Evaluate with Confidence Estimation: Machine ranking of translation outputs using grammatical features

Abstract

We present a pilot study on an evaluation method which is able to rank translation outputs with no reference translation, given only their source sentence. The system employs a statistical classifier trained upon existing human rankings, using several features derived from analysis of both the source and the target sentences. Development experiments on one language pair showed that the method has considerably good correlation with human ranking when using features obtained from a PCFG parser.

1 Introduction

Automatic evaluation metrics for Machine Translation (MT) have mainly relied on analyzing both the MT output against (one or more) reference translations. Though, several paradigms in Machine Translation Research pose the need to estimate the quality through many translation outputs, when no reference translation is given (n-best rescoring of SMT systems, system combination etc.). Such metrics have been known as *Confidence Estimation metrics* and quite a few projects have suggested solutions on this direction. With our submission to the Shared Task, we allow Confidence Estimation metrics to be directly compared with the conventional reference-aware MT metrics.

Our approach suggests building a Confidence Estimation metric using already existing human judgments. This has been motivated by the existence of human-annotated data containing comparisons of the outputs of several systems, as a result of the evaluation tasks run by the Workshops on Statistical Machine Translation (WMT) (Callison-Burch et al., 2008; Callison-Burch et al., 2009; Callison-Burch et al., 2010). This amount of data, which has been freely available for further research, gives an opportunity for applying machine learning techniques to model the human annotators' choices. Machine Learning methods over previously released evaluation data have been already used for tuning complex statistical evaluation metrics (e.g. SVM-Rank in (Callison-Burch et al., 2010)). Our proposition is similar, but works without reference translations. We develop a solution of applying machine learning in order to build a statistical classifier that performs similar to the human ranking: it is trained to rank several MT outputs, given analysis of possible qualitative criteria on both the source and the target side of every given sentence. As qualitative criteria, we use statistical features indicating the quality and the grammaticality of the output.

2 Automatic ranking method

2.1 From Confidence Estimation to ranking

Confidence estimation has been seen from the Natural Language Processing (NLP) perspective as a problem of binary classification in order to assess the correctness of a NLP system output. Previous work focusing on Machine Translation includes statistical methods for estimating correctness scores or correctness probabilities, following a rich search over the spectrum of possible features (Blatz et al., 2004; Ueffing and Ney, 2005; Specia et al., 2009; Raybaud et al., 2009; Rosti et al., 2007).

In this work we slightly transform the binary classification practice to fit the standard WMT human evaluation process. As human annotators have provided their evaluation in the form of ranking of five system outputs at a sentence level, we build our evaluation mechanism with similar functionality, aiming to training from and evaluating against this data. Evaluation scores and results can be then calculated based on comparative analysis of the performance of each system.

2.2 Internal pairwise decomposition

We build one classifier over all input sentences. While the evaluation mechanism is trained and evaluated on a multi-class (ranking) basis as explained above, the classifier is expected to work on a binary level: we provide the features from the analysis of the two system outputs and the source, and the classifier should decide if the first system output is better than the second one or not.

In order to accomplish such training, the *n* systems' outputs for each sentence are broken down to $n \times (n-1)$ pairs, of all possible comparisons between two system outputs, in both directions (similar to the calculation of the Spearman correlation). For each pair, the classifier is trained with a class value c, for the pairwise comparison of system outputs t_i and t_j with respective ranks r_i and r_j, determined as:

$$c(r_i, r_j) = \begin{cases} 1 & r_i < r_j \\ -1 & r_i > r_j \end{cases}$$

At testing time, after the classifier has made all the pairwise decisions, those need to be converted back to ranks. System entries are ordered, according to how many times each of them won in the pairwise comparison, leading to rank lists similar to the ones provided by the humans. Note that this kind of decomