radu Soricut, and Lucia Specia. 2012. Findings of the 2012 Workshop on Statistical Machine Translation. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, page 1051, Montral, Canada, June. Association for Computational Linguistics.



(a) French→English 2011

(b) English→Spanish 2011

(c) French→English 2011

(d) English→Spanish 2011

Figure 8: Distribution of post-editing times for (a),(b) different sentence lengths and (c),(d) different quality levels; (a),(c) French-to-English and (b),(d) English-to-Spanish output.

Kirchhoff, Katrin, Daniel Capurro, and Anne Turner. 2012. Evaluating user preferences in machine translation using conjoint analysis. In *Proceedings of the 16th Annual Conference of the European Association for Machine Translation (EAMT 12)*, pages 119–126, Trento, Italy, May.

Koponen, Maarit. 2012. Comparing human perceptions of post-editing effort with post-editing operations. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, pages 181–190, Montral, Canada, June. Association for Computational Linguistics.

Krings, Hans. 2001. Repairing texts: empirical investigations of machine translation post-editing processes. Kent, OH. Kent State University Press.

Popović, Maja. 2011. Hjerson: An Open Source Tool for Automatic Error Classification of Machine Translation Output. *The Prague Bulletin of Mathematical Linguistics*, (96):59–68, October.

Popović, Maja and Hermann Ney. 2011. Towards Automatic Error Analysis of Machine Translation Output. *Computational Linguistics*, 37(4), pages 657–688, December.

Specia, Lucia, Nicola Cancedda, and Marc Dymetman. 2010. A Dataset for Assessing Machine Translation Evaluation Metrics. In *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'2010)*, pages 3375–3378, Valletta, Malta, May.

Specia, Lucia. 2011. Exploiting Objective Annotations for Measuring Translation Post-editing Effort. In *Proceedings of the 15th Annual Conference of the European Association for Machine Translation (EAMT 11)*, pages 73–80, Leuven, Belgium, May.

Tatsumi, Midori. 2009. Correlation between automatic evaluation metric scores, post-editing speed and some other factors. In *Proceedings of MT Summit XII*, pages 332–339, Ottawa, Canada, August.

Tatsumi, Midori, Roturier, Johann. 2010. Source text characteristics and technical and temporal post-editing effort: what is their relationship?. In *Proceedings of the Second Joint EM+/CGNL Worskhop Bringing MT to the user (JEC 10)*, pages 43–51, Denver, Colorado, November.

Temnikova, Irina. 2010. Cognitive evaluation approach for a controlled language post-editing experiment. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta, May.

Wisniewski, Guillaume, Anil Kumar Singh, Natalia Segal, and François Yvon. 2013. Design and analysis of a large corpus of post-edited translations: quality estimation, failure analysis and the variability of post-edition. In *Proceedings of the MT Summit XIV*, pages 117–124, Nice, France, September.

Approaching Machine Translation from Translation Studies – a perspective on commonalities, potentials, differences

Oliver Čulo
Faculty of Translation Studies, Linguistics, and Cultural Studies
University of Mainz
76726 Germersheim, Germany
culo@uni-mainz.de

Abstract

The exchange between Translation Studies (TS) and Machine Translation (MT) has been relatively rare. However, given recent developments in both fields like increased importance of post-editing and reintegration of linguistic and translational knowledge into hybrid systems, it seems desirable to intensify the exchange. This paper aims to contribute to bridging the gap between the two fields. I give a brief account of the changing perspective of TS scholars on the field of translation as a whole, including MT, leading to a more open concept of *translation*. I also point out some potential for knowledge transfer from TS to MT, the idea here centring around the adoption of text-centric notions from TS both for the further development of MT systems and the study of post-editing phenomena. The paper concludes by suggesting further steps to be taken in order to facilitate an intensified future exchange.

1 Introduction

Translation Studies (TS) and Machine Translation (MT) share core goals, the most prominent among them being the study and accomplishment of translation between two languages. Still, exchange has been remarkably rare between the two disciplines in the past decades.

Despite possible reasons for misunderstandings and scepticism, some of them being discussed in section 2, this paper intends to show that intensified exchange between the two fields is possible and even desirable, especially in the light of recent developments. On the one hand, paradigms in MT have been shifting into a direction in which linguistic and translational knowledge is being reintegrated in various ways in hybrid architectures, cf. e.g. (Eisele et al., 2008), by means of adding morphological or word order information , e.g. (Koehn and Hoang, 2007; Collins et al., 2005), adding syntactic information, e.g. (Quirk et al., 2005; Ding and Palmer, 2005), or adapting models to domains, e.g. (Koehn and Schroeder, 2007; Bertoldi and Federico, 2009). On the other hand, TS has seen a rise of empirical, often corpus-based research in various areas, e.g. (Hansen-Schirra et al., 2012; Oakes and Ji, 2012; Rojo and Ibarretxe-Antunano, 2013), which in method and communication style certainly is more accessible to researchers from MT. Last but not least, post-editing – where humans and the machine meet – is of growing importance in the translator's world.

This paper thus addresses some points with regard to fostering exchange between TS and MT. Section 2 gives an account of some emerging views towards a more open concept of the phenomenon called translation. Section 3 makes suggestions how MT could benefit from adopting text-centred notions prominent in TS. Section 4 presents some initial findings from post-editing studies, indicating a case of how post-editing influences the process of translation, thus pointing out the need to further study these two processes in a contrastive manner. Section 5 discusses the observations presented and makes suggestions with respect to potential further directions in the endeavour to bridge the gap between TS and MT. Section 6 then concludes the paper.

Having been written by someone who is aware of some of the developments in MT but usually is concerned with more human-centric issues of translation, it is conceivable that in this paper some of the latest and greatest developments in MT have been missed. While inevitably this opinion piece is shaped in many ways by my personal view of MT, the main goal of this contribution is to give an account of a potential common ground of MT and TS as well as to promote further discussion on this topic.

2 Towards a cluster concept of translation

At first sight, the image that one could come up with for MT and TS is that of unequal twins. While being concerned with the same core goals, the approaches to translation taken by the two fields differ. While MT is often associated with a somewhat mechanistic view of language and seems more interested in "how to make things work", TS emphasises the importance of cultural factors and discusses problems such as (un)translatability or the dichotomy between freedom and loyalty in translation. In short, TS at least is much concerned with the "things that don't work" as with those that work. Also, MT discourse traditionally shows many characteristics of fields like engineering, e.g. the frequent use of mathematical symbols, while TS communication is more discursive in nature. Last but not least, the entity "at work" in the translation process is a very different one. Following Catford's (1965, 31) definitions, I see the translation machine as a device operating with co-textually based algorithms, whereas the human translator follows looser, more contextually based rules and norms which can deliberately be bent or ignored.

Present-day mainstream TS theory liberates the human translator from merely being an inter-operater decoding messages in one language and encoding them in another. Translation is seen as an intentional human act with the goal of producing a text in a target language with a specific relation to the target culture. Rozmyslowicz (in press) discusses this conceptualisation as a cause for a theoretical dilemma: By this definition of translation which emphasises the aspects *agentivity* and *intentionality*, MT is in fact discarded as a type of translation, as the criterion of intentionality is something a machine does not match.

Rozmyslowicz aims at helping to overcome the scepticism towards MT that exists amongst translators, translation scholars, etc., a scepticism which he connects to feelings of uncertainty in a progressively digitalised world. He proposes a view on MT as a tool available to humans; humans, then, would still be the agents in the translation process, as someone has to design and use MT systems. Rozmyslowicz's view might indeed help solve the dilemma of intentionality and agentivity and tear down some of the walls having been erected over time. After all, nobody would think of declaring lexicography as useless or not of interest to TS, and if dictionaries are merely "tools" available to us in the translation process, then so can be MT.

Ultimately, though, it will be necessary to redefine TS in a way which will not rule out MT as a field of interest to translation scholars. Cronin (2012), for instance, goes so far as to define *translation* as a technology by itself and describes the progressing digitalisation as a mere change in the nature of translation. This does not say much, however, about the different perspectives on and approaches to translation and their relation to each other. More promisingly, Tymoczko (2005) puts forward a view on translation as a *cluster concept*, i.e. an open concept in which the various clusters (e.g. linguistic and cultural translation theory, various national or regional traditions, etc.) are connected by family resemblances. Tymoczko also emphasises that the translation concept will in future inevitably extend further due to the ongoing technological changes. Her view underlines the diversity of approaches to translation, perspectives on it, etc, and, by the very meaning of diversity, does not bear any aspect of dominance[1] of one side over the other.

In this paper, I will adopt the views expressed by Tymoczko. If MT is related to human-centric TS by family resemblances, it is necessary to identify the common ground of MT and TS. In the following, I will discuss areas which may be of value to both, by means of knowledge transfer, exchange, or joint research. Of course, only a fraction of possible topics can be addressed here.

[1] As opposed to such concepts like *acceptance* or *tolerance* which I understand to presuppose certain structures of power or dominance, or the struggle for it

3 The translation unit *text* and its implications for Machine Translation

Translators have benefitted from many technical innovations in MT. Translation memories, term databases, and parallel corpora have radically changed the translator's workplace in the past decades; MT proper is set to equally become part of the translation process. This can also work the other way around, as will be argued in the following, with the translation world – or in this case TS – holding things in store that may be valuable to MT. We will look at how TS uses the concept *text* to model translation, and how MT could benefit from adopting notions associated with this concept.

MT has been using notions like domain which, quite obviously, have an effect on lexis, phraseology, and grammar, and of course also on translation. To just pick out one example: Words like *Mutter* should be translated differently depending on the domain a text is rooted in. In general language, *Mutter* will mostly mean 'mother', in engineering it would rather be translated as '(screw-)-nut'. Other notions connected to the concept *text* that seem underrepresented in mainstream MT are text type, translation direction, and text status. Before we turn to investigate these notions, a brief overview of one type of translation theory, functional theory, will serve to highlight the relevance of the concept text for translation.

3.1 Text-centric factors in Translation Studies: the examples of text type, translation direction, text status

In functional translation theory (Nord, 1997; House, 1997; Nord, 2006), the notion of text is predominant. The text as a whole is taken to be the main translation unit, and factors like cultural and situational context as well as purpose of the translation are decisive factors in the process. A text can retain or change its function, either by someone's intention (e.g. when toning down a pamphlet and translating it as political program) or because it is differently received in the target culture than in the source culture. The function of the text is marked on various linguistic levels, from orthography (e.g. progressive vs. conservative spelling in German) to text structure; in other words, the translation unit is not a horizontal, but a vertical phenomenon (Nord, 2011). Moreover, translations can be either documentary, highlighting features of the source text, or instrumental, i.e. appearing and behaving like a target culture text. In terms of functional translation theory, one could characterise MT as a kind of translation which generally aims at being instrumental and functionally constant (i.e. retaining text function).

With text as a key concept for translation, text type is one of the factors that comes into focus. While it is useful to think of translation happening in different domains with all the effects described above, two different text types in the same domain may be of very different nature – even more so in two different languages, thus adding the factor of translation direction to the set of relevant factors. Let us look, for instance, at the business domain. A financial report will be very formal both in English and German. Shareholder letters, however, exhibit various differences in style and grammar: English shareholder letters are of much more colloquial style. Emotive expressions like "We can make it!" remain untranslated in translations from English to German, as they are not deemed appropriate (Čulo et al., 2011). Also English resorts to less formal phrasing than German, regarding e.g. the verb phrase, with English simply using forms of the verb *be* where German uses formulaic expressions such as *betragen* 'amount to' (Čulo, 2010).

Some of the differences in style between English and German shareholder letters can also be quantified in terms of grammar. Part of the CroCo project (Hansen-Schirra et al., 2012) was the study of grammatical properties of originals and translations. The corpus compiled for the study contained a parallel part with texts from 8 registers like computer manuals (INSTR), shareholder letters, or political essays (ESSAY), both with English originals translated to German (E2G) or vice versa (G2E). Each register contains at least ten texts totalling around 30,000 tokens.

One study within this project investigated the shifts of grammatical function that occur in translation. The study was performed on data which were automatically aligned on word level, manually aligned on sentence level, and manually annotated with grammatical functions. All the instances in which two aligned content words (i.e. noun, verb, adjective, or adverb) did not appear in the same grammatical functions in original and translation were counted as indicative of a grammatical shift. Figure 1 shows how the proportion of subject-to-object shifts in relation to all subject shifts varies depending on translation direction and register.

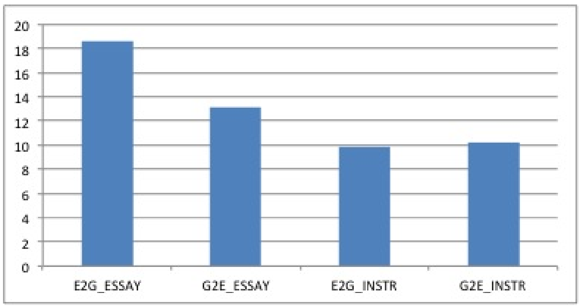


Figure 1: The proportion of subject-to-object shifts in all cases of subject shifts for the registers ESSAY and INSTR and for the two translation directions E2G and G2E

Lexico-grammatical features such as objects in theme position do not only behave differently with respect to translation direction, but also with respect to *text status*. In other words, originals and translations in one language differ in the distribution of these features. For instance, Teich (2003) observes *shining through* of grammatical features: German texts translated from English over-exhibit passive forms when compared to original German texts. Diwersy et al. (in press) analyse a broad range of lexico-grammatic features for English and German originals and translations (amongst others) and make similar observations for features like objects in theme (i.e. sentence initial) position: English translations over-exhibit these, while German translations under-exhibit them when compared to original texts in the same language.

3.2 Existing and potential applications of text-centric concepts

How can such findings as those cited above be of value to MT? That factors like translation direction and text status can be made fruitful for MT purposes has been demonstrated e.g. by Kurokawa et al. (2009). In their experiments, the authors found that they were able to train an equally performant translation model on a fifth of the data size when classifying the training data according to whether they were from originals or translations prior to using them in the training phase, as opposed to training their model on all data available regardless of their status. When considering the findings on the different lexico-grammatical behaviour depending on translation direction and text status, a positive effect on the performance of a translation model was to be expected. Similarly, an adaptation of MT systems to the patent domain, not only to the lexis, but also to its "various stylistic and formatting peculiarities" (Ceauşu et al., 2011, 25) – conforming to the concept of text type – results in significant gains in the system's performance.

The study of linguistic features of translated texts has also been applied to MT products, e.g. by Lapshinova-Koltunski (2013). She compares the distribution of features like nominality vs. verbality in various types of translations such as human translations from scratch, translations made with CAT tools, and translations made by statistical MT systems. She finds, for instance, that output from statistical MT systems tends to be more nominal than output produced by using a translation memory system. She also presents a pilot experiment of how this method of comparison can be extended to more complex features such as verb-last vs. verb-second position for passives in German. Such a metric could be used complementary to existing metrics which are sentence-bound and relate to reference translations, such as BLEU (Papinieni et al., 2002) or METEOR (Banerjee and Lavie, 2005).

From the viewpoint of TS, Lapshinova-Koltunski's method constitutes a text-wide (and thus text-centric) metric, examining how machine translations behave with respect to certain features. This metric could be useful for studying in greater depth the performance of MT systems which are already adapted to a certain domain or text type, as in the case of patent MT, or which in general achieve well higher than average results, comparing them not only to translations, but also to original language.

In a second step, the products of post-editing could be analysed using this metric and contrasted with the feature analysis of the preceding MT product, in order to investigate whether and how the post-editing process influences the outcome of the translation process in contrast to a human from-scratch translation. The following section deals with this question from the viewpoint of lexical consistency.

4 The influence of post-editing on the translation process

Post-Editing (O'Brien, 2010), i.e. the task of correcting MT output, is a process in which human translators and the machine meet. As O'Brien notes, post-editing and revision are similar but different tasks: They differ in such dimensions as the types of errors translators are faced with or the time available. There are studies on the efficiency of post-editing, e.g. with regard to gains in processing time and/or errors typically changed in the post-editing process, e.g. (Groves and Schmidtke, 2009; De Almeida and O'Brien, 2010). The changes observed are typically *local* phenomena, like inserting missing articles or correcting terminology. However, as pointed out above, in terms of functional translation theory, the main unit of translation is the text. The following example, an individual observation from an ongoing pilot study on post-editing, shall highlight that the rendering of textual features may, too, be influenced by the post-editing process.

In a recent pilot study, students and professionals were asked to translate, blind-edit (i.e. edit the MT product without the source text as reference) and post-edit short snippets from newspaper texts. The products from the three processes were contrasted with regard to lexico-grammatical errors as well as with regard to *global* translation strategies like ensuring lexical consistency.

Consistency in translation is ensured by various strategies like determining a terminology to be used, backtracking during translation, or including a drafting phase in the translation workflow. As post-editing already constitutes something like a (first) drafting phase, one would hope that it would aid the goal of reaching consistency in a text. Let us look at the following sentence pair which consists of an original English title of a newspaper article plus its first sentence, one post-edited translation into German, and the gloss of the German translation:

> *Killer nurse receives four life sentences. Hospital nurse C.N. was imprisoned for life today for the killing of four of his patients. (source text)*
>
> *Killer-Krankenschwester zu viermal lebenslanger Haft verurteilt. Der Krankenpfleger C.N. wurde heute auf Lebenszeit eingesperrt für die Tötung von vier seiner Patienten. (post-edited)*
>
> Lit. "Killer female-nurse to four times life-long imprisonment sentenced. The male-nurse C.N. was today for lifetime imprisoned for the killing of four of-his patients."

Besides issues of lexical choice and grammar, there is a noteworthy problem with lexical consistency in the post-editing product. The MT system had in both cases translated *nurse* into the German word *Krankenschwester* which indicates a female nurse, though the text refers to a male nurse. The post-editor failed to edit the first occurrence of *nurse* such that it reflects in German that this is a male nurse (*Krankenpfleger* rather than *Krankenschwester*). The second occurrence was edited accordingly, facilitated by the fact that the gender of the nurse is made explicit by the pronoun *his* in the same sentence.

When looking at the distribution of these errors as shown in Table 4, the picture seems quite clear: This specific error only occurs in the post-editing task, in four out of eight cases; it does so for students and professionals alike. The playback of the translation sessions reveals that in the human translation task four of the translators first translated nurse as Krankenschwester (female nurse) and revised it during the translation of the rest of the text. The remaining four translators read the whole text first or performed a search on the topic in the internet before they started translating. Therefore, they translated *nurse* correctly right from the start. We get very similar results for the blind editing: Four of the editors changed other words/phrases first, before they realised that *Krankenschwester* was not correct, while the other three editors started editing after reading the complete MT output and corrected *Krankenschwester* right away.

Table 1: Number of inconsistent translations for human translation (HT), blind editing (ED), and post-editing (PE)

	HT	ED	PE
(fe)male nurse inconsist.	0 (8)	0 (7)	4 (8)

The point to be made here is thus not that MT "got it wrong". It is more remarkable that half of the post-editors did not seem to care or manage to correct this striking inconsistency. Similar observations are currently being made with regard to terminological consistency in a follow-up study using not general language texts, but LSP texts such as technical documentation; this data is still being evaluated, though.

At this time, I can only speculate about the reasons. It might be that working with two texts in parallel (the source and the MT output) results in a cognitive load which makes it harder to perform other operations. Another possibility is that the post-editors relied more on the MT output than they would admit or even be aware of. We might even be looking at a combination of these factors; however, this remains mere speculation at this point, as I am not aware of any study which investigates such a phenomenon in depth. In any case, something about the post-editing process seems different enough to lead to such errors. While consistency is an important textual criterion, the text-oriented scrutiny of the data from this study will extend to other textual factors like, for instance, the grammatical marking of text function (e.g. addressee vs. content orientation by means of avoiding resp. using impersonal constructions etc.).

5 Discussion

This paper has approached translation as an open concept which includes MT as an area of interest for translation scholars; a view that has been voiced before and has positively evolved in the past years, as described in section 2.

The goal of this paper is to be another step on the way to more intense exchange and collaboration between TS and MT. The sections 3 and 4 depart from text-centric notions prominent in TS and show how some of the phenomena described and studied by means of these can be of common interest to both disciplines. In section 3, I discuss some examples of how factors like translation direction have already successfully been applied in MT. I then propose to extend one of these approaches to make it a text-centric metric for the distribution of linguistic features in MT products and to subsequently use this metric to study the influence of the post-editing task on the outcome of the translation process in contrast to from-scratch translations.

In section 4, I show that the post-editing task can have an influence on the translation process when seen on a textual level and with respect to the global strategy of ensuring lexical consistency. In consequence, this finding emphasising that the two processes seem to differ enough to deserve being studied further; in fact, we might learn a lot more about both kinds of processes by further contrasting them. With respect to the findings presented in section 4, one might be inclined to criticise that such inconsistencies in post-editing may occur due to lack of familiarity with the task. But one might as well reply to this that if the task and the problems were understood and taught well, such inconsistencies and other potential problems should be minimised right from the start (cf. e.g. O'Brien, 2002).

On a more general level, I would suggest several steps to be taken in order to continue establishing a common ground for TS and MT:

- identify more common areas of interest
- identify concepts and methods that can be shared
- define, create, or learn a common or at least mutually understandable terminology
- find platforms for exchange, e.g. common workshops, publication platforms etc.

With this paper, I have attempted to contribute to the first two points.

6 Conclusion

Both TS and MT have seen developments in the past years which have paved new ways for potential collaboration. This paper has addressed some commonalities, potentials, and differences for and between the two disciplines from the perspective of TS. I have laid out some possibilities for knowledge transfer and further collaborative research both in corpus-based translation research as well as process-based research on the human translation process and post-editing. At the end, some more general suggestions as to how exchange could be intensified were made. The views stated and suggestions made in this paper are inevitably influenced by the perspective of the author rooted in human-centric translation and are certainly incomplete. In any case, MT scholars are more than welcome to join the discourse.

References

Banerjee, Satanjeev and Alon Lavie. 2005. METEOR: an automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72.

Bertoldi, Nicola and Marcello Federico. 2009. Domain adaptation for statistical machine translation with monolingual resources. In *Proceedings of the fourth workshop on statistical machine translation*, pages 182–189, Athens, Greece. Association for Computational Linguistics.

Catford, John C. 1965. *A linguistic theory of translation. An essay in applied linguistics*. Oxford University Press, Oxford.

Ceauşu, Alexandru, John Tinsley, Jian Zhang, and Andy Way. 2011. Experiments on domain adaptation for patent machine translation in the PLuTO project.

Collins, Michael, Philipp Koehn, and Ivona Kučerová. 2005. Clause restructuring for statistical machine translation. In *Proceedings of the 43rd annual meeting on association for computational linguistics*, pages 531–540. Association for Computational Linguistics.

Cronin, Michael. 2012. The translation age: translation, technology, and the new instrumentalism. In Venuti, Lawrence, editor, *The translation studies reader*, pages 469–482. Routledge.

Čulo, Oliver, Silvia Hansen-Schirra, Karin Maksymski, and Stella Neumann. 2011. Empty links and crossing lines: querying multi-layer annotation and alignment in parallel corpora. *Translation: Computation, Corpora, Cognition. Special Issue Parallel Corpora: Annotation, Exploitation, Evaluation*, 1(1):75–104.

Čulo, Oliver. 2010. Valency, translation and the syntactic realisation of the predicate. In Vitas, Duško and Cvetana Krstev, editors, *Proceedings of the 29th International Conference on Lexis and Grammar (LGC)*, pages 73–82, Belgrade, Serbia.

De Almeida, Giselle and Sharon O'Brien. 2010. Analysing post-editing performance: correlations with years of translation experience. In *Proceedings of the 14th annual conference of the European association for machine translation, St. Raphaël, France*, pages 27–28.

Ding, Yuan and Martha Palmer. 2005. Machine translation using probabilistic synchronous dependency insertion grammars. In Arbor, Ann, editor, *Proceedings of the 43rd annual meeting of the ACL*, pages 541–548.

Diwersy, Sascha, Stefan Evert, and Stella Neumann. in press. A weakly supervised multivariate approach to the study of language variation. In Szmrecsanyi, Benedikt and Bernhard Wälchli, editors, *Aggregating Dialectology, Typology, and Register Analysis. Linguistic Variation in Text and Speech*, Linguae & Litterae. De Gruyter, New York/Berlin.

Eisele, Andreas, Christian Federmann, Hans Uszkoreit, Saint-Amand Hervé, Martin Kay, Michael Jellinghaus, Sabine Hunsicker, Teresa Herrmann, and Yu Chen. 2008. Hybrid architectures for multi-engine machine translation. In *Translating and the Computer 30*, London, UK.

Groves, Declan and Dags Schmidtke. 2009. Identification and analysis of post-editing patterns for MT. In *MT Summit XII: proceedings of the twelfth Machine Translation Summit*, pages 429–436, Ottawa, Canada.

Hansen-Schirra, Silvia, Stella Neumann, and Erich Steiner. 2012. *Cross-linguistic Corpora for the Study of Translations. Insights from the Language Pair English-German*. De Gruyter, Berlin.

House, Juliane. 1997. *Translation quality assessment. A model revisited*. Gunter Narr Verlag, Tübingen.

Koehn, Philipp and Hieu Hoang. 2007. Factored translation models. In *Proceedings of EMNLP-CoNLL*, pages 868–876.

Koehn, Philipp and Josh Schroeder. 2007. Experiments in domain adaptation for statistical machine translation. In *ACL Workshop on Machine Translation 2007*.

Kurokawa, David, Cyril Goutte, and Pierre Isabelle. 2009. Automatic detection of translated text and its impact on machine translation. *Proceedings. MT Summit XII, The twelfth Machine Translation Summit International Association for Machine Translation hosted by the Association for Machine Translation in the Americas*.

Lapshinova-Koltunski, Ekaterina. 2013. VARTRA: a comparable corpus for the analysis of translation variation. In *Proceedings of the 6th Workshop on Building and Using Comparable Corpora*, pages 77–86, Sofia, Bulgaria.

Nord, Christiane. 1997. *Translating as a purposeful activity. Functionalist approaches explained*. Number 1 in Translation Theories Explained. Jerome, Manchester.

Nord, Christiane. 2006. Translating for communicative purposes across culture boundaries. *Journal of translation studies*, 9(1):43–60.

Nord, Christiane. 2011. Vertikal statt horizontal: die bersetzungseinheit aus funktionaler sicht. In *Funktionsgerechtigkeit und Loyalität. Theorie, Methode und Didaktik des funktionalen Übersetzens*, volume 32 of *TRANSÜD Arbeiten zur Theorie und Praxis des Dolmetschens*. Frank & Time, Berlin.

Oakes, Michael P. and Meng Ji, editors. 2012. *Quantitative methods in corpus-based translation studies: a practical guide to descriptive translation research*. Studies in corpus linguistics; v. 51. John Benjamins Pub. Co, Amsterdam ; Philadelphia.

O'Brien, Sharon. 2002. Teaching post-editing: a proposal for course content. In *Sixth EAMT Workshop*, pages 99–106, Manchester, U.K.

O'Brien, Sharon. 2010. Introduction to PostEditing: who, what, how and where to next? In *Proceedings of AMTA 2010*, Denver, Colorado.

Papinieni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, USA.

Quirk, Christopher, Arul Menezes, and Colin Cherry. 2005. Dependency treelet translation: syntactically informed phrasal SMT. In Arbor, Ann, editor, *Proceedings of the 43rd annual meeting of the ACL*, pages 271–279.

Rojo, Ana and Iraide Ibarretxe-Antunano, editors. 2013. *Cognitive linguistics and translation: advances in some theoretical models and applications*. Number 23 in Applications of Cognitive Linguistics. De Gruyter Mouton, Berlin.

Rozmyslowicz, Tomasz. in press. Machine translation: A problem for translation theory. *New voices in translation studies*.

Teich, Elke. 2003. *Cross-linguistic variation in system and text. A methodology for the investigation of translations and comparable texts*, volume 5 of *Text, Translation, Computational Processing*. Mouton de Gruyter, Berlin/New York.

Tymoczko, Maria. 2005. Trajectories of research in translation studies. *Meta*, 50(4):1082–1097.

Oral Session 6
User Papers

Application of Machine Translation in Localization into Low-Resourced Languages

Raivis Skadiņš[1], Mārcis Pinnis[1], Andrejs Vasiļjevs[1], Inguna Skadiņa[1], Tomas Hudik[2]
Tilde[1], Moravia[2]
{raivis.skadins;marcis.pinnis;andrejs;inguna.skadina}@tilde.lv,
xhudik@gmail.com

Abstract

This paper evaluates the impact of machine translation on the software localization process and the daily work of professional translators when SMT is applied to low-resourced languages with rich morphology. Translation from English into six low-resourced languages (Czech, Estonian, Hungarian, Latvian, Lithuanian and Polish) from different language groups are examined. Quality, usability and applicability of SMT for professional translation were evaluated. The building of domain and project tailored SMT systems for localization purposes was evaluated in two setups. The results of the first evaluation were used to improve SMT systems and MT platform. The second evaluation analysed a more complex situation considering tag translation and its effects on the translator's productivity.

1 Introduction

In recent years, machine translation has received more and more interest from the localization industry. To stay competitive in the market, localization companies have to increase the volume of translation and decrease costs of services. For this reason, the localization industry is increasingly interested in combining translation memories (TM) with machine translation solutions adapted for the particular domain or customer requirements.

Building usable machine translation systems for less-resourced languages with complex morphology and syntax is difficult due to a lack of linguistic resources, on one hand, and the complexity of the language, on the other hand.

The benefits of the application of machine translation in localization are also recognized by developers of computer aided translation (CAT) tools. Such widely used CAT tools as SDL Trados Studio, Kilgray memoQ, ESTeam Translator, Swordfish, MemSource and Wordfast besides traditional translation memory support provides integration with machine translation systems. Several cloud-based platforms offer machine translation services for the localization industry: KantanMT[1], LetsMT[2] and tauyou[3], and others.

This paper describes the methodology used for MT evaluation in localization process and results of two experiments where MT was integrated into CAT tool and used in two professional localization companies – Tilde and Moravia.

In the first experiment we evaluated the impact of in-domain SMT on the productivity of translation of plain text, i.e., text without any formatting. Application of in-domain English-Latvian, English-Czech, English-Hungarian and English-Polish MT systems were evaluated by using MT plug-in to integrate them in the SDL Trados Studio translation environment.

In the second experiment, we set a more complex scenario where translatable documents are slightly out of the domain of the SMT system, contain formatting tags, and are written in a more technical language than in the previous experiment. The second experiment was carried out on English-Latvian, English-Lithuanian, and English-Estonian language pairs. In both experiments, in addition to the productivity evaluation we also performed assessment of the translation quality according to the standard internal quality assessment procedure.

[1] http://www.kantanmt.com
[2] https://www.letsmt.eu
[3] http://www.tauyou.com

2 Related Work

Although experiments on the application of MT for assisting humans in professional translation started more than four decades ago (e.g., Bisbey and Kay 1972; Kay, 1980), it got more attention from the research community only in the late 1990s, with various studies on post-editing and machine translatability (e.g., Berry, 1997; Bruckner and Plitt, 2001). A comprehensive overview of research on machine translatability and post-editing is provided by O´Brien (2005).

Several productivity tests have been performed in translation and localization industry settings at Microsoft, Adobe, Autodesk and others. The Microsoft Research trained SMT on MS tech domain and used it for Office Online 2007 localization into Spanish, French and German. By applying MT to all new words, on average a 5-10% productivity improvement was gained (Schmidtke, 2008).

In experiments performed by Adobe, about 200,000 words of new text were localized using rule-based MT for translation into Russian (PROMT) and SMT for Spanish and French (Language Weaver). Authors reported an increase of translator's daily output by 22% to 51% (Flournoy and Duran, 2009). They also found that quality of MT output varied significantly: while some sentences needed no editing and others required full retranslation.

At Autodesk, a Moses SMT system was evaluated for translation from English into French, Italian, German and Spanish (Plitt and Masselot, 2010). To measure translation time, a special workbench was designed to capture keyboard and pause times for each sentence. Authors reported that although by using MT all translators worked faster, it was in varying proportions from 20% to 131%.

For many years, the Directorate General for Translation (DGT) of the European Commission has probably been the largest user of MT. In 2010, DGT launched its MT@EC project to work on Moses-based SMT for all official EU languages. In July 2013, the first versions of MT@EC systems were released for use in everyday work of translators. The translator's survey (Fontes, 2013) showed that most of MT engines were rated as *'many words or partial phrases reusable with acceptable editing'*. Another conclusion was made regarding quality. According to the feedback for some translation directions, MT quality was excellent (e.g. English-Swedish) but useless for translation from English to Estonian and Hungarian (Verleysen, 2013).

We started our experiments in 2011 with a simplified scenario (Skadiņš et al., 2011). In the following years we extended this evaluation with new languages as described in Section 4 and made a numerous improvements followed by other evaluation experiment as described in Section 5.

3 Methodology

The aim for our experiments was to assess MT impact on translator's productivity and translation quality in a typical localization scenario. For MT application to be useful it has to bring significant improvement in the productivity of translation process - decrease the total time spent on translation while keeping the required level of quality. To assess this we measure:

- translator's productivity,
- quality of translation,
- time spent identifying and correcting errors in the translations.

Unlike in many other post-editing experiments (e.g. Plitt and Masselot, 2010; Teixeira, 2011) where automatic tools were used to measure time spent on individual activities, to log translator key strokes, etc., we evaluated productivity and quality in realistic working environment. In both localization companies, we applied the typical everyday translation workflow using the same tools for process management, time reporting and quality checking as in everyday work.

We ran experiments in two scenarios:

Scenario 1. Translation using TM only (the baseline scenario).

Scenario 2. Translation using TM and MT; MT suggestions are provided for every translation unit that does not have a 100% match in TM.

For training and running SMT systems we used the cloud-based platform LetsMT (Vasiļjevs et al., 2012).

3.1 Data for evaluation

Evaluation was made in the software localization domain for translations from English into target language(s). In this domain, the same sentences frequently appear in different texts (e.g., "Open file") and translators receive such translations (or translations of closely matching sentences) from translation memories of previously translated projects. To take this into account, the following criteria were applied in selecting the source text (documents) for evaluation:

- the documents have not been translated in the organization before;
- about 50% of the documents contain at least 95% new words (texts in less used sub-domain, TM does not contain many segments from this sub-domain);
- about 50% of documents contain sentences with different level of fuzzy matches (texts in typical sub-domains, TM contains segments from this sub-domain).
- The size of each document has to be about 1,000 weighted words on average.

The *weighted word count* is a metric widely used in localization; it means the word count adjusted to take into account the translation effort required. The translator spends less time checking or revising a sentence that has already been translated (exact or fuzzy matches to translation memory) than translating a new sentence (no match in the translation memory). The number of words in the document is therefore "weighted" by the matching rate to the translation memory.

All documents were split into 2 equally sized parts to perform two translation scenarios described above. Texts were selected from user assistance and user interface sub-domains. In the first experiment the following requirements were applied for the selection of the test set:

- Only plain text documents containing no formatting tags,
- Documents related to the topics of the data on which the SMT systems are trained (thus ensuring in-domain translation characteristics of SMT translation suggestions),
- Documents with a similar style and terminology as in the training data used for generating SMT.

For the second experiment a different test set was selected:

- Documents containing text with a mark-up (formatting or tags, placeholders, etc.),
- Documents have to be in the same domain as the data on which the SMT systems were trained, but sub-domains may differ,
- Documents that have different style and terminology to the training data.

The different approaches in the selection of the test sets make the two experiments not comparable. But that was to be expected, as the goals of the two experiments differ significantly.

3.2 Evaluation Process

The evaluation process was the same for all languages. At least 5 translators were involved with different levels of experience and average (or above average) productivity. All translators were trained to use MT systems and SDL Trados Studio 2009 or 2011 in their translation work before the evaluation process started.

In both scenarios, translators were allowed to use whatever external resources they needed (dictionaries, online reference tools, etc.), just as during regular operations.

Translators performed the test without interruption and without switching to other translation tasks during their working day – 8 hours – because splitting the time into short periods would not show reliable evaluation results. Each scenario was performed on a different working day. The time spent for translation was manually reported.

To avoid any “start-up" impact, in *Scenario 2* we removed from the result analysis the first translation task performed by each translator.

3.3 Productivity and Quality Assessment

The translator’s productivity was calculated as a number of weighted words translated per hour.

The translation quality for each document was evaluated by at least 2 experienced editors. Editors were not aware of the scenario used (whether MT was applied or not). Editors reported the time spent on identifying and correcting errors and quality assessment. There was no inter-editor (inter-annotator) agreement measured, as this is not an everyday practice in localization.

The quality of translation is measured by filling in a Quality Assessment (QA) form in accordance with the QA methodology based on the Localization Industry Standards Association (LISA) QA model[4]. The evaluation process involves inspection of translations and classifying errors according to the error categories.

The productivity and quality of work was measured and compared for every individual translator. An error score was calculated for every translation task by counting errors identified by the editor and applying a weighted multiplier based on the severity of the error type. The error score is calculated per 1,000 weighted words and is calculated as:

$$ErrorScore = \frac{1000}{n} \sum_i w_i e_i$$

[4] LISA QA model: http://web.archive.org/web/20080124014404/http://www.lisa.org/products/qamodel/

where n is a weighted word count in a translated text, e_i is a number of errors of type i and w_i is a coefficient (weight) indicating severity of type i errors.

There are 15 different error types grouped in 4 error classes: accuracy, language quality, style and terminology. Different error types influence the error score differently because errors have a different weight depending on the severity of error type. For example, errors of type comprehensibility (an error that obstructs the user from understanding the information; very clumsy expressions) have weight 3, while errors of type omissions/unnecessary additions have weight 2.

Depending on the error score the translation is assigned a translation quality grade (Table 1).

Error Score	Quality Grade
0…9	Superior
10…29	Good
30…49	Mediocre
50…69	Poor
>70	Very poor

Table 1. Quality evaluation based on the score of weighted errors

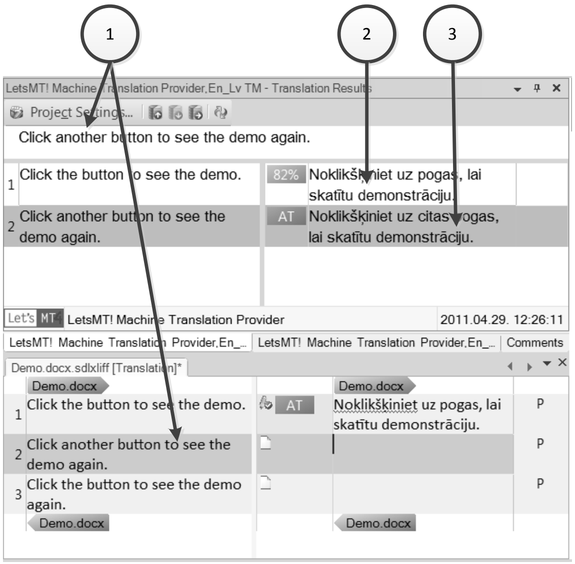
3.4 Tools

The LetsMT (Vasiļjevs et al., 2012) plug-in for the SDL Trados 2009 (or 2011) CAT environment was used in all experiments. It was developed using standard MT integration approach described in SDL Trados SDK.

The plug-in was loaded when the user started SDL Trados Studio. During translation of a document, MT suggestions from the selected MT system are provided as shown in Figure 1.

The *Scenario 1* (baseline) establishes the productivity baseline of the current translation process using SDL Trados Studio when texts are translated unit-by-unit (sentence-by-sentence). The *Scenario 2* measures the impact of MT on the translation process when translators are provided with matches from the translation memory (as in baseline scenario) and with MT suggestions for every translation unit that does not have a 100% match in TM. Suggestions coming from the MT systems are clearly marked; according to Teixeira (2011), identification of suggestion origin helps increase translator performance.

We chose to mark MT suggestions clearly because it allows translators to pay more attention to these suggestions. Usually translators trust suggestions coming from the TM and they make only small changes if necessary. They usually do not double-check terminology, spelling and grammar, because the TM is supposed to contain good quality data. However, translators must pay more attention to suggestions coming from MT, because MT output may be inaccurate, ungrammatical, it may use wrong terminology, etc.



LetsMT! Machine Translation Provider,En_Lv TM - Translation Results

Click another button to see the demo again.

1	Click the button to see the demo.	82%	Noklikšķiniet uz pogas, lai skatītu demonstrāciju.
2	Click another button to see the demo again.	AT	Noklikšķiniet uz citas pogas, lai skatītu demonstrāciju.

LetsMT! Machine Translation Provider 2011.04.29. 12:26:11

LetsMT! Machine Translation Provider,En_... | LetsMT! Machine Translation Provider,En_... | Comments

Demo.docx.sdlxliff [Translation]*

	Demo.docx		Demo.docx	
1	Click the button to see the demo.	AT	Noklikšķiniet uz pogas, lai skatītu demonstrāciju.	P
2	Click another button to see the demo again.			P
3	Click the button to see the demo again.			P
	Demo.docx		Demo.docx	

Figure 1. Translation suggestions in SDL Trados Studio; 1 – source text, 2 – a suggestion from the TM, 3 – a suggestion from the MT

4 Experiment 1

A goal of the first experiment was to test hypothesis that MT can be beneficial in a translator's everyday operations and can increase their productivity. The experiment was performed for four language pairs: English-Latvian, English-Polish, English-Czech and English-Hungarian with domain specific SMT systems.

4.1 MT Systems

The MT systems were slightly different for different language pairs depending on available training resources. We used domain specific training data available to the companies participating in the experiment. For English-Latvian MT we used the best available MT system (Skadiņš et al., 2010) that also includes knowledge about Latvian morphology and some out-of-domain publicly available training data, like DGT-TM (Steinberger et al., 2012) and OPUS EMEA (Tiedemann, 2009).

Two different SMT systems where trained for Polish and Czech. The first Polish MT engine (v1) was trained using all available parallel data from localization company production data (data of various clients); the second MT engine (v2) was trained on smaller client specific data. The first Czech MT engine (v1) was trained using small client specific parallel data from localization company production data and the Czech National Corpus (topic: tech domain)[5]; the second MT engine (v2) was trained using only company production data (data of various clients).

We used the BLEU (Papineni et al., 2002) and METEOR (Denkowski and Lavie, 2011) metrics for the automatic MT system evaluation. The IT domain tuning (2,000 sentences) and testing (1,000 sentences) data were automatically filtered out from the training data before the training process. Table 2 shows details of the MT systems.

MT System	**Size** (sentences)	**Eval. corpus**	**BLEU score**	**METEOR score**
EN-LV	5.37 M*	IT	70.37	N/A
EN-PL v1	1.5 M	IT	70.47	0.48
EN-PL v2	0.5 M	IT	71.90	0.49
EN-CS v1	0.9 M	IT	67.97	0.46
EN-CS v2	1.5 M	IT	71.60	0.49
EN-HU	0.5 M	IT	59.50	0.41

Table 2. Details of the MT systems and results of automatic MT system quality evaluation. * 1.29 M in-domain data.

4.2 Evaluation Data Sets

The data sets for the productivity evaluation were created by selecting documents in the software localization domain from the tasks that had not been translated by the translators in the organizations before the SMT engines were built. This ensures that translation memories do not contain all the segments of texts used in evaluation.

Documents for translation were selected from the incoming work pipeline if they contained about 1,000 weighted words each. Each document was split in half; the first part was translated as described in the baseline scenario (*Scenario 1*), and the second half of the document was translated using the MT scenario (*Scenario 2*). Every document was entered in the translation project tracking system as a separate translation task. The size of evaluation data set varied from 33 to 54 documents, depending on language pair.

All MT systems used in the evaluation were trained using specific vendor translation memories as a significant source of parallel corpora. Therefore, the SMT systems may be considered slightly biased to a specific IT vendor, or a vendor specific narrow IT domain. The evaluation set contained texts from this vendor and another vendor whose translation memories were not included in the training of the SMT system. We will refer to these texts as narrow IT domain and broad IT domain for easier reference in the following sections. From 33% to 50% of texts (depending on language pair) translated in each scenario were in broad IT domain.

4.3 Results

The results are assessed by analysing average values of translator's productivity and an error score for translated texts.

Usage of MT suggestions in addition to the use of TMs increased productivity of the translators in all evaluation experiments (Table 3).

MT System	**Scenario 1** (1)	**Scenario 2** (2)	**Increase** (3)
EN-LV	550	731	32.9 %
EN-PL v1	305	392	28.5 %
EN-PL v2	294	357	21.5 %
EN-CS v1	315	394	25.1 %
EN-CS v2	291	351	20.8 %
EN-HU	287	339	18.0 %

Table 3. Productivity (weighted words translated per hour) evaluation results. (1) Average productivity, *Scenario 1*, (2) Average productivity, *Scenario 2*, (3) Average productivity increase.

There were significant productivity differences in the various translation tasks. The standard deviation of productivity for English-Latvian evaluation in the baseline and MT scenarios were 213.8 and 315.5, respectively. Significant differences in the results of different translators have been observed; the results for English-Latvian evaluation vary from a 64% increase in productivity to a 5% decrease in productivity for one of the translators. Further analysis is necessary, but most likely the differences are caused by the working patterns and skills of individual translators.

At the same time, the error score increased in all but one evaluation experiments (Table 4) still remaining at the quality grade "Good". We have not performed a detailed analysis of the reasons causing an increase in error score, but this can be explained by the fact that translators tend to trust suggestions coming from the CAT tool and do not sufficiently check them, even if they are marked as a MT suggestion.

[5] Institute of Formal and Applied Linguistics (ÚFAL) http://ufal.mff.cuni.cz

MT System	Error score, Scenario 1	Error score, Scenario 2
EN-LV	**20.2**	28.6
EN-PL v1	**16.8**	23.6
EN-PL v2	26.1	**24.2**
EN-CS v1	**19.0**	27.0
EN-CS v2	**19.0**	25.0
EN-HU	**16.9**	22.9

Table 4. Linguistic quality evaluation results

We also analysed how translator productivity and quality is affected by text domain for English-Latvian language pair. Grouping of the translation results by narrow/broad domain attribute reveals that MT-assisted translation provides a better increase in productivity for narrow domain (37%) than for broad domain texts (24%). Error scores for both text types are very similar – 29.1 and 27.6, respectively. The number of errors for each error class is shown in Table 5.

MT System	Accuracy		Language quality		Style		Terminology	
Scenario	**S1**	**S2**	**S1**	**S2**	**S1**	**S2**	**S1**	**S2**
EN-LV	6	9	6	10	3	4	5	7
EN-PL v1	2	4	1	2	3	4	2	3
EN-PL v2	4	4	3	3	4	3	2	3
EN-CS v1	4	6	1	3	3	3	1	2
EN-CS v2	3	5	1	3	2	3	2	3
EN-HU	3	5	2	3	3	4	3	2

Table 5. Comparison by error classes in both *Scenario 1* (S1) and *Scenario 2* (S2).

5 Experiment 2

Although our first experiment showed significant productivity increase, translators were reluctant to use MT in their everyday work. There reason was various mark-ups (tags, placeholders, etc.) which are very frequent in real-life translation segments but were not properly handled by the MT requiring a lot of additional post-editing efforts.

The goal of the second experiment was to evaluate a more complex translation scenario where source documents contain formatting tags, placeholders and differs in used terminology and language style, and thus are slightly out-of-domain for the SMT system than in the previous experiments. We performed this experiment to analyse the LetsMT platform and SMT systems trained on it in a difficult scenario, to find more detailed beneficial aspects of MT usage in localization workflows and to identify areas that require improvements. The experiment was performed for three language pairs: English-Estonian, English-Latvian and English-Lithuanian.

5.1 MT Systems

All three MT systems were trained on proprietary parallel corpora in the IT domain (consisting of user manuals, user interface strings, technical documents, etc.). See Table 6 for the size of the parallel corpora for translation model training.

All systems were trained as typical phrase-based SMT systems using the Moses SMT engine (Koehn et al., 2007) and tuned with the Minimum Error Rate Training (MERT) (Bertoldi et al., 2009). The sentence pairs used for tuning and also automatic evaluation of the SMT systems were randomly extracted from the parallel corpora and manually verified and cleaned by professional translators. The size of the tuning and automatic evaluation data sets were c.a. 2,000 and 1,000, respectively.

Two different English-Latvian MT systems were trained; the second MT system (v2) had much better support for different formatting tags, URLs, numbers and other non-translatable units. The results of the SMT system automatic evaluation are given in Table 6.

MT System	Size (sentences)	BLEU score	METEOR score
EN-LV (v1)	1.70 M	69.57	0.48
EN-LV (v2)	3.80 M	66.98	0.46
EN-LT	2.14 M	59.72	0.43
EN-ET	3.56 M	55.88	0.40

Table 6. Results of automatic MT system quality evaluation for the second experiment.

5.2 Evaluation Data Sets

For all three language pairs of the second experiment, we created the evaluation data sets by selecting documents in the IT domain that had not been translated by the translators before the evaluation. Similarly to the first experiment, this ensured that translation memories did not contain the translatable segments. We also selected documents aiming at different target audiences (system administrators, programmers, everyday users) as well as from vendors contrasting to the ones those translation memories were used in the training of SMT systems (usually having different translation guidelines and writing styles). This ensured that the selected texts were of different linguistic characteristics (including syntax, terminology usage, style, etc.), thus making the translation task more difficult for the SMT systems.

Documents for translation were selected if they contained c.a. 1,000 weighted words each and had formatting tags (on average in ¼ to ⅓ of all translation segments). Similarly to the first experiment, each document was split in half and the first part was translated by the translators without SMT system support (*Scenario 1*) and the second part of the document – using SMT systems (*Scenario 2*). Altogether 100 documents were translated for each language pair by 5 professional translators. Every document was entered in the translation project tracking system as a separate translation task.

Documents for the experiment were selected from four different topics: (1) tablet computer manuals (aimed at general public); (2) programming language manuals (aimed at programmers); (3) navigations software manuals (aimed at general public); and (4) networking system set-up manuals (aimed at system administrators).

5.3 Results

Following the evaluation procedure of the first experiment, we analysed the average values for productivity and the error score for translated texts. We also asked translators to provide system-performance related feedback for more detailed analysis of the experiment.

Language pair	**Productivity changes**	**Standard deviation changes in %**
EN-LV (v1)	-3.10% ± 5.76%	20.80%
EN-ET	-4.70% ± 7.53%	27.17%
EN-LT	-3.76% ± 8.11%	29.28%

Table 7. Productivity changes from *Scenario 1* to *Scenario 2* with a 95% confidence interval

Bearing in mind the complexity of this experiment (formatting tags, more complex language and slight subdomain deviations from the data the SMT system is trained on), the results suggest that the average productivity slightly decreases for all language pairs; however, this cannot be statistically proved in a 95% confidence interval (as shown in Table 7). The large confidence interval is caused by the significant productivity differences (as shown by the changes of the standard deviation of productivity) in the various translation tasks. The average translator productivity with a 95% confidence interval in both translation scenarios is given in Table 8.

Language pair	**Scenario 1**		**Scenario 2**	
	Average productivity	Standard deviation	Average productivity	Standard deviation
EN-LV	576 ± 47	171	558 ± 49	178
EN-ET	470 ± 49	178	448 ± 40	143
EN-LT	728 ± 87	314	700 ± 67	240

Table 8. Average translator productivity and standard deviation of productivity results.

The quality review results for all three language pairs are given in Table 9. The results show a minor decrease of translation quality, from 18.7 to 23.0 points for English-Latvian and from 17.0 to 22.7 points for English-Lithuanian. For English-Estonian the quality of translated texts slightly increased (from 12.9 to 12.0), which is mainly because of "Superior" quality rating for two translators. Although for two language pairs we see a slight drop, the quality evaluation grade is still in the level "Good", which is acceptable for production.

Language pair	**Error score Scenario 1**	**Error score Scenario 2**
EN-LV (v1)	**18.7**	23.0
EN-LT	**17.0**	22.7
EN-ET	12.9	**12.0**

Table 9. Linguistic quality evaluation results of the second experiment

After evaluation, translators submitted informal feedback describing their SMT post-editing experience. Three main directions for further improvements were evident:

- In many cases segments with formatting tags were not translated correctly due to limitations and errors in our implementation of the tag translation functionality.
- As every segment was sent to MT system only at the time of its translation, translators had to wait up to 3 sec. while SMT translation suggestion was provided. Pre-translation or increase of MT speed would solve this problem.
- SMT made a lot of errors in handling and translating named entities, terminology, numbers, non-translatable phrases (e.g., URLs, file paths, etc.).

Since the second experiment, we have actively worked to address the issues raised by the translators. Bugs in the tag translation framework have been fixed, specific non-translatable named entity (e.g., directory paths, URLs, number sequences, etc.) as well as some structured named entity (e.g., dates, currencies) handling has been implemented in the LetsMT platform, and most importantly SMT pre-translation was enabled for the translators. Our preliminary analysis on a small-scale evaluation scenario (following the guidelines of the second experiment) for English-Latvian with two involved translators and 16 translation tasks (8 translation tasks per scenario) shows that the average productivity using the improved LetsMT platform increases from 16.7% up to 35.0% (with a 95% confidence interval) when using SMT support over manual translation without SMT support. This suggests that even for very difficult scenarios SMT systems can be beneficial and lead to significant productivity increases.

Acknowledgements

The research leading to these results has received funding from the research project “2.6. Multilingual Machine Translation” of EU Structural funds, contract nr. L-KC-11-0003 signed between ICT Competence Centre and Investment and Development Agency of Latvia.

References

Berry M. 1997. Integrating Trados translator’s workbench with Machine Translation. *Proceedings of Machine Translation Summit VI.*

Bertoldi N, Haddow B, Fouet J B. 2009. Improved Minimum Error Rate Training in Moses. *The Prague Bulletin of Mathematical Linguistics*, Vol. 91 (2009). Prague, Czech Republic, 7-16.

Bisbey R, and Kay M. 1972. The MIND translation system: a study in man-machine collaboration. *Tech. Rep. P-4786,* Rand Corp.

Bruckner C. and Plitt M. 2001. Evaluating the operational benefit of using machine translation output as translation memory input. *MT Summit VIII, MT evaluation: who did what to whom (Fourth ISLE workshop)*, 61–65.

Denkowski M, and Lavie A. 2011. Meteor 1.3: Automatic Metric for Reliable Optimization and Evaluation of Machine Translation Systems. *Proceedings of the EMNLP 2011 Workshop on Statistical Machine Translation.*

Flournoy R, and Duran C. 2009. Machine translation and document localization at Adobe: From pilot to production. *MT Summit XII: Proceedings of the Twelfth Machine Translation Summit*. Ottawa, Canada.

Fontes H L. 2013. Evaluating Machine Translation: preliminary findings from the first DGT-wide translators’ survey. *Languages and translation*, 02/2013 #6, 6-9.

Kay M. 1980. The proper place of men and machines in language translation. *Tech. Rep. CSL-80-11*. Xerox Palo Alto Research Center (PARC).

Koehn P, Federico M, Cowan B, Zens R, Duer C, Bojar O, Constantin A, Herbst E. 2007. Moses: Open Source Toolkit for Statistical Machine Translation. *Proceedings of the ACL 2007 Demo and Poster Sessions*. Prague, 177-180.

O´Brien S. 2005. Methodologies for measuring the correlations between post-editing effort and machine translatability. *Machine Translation*, 19(1):37–58.

Papineni K, Roukos S, Ward T, Zhu W. 2002. BLEU: a method for automatic evaluation of machine translation. *Proceedings of the 40th Annual Meeting of the Association of Computational Linguistics (ACL).*

Plitt M, and Masselot P. 2010. A Productivity Test of Statistical Machine Translation Post-Editing in a Typical Localisation Context. The Prague Bulletin of Mathematical Linguistics, 93 (January 2010), 7–16.

Schmidtke D. 2008. Microsoft office localization: use of language and translation technology.

Skadiņš, R., Goba, K., & Šics, V. 2010. Improving SMT for Baltic Languages with Factored Models. In *Proceedings of the Fourth International Conference Baltic HLT 2010, Frontiers in Artificial Intelligence and Applications, Vol. 2192* (pp. 125–132). Riga: IOS Press.

Skadiņš, R., Puriņš, M., Skadiņa, I., Vasiļjevs, A. 2011. Evaluation of SMT in localization to under-resourced inflected language. *Proceedings of the 15th International Conference of the European Association for Machine Translation EAMT 2011*, 35-40.

Steinberger R, Eisele A, Klocek S, Pilos S, Schlüter P. 2012. DGT-TM: A freely Available Translation Memory in 22 Languages. *Proceedings of the 8th international conference on Language Resources and Evaluation (LREC'2012)*, Istanbul.

Teixeira, C. 2011. Knowledge of provenance and its effects on translation performance in an integrated TM/MT environment. *Proceedings of the 8th International NLPCS Workshop - Special theme: Human-Machine Interaction in Translation*, 107-118.

Tiedemann J. 2009. News from OPUS - A Collection of Multilingual Parallel Corpora with Tools and Interfaces. *Recent Advances in Natural Language Processing* (vol V). John Benjamins, Amsterdam/Philadelphia, 237-248.

Vasiļjevs, A., Skadiņš, R., & Tiedemann, J. (2012). LetsMT!: a cloud-based platform for do-it-yourself machine translation. *Proceedings of the ACL 2012 System Demonstrations* (pp. 43–48). Jeju Island, Korea: Association for Computational Linguistics.

Verleysen P. 2013. MT@Work Conference: by practitioners for practitioners. *Languages and translation*, 02/2013 #6, 10-11.

SMT of German Patents at WIPO: Decompounding and Verb Structure Pre-reordering

Marcin Junczys-Dowmunt
Adam Mickiewicz University
Information Systems Laboratory
ul. Umultowska 87
61-614 Poznań, Poland
junczys@amu.edu.pl

Bruno Pouliquen
World Intellectual Property Organization
Global Database Service
34, chemin des Colombettes
CH-1211 Geneva
bruno.pouliquen@wipo.int

Abstract

We describe fragments of the SMT pipeline at WIPO for German as a source language. Two subsystems are discussed in detail: word decompounding and verb structure pre-reordering. Apart from automatic evaluation results for both subsystems, for the pre-reordering mechanism manual evaluation results are reported.

1 Introduction

German is one of the 10 official publication languages in which a Patent application can be filed at WIPO [1]. Among the European languages, German proves to be the most challenging one for WIPO's in-house SMT system.

In contrast to French, English, or Spanish, extensive preprocessing has to be applied when German is the source language. In this paper we will illustrate fragments of the Patent SMT pipeline deployed at WIPO that deal with these problems (Pouliquen and Mazenc, 2011). Decompounding has been an established part of the WIPO pipeline, verb structure pre-reordering is a recent addition.

2 German Compound Words

German has the particularity to join individual words into compound words. This is a challenge for SMT as it generates OOV words and data sparseness. Especially patents "suffer" from compound words, e.g. a recent German patent was titled "gasballongetragener flugroboter"[2], both words previously unseen. To solve this problem we apply a "decompounding" process ("gas∼ ballon∼ getragener flug∼ roboter") before training and then proceed with the standard SMT training process.

2.1 Related Work

Koehn and Knight (2003) use parallel texts to train a compound splitter: after aligning the segments, they search for possible splits where each part has a translation as one word in the target segment. POS-information is used as a filter. Popović et al. (2006) experiment with two compound splitting methods (German-English): linguistic and corpus-based and reach similar results for both methods. Junczys-Dowmunt (2008) proposes high-accuracy methods for compound splitting.

At WIPO, decompounding is also used in the in-house developed tools for patent search, CLIR and PATENTSCOPE (Pouliquen and Mazenc, 2011). As our goal is two-fold (SMT and IR), we have to increase precision and recall of our decompounder. Leveling et al. (2011) mentions "Patents have a specific writing style and vocabulary", so we adopt a bottom-up approach learning compound words from the available parallel data. As we plan to use the tool for other languages in the future, no POS information is used.

2.2 Method

We train an SMT system on our parallel English-German data (1.8M segments, 570M English words) and use phrase tables entries as input for the following "compound word guessing" process:

[1] The 10 publication languages under the Patent Cooperation Treaty are Arabic, Chinese, English, French, German, Japanese, Korean, Portuguese, Russian and Spanish.

[2] "gas balloon carried flight robot". We will refer to all German compound words using lowercase letters.

1. Create a German-English dictionary of 1-1 entries (eg. “roboter” → “robot” “gas” → “gas”, “flug” → “flight”) at a probability threshold of 0.01.
2. Create a dictionary of 1-2 entries (eg. “flugroboter” → “flight robot”)
3. Check that segments of two-part decompositions have translations in the 1-1 dictionary (ie. “flug∼ roboter” “flug” → “flight” and “roboter” → “robot”), allowing for “filler” letters like “s” or “er” (e.g “publikations∼ programm”)
4. Create a dictionary of 1-3 entries of compound words used as prefixes (“flugroboterprogramm” → “flight robot program”)
5. Repeat from 3) until no more compound words can be learned.

German compounds words that are commonly translated as single English words are blacklisted (i.e. the “neu∼ ordnung” can be decomposed as “new order”, but “reorganization” is preferred). This blacklist can contain false negatives if English words are compound words themselves, e.g. “roll∼ stühle” → “wheelchair”. Therefore we also check against a German→French list (“roll∼ stühle” → “fauteuil roulant”). To increase the list of compounds we repeat the process for our German-French and join both lists. With time, many compounds are added manually to that list. This results in a list of 644,275 compound words.

The decompounding algorithm is straightforward: we decompose the given word in seen compound words or seen compound segments. A last filter is applied: we check that the average segment length is at least 3.5 characters (avoiding decompositions like “co∼ de∼ bit” for “codebit”).

So far, the longest compound word found in our corpus is “verteil∼ vorrichtung∼ luft∼ strömungs∼ wärme∼ regulierungs∼ kreislauf∼ element∼ kennzeichnungs∼ system”.

2.3 Evaluation

Decompounding is evaluated for English and German in both directions on a small subcorpus of 1 million segments (42 million English words). Table 1 summarizes these results. We observe an improvement of 3 to 4 points BLEU in both cases.

Direction	W/o decomp. BLEU	With decomp. BLEU
en→de	35.18	38.01
de→en	44.86	48.85

Table 1: Automatic evaluation for decompounding

3 Verb Structure Pre-Reordering

German clause structures pose another difficult problem. Often the meaningful part of a German verbal complex appears at the end of the sentence. Patents seem to favour long sentences. Thus, the meaningful verb part may appear at the end of a long sentence, many words away from the subject. Phrase-based SMT is not capable of capturing such long-distance relationships and often fails to translate the verb entirely.

```
V.FIN ⋆ V.(PP|INF) → 1 3 2
V.FIN ⋆ PTKVZ → 3 1 2
^ KON ⋆ PTKZU V.INF → 1 3 4 2
```

Figure 1: Reordering Example Rules

3.1 Related Work

Many approaches for clause restructuring exist, we only refer to a few. For German, Collins et. al (2005) describe a syntactic parsing approach with manually written reordering rules for the parsed trees. Reordering rules inferred automatically from parse trees and word alignments, have been proposed for Chinese (Li et al., 2007).

Syntactic parsing is resource-hungry and time-intensive and cannot be part of our pipeline. Less demanding approaches rely on part-of-speech taggers, see Popović and Ney (2006) for manually written rules or Niehues and Kolss (2009) for automatically induced reordering pattern.

3.2 Our Method

Our approach is a shallow one with manually written rules that rely on POS tags. These rules are combined with selection algorithms that are based on alignment data or if alignment data is unavailable on a maximum entropy classifier. Both, part-of-speech tagger and the maximum entropy classifier, are part of the open-source package Apache OpenNLP[3]. Figure 1 contains a few example rules. The first part consists of regular-expression-like pattern that has to be matched by the POS-tagged sentence. The second part illustrates the reordering operation. Numbers correspond to positions of matched tokens in the pattern.

[3] http://opennlp.apache.org

3.2.1 Alignment-based Reordering Selection

Alignment-based rule selection can only be applied during translation model training. The training procedure is interrupted after word-alignment symmetrization and before phrase table extraction. The source training corpus is reordered and the corresponding alignment is modified to match the newly reordered German sentences.

Algorithm 1 is applied to a source sentence s and the corresponding alignment A. The function matchingRules returns a set of candidate reordering rules applicable to s. Each subset of rules is applied to the input sentence and the input alignment ($\mathcal{P}(M)$ is the powerset of the set of all rules). If the reordered alignment scores better according to linedist than the previous best reordering, the new best reordered sentence, alignment and rule set are preserved. At the end, the overall best candidates are returned. Candidate reorderings are scored based on the distance of the reordered alignment from an idealized line (linear least squares):

$$
\begin{aligned}
a &= \min\{i|(i,j) \in A\} \\
b &= \max\{i|(i,j) \in A\} \\
c &= \min\{j|(i,j) \in A\} \\
d &= \max\{j|(i,j) \in A\}
\end{aligned}
$$

$$\text{linedist}(A) = \sum_{(i,j)\in A} \left(j - \frac{d-c}{b-a}(i-a) + c \right)^2$$

The smaller the distance the more similar is the word order of source and target sentence. Rules in a rule set may be mutually exclusive or overlapping. In that case the rules with the largest matching span take precedence over other rules.

3.2.2 Classifier-based Reordering Selection

During deployment, alignment data is unavailable for unseen sentences and we replace the alignment information with a probabilistic classifier.

The binary maximum entropy classifier used decides whether a rule should be applied ("YES") or not ("NO"). Samples are collected during the translation model training step described above. Figure 2 shows three example samples, table 2 contains applied the feature types. Applied rules are assigned a "YES" all other rules "NO".

Algorithm 2 illustrates the application of the classifier. Matching rules for a German source sentence are identified and features for each rule are generated. If the probability of rule application is higher than the probability of the opposite case the rule is kept and applied to the sentence.

```
Input:
$s$ – source sentence (POS-tagged);
$A$ – word alignment; $R$ – reordering rules;
Output:
Best reordered sentence, alignment, applied rules.
begin
    $\hat{s} \leftarrow s$; $\hat{A} \leftarrow A$; $\hat{M} \leftarrow \emptyset$
    $M \leftarrow \text{matchingRules}(s, R)$
    foreach $M' \in \mathcal{P}(M)$ do
        $(s', A') \leftarrow \text{reorder}(s, A, M')$
        if $\text{linedist}(A') < \text{linedist}(\hat{A})$ then
            $\hat{s} \leftarrow s'$; $\hat{A} \leftarrow A'$; $\hat{M} \leftarrow M'$
        end
    end
    return $(\hat{s}, \hat{A}, \hat{M})$
end
```

Algorithm 1: Reordering by alignment

3.3 Automatic and Manual Evaluation

We favour a high precision tool that should not modify a sentence if it might decrease translation quality. The percentage of reordered sentences varies is 5% to 15%. Improvements in BLEU on the test set (1000 sentences) are moderate, but persist when weights are exchanged between optimization runs with and without pre-reordering to exclude optimizer instability. BLEU results for our systems are reported in Tab. 3, "All" is the full test set, "Diff." reordered sentences (79/1000).

We perform a quick manual evaluation on the 79 changed sentences (Tab. 4). All sentences are evaluated in form of a tournament. Given the source sentence and two outputs, the evaluator declares a win or a draw. System outputs are shuffled,

Feature	Description
`name`	Current rule name
`spanN`	Matched symbol spans
`prevtag`	POS-tag preeceding match
`nexttag`	POS-tag following match
`symN`	Matched rule symbols
`⋆tagN`	POS-tags spanned by *
`other`	Other possible rules

Table 2: Feature types used

```
NO name=^_*_VVIZU_::_2_1 span0=(0,3) span1=(4,4) nexttag=ADJA
  other=PRELS_*_V.*?_::_1_3_2 sym0=^ *tag0=ART *tag0=NN sym0=* sym1=VVIZU
YES name=PRELS_*_V.*?_::_1_3_2 span0=(12,12) span1=(13,17) span2=(18,18)
  nexttag=$, prevtag=$, other=^_*_VVIZU_::_2_1 sym0=PRELS *tag0=ART *tag0=ADJA
  sym1=* sym2=V.*?
NO name=PRELS_*_V.*?_::_1_3_2 span0=(27,27) span1=(28,29) span2=(30,30)
  nexttag=APPR prevtag=$, sym0=PRELS *tag0=NN *tag1=APPR sym1=* sym2=V.*?
```

Figure 2: Samples used for classifier training, first element is class.

```
Input:
$s$ – source sentence (POS-tagged);
$C$ – ME classifier; $R$ – reordering rules;
Output:
Best-scored reordered sentence, applied rules.
begin
    $\hat{M} \leftarrow \emptyset$; $M \leftarrow \text{matchingRules}(s, R)$
    foreach $m \in M$ do
        $\omega \leftarrow \text{features}(s, m, M)$
        if $P_C(\text{YES}|\omega) > P_C(\text{NO}|\omega)$ then
            $\hat{M} \leftarrow \hat{M} \cup \{m\}$
        end
    end
    $s' \leftarrow \text{reorder}(s, \hat{M})$
    return $(\hat{s}, \hat{M})$
end
```

Algorithm 2: Reordering by classifier

the evaluator is unaware which system produced which output. 26 sentences were translated better than their original counterpart, 10 worse and 43 equally good or bad. Among those equally rated 43 sentences, 13 translation were identical.

4 Conclusions

We presented parts of the WIPO patent machine translation pipeline that deal with translation from German. We show that good-practice methods applied in research (e.g. at WMT) can be successfully transferred into user settings (The described method is now in production and publically accessible at: `http://patentscope.wipo.int/translate/`). Decompounding for German achieves good results even with frequent oversplitting. Verb structure reordering is currently very conservative and has only a small but nevertheless beneficial effect on translation from German into other languages.

System	All	Diff.
Baseline	44.91	39.21
Pre-reordered	45.18	41.15

Table 3: BLEU for all and changed sentences

Total	Better	Worse	Equal
79	26 (33%)	10 (13%)	43 (54%)

Table 4: Manual evaluation of pre-reordered sentences compared to original sentences

References

Collins, Michael, Philipp Koehn, and Ivona Kucerova. 2005. Clause Restructuring for Statistical Machine Translation. In *Proc. of ACL*, pages 531–540.

Junczys-Dowmunt, Marcin. 2008. Influence of Accurate Compound Noun Splitting on Bilingual Vocabulary Extraction. In *Proc. of Konvens)*, pages 91–105.

Koehn, Philipp and Kevin Knight. 2003. Empirical Methods for Compound Splitting. In *Proc. of EACL*, pages 187–193.

Leveling, Johannes, Walid Magdy, and Gareth J. F. Jones. 2011. An Investigation of Decompounding for Cross-Language Patent Search. In *Proc. of ACM SIGIR*, pages 1169–1170.

Li, Chi-Ho, Minghui Li, Dongdong Zhang, Mu Li, Ming Zhou, and Yi Guan. 2007. A Probabilistic Approach to Syntax-based Reordering for Statistical Machine Translation. In *ACL*, pages 720–727.

Niehues, Jan, Muntsin Kolss, and Universitt Karlsruhe. 2009. A POS-Based Model for Long-Range Reorderings in SMT. In *Proc. of WMT*, pages 206–214.

Popović, Maja and Hermann Ney. 2006. POS-based Word Reorderings for Statistical Machine Translation. In *Proc. of LREC*, pages 1278–1283.

Popović, Maja, Daniel Stein, and Hermann Ney. 2006. Statistical Machine Translation of German Compound Words. In *Proc. of FinTAL*, pages 616–624.

Pouliquen, Bruno and Christophe Mazenc. 2011. COPPA, CLIR and TAPTA: three tools to assist in overcoming the patent barrier at WIPO. In *Proc of MT-Summit XIII*, pages 24–30, Xiamen, China.

Extrinsic Evaluation of Web-Crawlers in Machine Translation: a Case Study on Croatian–English for the Tourism Domain*

Antonio Toral[†], Raphael Rubino[⋆], Miquel Esplà-Gomis[‡],
Tommi Pirinen[†], Andy Way[†], Gema Ramírez-Sánchez[⋆]
[†] NCLT, School of Computing, Dublin City University, Ireland
{atoral,tpirinen,away}@computing.dcu.ie
[⋆] Prompsit Language Engineering, S.L., Elche, Spain
{rrubino,gramirez}@prompsit.com
[‡] Dep. Llenguatges i Sistemes Informàtics, Universitat d'Alacant, Spain
mespla@dlsi.ua.es

Abstract

We present an extrinsic evaluation of crawlers of parallel corpora from multilingual web sites in machine translation (MT). Our case study is on Croatian to English translation in the tourism domain. Given two crawlers, we build phrase-based statistical MT systems on the datasets produced by each crawler using different settings. We also combine the best datasets produced by each crawler (union and intersection) to build additional MT systems. Finally we combine the best of the previous systems (union) with general-domain data. This last system outperforms all the previous systems built on crawled data as well as two baselines (a system built on general-domain data and a well known online MT system).

1 Introduction

Along with the addition of new member states to the European Union (EU), the commitment with multilingualism in the EU is strengthened to give support to new languages. This is the case of Croatia, the last member to join the EU in July 2013, and of the Croatian language, which became then an official language of the EU.

Croatian is the third official South Slavic language in the EU along with Bulgarian and Slovene. Other surrounding languages (e.g. Serbian and Bosnian), although still not official in the EU, belong also to the same language family and are the official languages of candidate member states, thus being also of strategic interest for the EU.

We focus on providing machine translation (MT) support for Croatian and other South Slavic languages using and producing publicly available resources. Following our objectives, we developed a general-domain MT system for Croatian–English and made it available online on the day Croatia joined the EU. It is, to the best of our knowledge, the first available MT system for this language pair based on free/open-source technologies.

New languages in the EU like Croatian can benefit from MT to speed up the flow of information from and into other EU languages. While this is the case for most types of content it is especially true for official documentation and for content in particular strategic sectors.

Tourism is one of the most important economic sectors in Croatia. It represented 15.4% of Croatia's gross domestic product in 2012 (up from 14.4% in 2011).[1] With almost 12 million foreign tourists visiting Croatia annually, the tourism sector results in income of 6.8 billion euro.

The increasing number of tourists in Croatia makes tourism a relevant domain for MT in order to provide them with quick and up-to-date information about the country they are visiting. Although most visitors come from non-English speaking countries,[2] English is frequently used as a lingua franca. This observation led us to our first approach to support the Croatian tourism sec-

[*] The research leading to these results has received funding from the European Union Seventh Framework Programme FP7/2007-2013 under grant agreement PIAP-GA-2012-324414 (Abu-MaTran).

[1] http://www.eubusiness.com/news-eu/croatia-economy.nrl

[2] According to the site croatia.eu, top emitting countries are Germany (24.2%), Slovenia (10.8%), Austria (8.9%), Italy (7.9%), Czech Republic (7.9%), etc.

tor: to provide MT adapted to the tourism domain from Croatian into English. Later, we will provide MT in the visitors' native languages, i.e. German, Slovene, etc.

We take advantage of a recent work that crawled parallel data for Croatian–English in the tourism domain (Esplà-Gomis et al., 2014). Several datasets were acquired by using two systems for crawling parallel data with a number of settings. In this paper we assess these datasets by building MT systems on them and checking the resulting translation performance. Hence, this work can be considered as an extrinsic evaluation of these crawlers (and their settings) in MT.

Besides building MT systems upon the domain-specific crawled data, we study the concurrent exploitation of domain-specific and general-domain data, with the aim of improving the overall performance and coverage of the system. From this perspective, our case study falls in the area of domain adaptation of MT, following previous works in domains such as labour legislation and natural environment for English–French and English–Greek (Pecina et al., 2012) and automotive for German to Italian and French (Läubli et al., 2013).

The rest of the paper is organised as follows. Section 2 presents the crawled datasets used in this study and details the processing undertaken to prepare them for MT. Section 3 details the different MT systems built. Section 4 shows and comments the results obtained. Finally, Section 5 draws conclusions and outlines future lines of work.

2 Crawled Datasets

Datasets were crawled using two crawlers: ILSP Focused Crawler (FC) (Papavassiliou et al., 2013) and Bitextor (Esplà-Gomis et al., 2010). The detection of parallel documents was carried out with two settings for each crawler: 10best and 1best for Bitextor and reliable and all for FC (see (Esplà-Gomis et al., 2014) for further details). It is worth mentioning that reliable and 1best are subsets of all and 10best, respectively. These subsets were obtained with a more strict configuration of each crawler and, therefore, are expected to contain higher quality parallel text. In addition, a set of parallel segments was obtained by aligning only those pairs of documents which were checked manually by two native speakers of Croatian.

Both Bitextor and FC segment the documents aligned by using the HTML tags. These segments were re-segmented in shorter segments and tokenised with the sentence splitter and tokeniser included in the Moses toolkit.[3]

The resulting segments were then aligned with Hunalign (Varga et al., 2005), using the option `realign`, which provides a higher quality alignment by aligning the output of the first alignment. The documents from each website were concatenated prior to aligning them using tags (`<p>`) to mark document boundaries. Aligning multiple documents at once allows Hunalign to build a larger dictionary for alignment while ensuring that only segments belonging to the same document pair are aligned to each other. The resulting pairs of segments were filtered to remove those with a confidence score lower than 0.4.[4]

From the aligned segments coming from manually checked document pairs we remove duplicate segments. We only keep pairs of segments with confidence score higher than 1.[5] These segments are randomised and we keep two sets, one of 825 segmens for the development set and one of 816 segments for the test set.

From the other 4 datasets, those obtained with the different settings of the two crawlers (1best, 10best, all and reliable), duplicate pairs of segments were also removed. Pairs of segments appearing either in the test or development set were also removed. The remaining pairs of segments are kept and will be used for training MT systems.

Apart from the domain-specific crawled data we use additional general-domain (gen) data gathered from several sources of Croatian–English parallel data: hrenWaC,[6] SETimes[7] and TED Talks.[8] These three datasets are concatenated and will be used to build a baseline MT system.

Table 1 presents statistics (number of sentence pairs, number of tokens and number of unique tokens in source (Croatian) and target (English) language) of the previously introduced parallel datasets for Croatian–English. The table shows

[3] `https://github.com/moses-smt/mosesdecoder`

[4] Manual evaluation for English, French and Greek concluded that 0.4 was an adequate threshold for Hunalign's confidence score (Pecina et al., 2012).

[5] While segment pairs with score above 0.4, as shown above, are deemed to be of reasonable quality for training, we raise the threshold to 1 for test and development data.

[6] `http://nlp.ffzg.hr/resources/corpora/hrenwac/`

[7] `http://nlp.ffzg.hr/resources/corpora/setimes/`

[8] `http://zeljko.agic.me/resources/`

Dataset	# s. pairs	# tokens	# uniq t.
dev	825	30,851	10,119
		34,558	7,588
test	816	28,098	9,585
		31,541	7,366
gen	387,259	8,084,110	288,531
		9,015,757	149,430
1best	27,761	592,236	80,958
		680,067	46,671
10best	34,815	760,884	86,391
		864,326	52,660
reliable	23,225	613,804	71,657
		706,227	37,399
all	27,154	719,526	77,291
		819,353	40,095
union	52,097	1,243,142	103,671
		1,418,950	60,956
intersection	5,939	131,569	28,761
		155,432	16,290

Table 1: Statistics of the parallel datasets. For each dataset the first line corresponds to statistics for Croatian and the second to English.

two additional datasets: union and intersection. These are the union and intersection of datasets 10best and reliable.

3 Machine Translation Systems

Phrase-based statistical MT (PB-SMT) systems are built with Moses 2.1 (Koehn et al., 2007). Tuning is carried out on the development set with minimum error rate training (Och, 2003).

All the MT systems use an English language model (LM) from our system for French→English at the WMT-2014 translation shared task (Rubino et al., 2014).[9] We built individual LMs on each dataset provided at WMT-2014 and then interpolated them on a development set of the news domain (news2012).

Most systems are built on a single dataset, hence they have one phrase table and one reordering table. These systems include a baseline built on the general-domain data (gen), four systems built on the crawled datasets (1best, 10best, reliable and all) and two systems built on the union and intersection of the best performing[10] dataset of each crawler: 10best and reliable.

There is also one system (gen+u) built on two datasets, the general-domain (gen) dataset and a domain-specific dataset (union). Phrase tables from the individual systems gen and union are interpolated so that the perplexity on the development set is minimised (Sennrich, 2012).

System	BLEU	METEOR	TER	OOV
gen	0.4092	0.3005	0.5601	9.5
google	0.4382	0.2947	0.5295	-
1best	0.5304	0.3478	0.4848	7.6
10best	0.5176	0.3436	0.5016	7.2
reliable	0.4064	0.2945	0.5755	12.6
all	0.4105	0.2927	0.5756	12.4
union	0.5448	0.3583	0.4726	6.3
inters.	0.3224	0.2456	0.6582	23.1
gen+u	0.5722	0.3767	0.4451	4.1

Table 2: SMT results.

4 Results

The MT systems are evaluated with a set of state-of-the-art evaluation metrics: BLEU (Papineni et al., 2002), TER (Snover et al., 2006) and METEOR (Lavie and Denkowski, 2009). For each system we also report the percentage of out-of-vocabulary (OOV) tokens.

Table 2 shows the scores obtained by each MT system. We compare our systems to two baselines: a PB-SMT system built on general-domain data (gen) and an on-line MT system, Google Translate[11] (google).

Systems built solely on in-domain data outperform the baselines (1best and 10best) or obtain similar results (reliable and all). Different crawling parameters of the same crawler (10best vs 1best and reliable vs all) do not seem to have much of an impact. In fact, while the scores by 1best are slightly better than scores by 10best, the latter scored slightly better on the development set (and thus it is used in system union).

The union of data crawled by both Bitextor (10best) and FC (reliable) achieves a further improvement over the top performing system built on data by a single crawler (BLEU 0.5448 vs 0.5304). The system built on the intersection is the least performing system (BLEU 0.3224) but it should be noted that this system is built on a very small amount of data (5,939 sentence pairs, cf. Table 1).

Finally a system built on the interpolation of the systems union and gen obtains the best performance, beating all the other systems for all metrics. In the interpolation procedure system union was weighted around 85% and system gen around 15%. Hence, the data provided by the union of the crawlers, although considerably smaller than the general-domain data (52,097 vs 387,259 sentence pairs), is considered more valuable for translating the domain-specific development set.

[9] http://www.statmt.org/wmt14/translation-task.html

[10] According to the BLEU score on the development set.

[11] http://translate.google.com/

5 Conclusions and Future Work

We have presented an extrinsic evaluation of parallel crawlers in MT. Our case study is on Croatian to English translation in the tourism domain.

Given two crawlers, we have built PB-SMT systems on the datasets produced by each crawler using different settings. We have then combined the best datasets produced by each crawler (both intersection and union) and built additional MT systems. Finally we have combined the best of the previous systems (union) with general-domain data. This last system outperforms all the previous systems built on crawled data as well as two baselines (a PB-SMT system built on general-domain data and a well known on-line MT system).

As future work we plan to build MT systems for other relevant languages. As German, Slovene and Italian account for over 50% of incoming tourists in Croatia, we consider of strategic interest to build systems that translate from Croatian into these languages. Even more as it seems that on-line MT systems covering these pairs do not perform the translation directly but use English as a pivot.

Croatian–Slovene is a pair of closely-related languages, already covered by Apertium.[12] We plan to perform domain adaptation on tourism of this rule-based MT system following previous work in this area (Masselot et al., 2010). For the remaining languages (German and Italian), we plan to build SMT systems with crawled data following the approach presented in this paper.

References

Esplà-Gomis, Miquel, Felipe Sánchez-Martínez, and Mikel L. Forcada. 2010. Combining content-based and url-based heuristics to harvest aligned bitexts from multilingual sites with Bitextor. *The Prague Bulletin of Mathemathical Linguistics*, 93:77–86.

Esplà-Gomis, Miquel, Filip Klubička, Nikola Ljubešić, Sergio Ortiz-Rojas, Vassilis Papavassiliou, and Prokopis Prokopidis. 2014. Comparing two acquisition systems for automatically building an English–Croatian parallel corpus from multilingual websites. In *Proceedings of the 9th Language Resources and Evaluation Conference*.

Koehn, Philipp, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, pages 177–180.

Läubli, Samuel, Mark Fishel, Manuela Weibel, and Martin Volk. 2013. Statistical machine translation for automobile marketing texts. In *Machine Translation Summit XIV: main conference proceedings*, pages 265–272.

Lavie, Alon and Michael J. Denkowski. 2009. The meteor metric for automatic evaluation of machine translation. *Machine Translation*, 23(2-3):105–115.

Masselot, François, Petra Ribiczey, and Gema Ramírez-Sánchez. 2010. Using the apertium spanish-brazilian portuguese machine translation system for localisation. In *Proceedings of the 14th Annual conference of the European Association for Machine Translation*.

Och, Franz Josef. 2003. Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, pages 160–167.

Papavassiliou, Vassilis, Prokopis Prokopidis, and Gregor Thurmair. 2013. A modular open-source focused crawler for mining monolingual and bilingual corpora from the web. In *Proceedings of the Sixth Workshop on Building and Using Comparable Corpora*, pages 43–51.

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 311–318.

Pecina, Pavel, Antonio Toral, Vassilis Papavassiliou, Prokopis Prokopidis, and Josef van Genabith. 2012. Domain adaptation of statistical machine translation using web-crawled resources: a case study. In *Proceedings of the 16th Annual Conference of the European Association for Machine Translation*, pages 145–152.

Rubino, Raphael, Antonio Toral, Victor M. Sánchez-Cartagena, Jorge Ferrández-Tordera, Sergio Ortiz-Rojas, Gema Ramírez-Sánchez, Felipe Sánchez-Martínez, and Andy Way. 2014. Abu-MaTran at WMT 2014 Translation Task: Two-step Data Selection and RBMT-Style Synthetic Rules. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, Baltimore, USA. Association for Computational Linguistics.

Sennrich, Rico. 2012. Perplexity minimization for translation model domain adaptation in statistical machine translation. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 539–549.

Snover, Matthew, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and Ralph Weischedel. 2006. A study of translation error rate with targeted human annotation. In *Proceedings of the Association for Machine Translation in the Americas*.

Varga, Dániel, László Németh, Péter Halácsy, András Kornai, Viktor Trón, and Viktor Nagy. 2005. Parallel corpora for medium density languages. In *Proceedings of RANLP*, pages 590–596.

[12] https://svn.code.sf.net/p/apertium/svn/trunk/apertium-hbs-slv/

ISBN 978-953-55375-3-3

Proceedings of the
17th Annual Conference of the
European Association for Machine Translation
EAMT2014

Dubrovnik, Croatia, 16th-18th June 2014

Sponsored by

Bloomberg

EUROPEAN LANGUAGE RESOURCES ASSOCIATION
ELRA

Springer

ISBN 978-953-55375-3-3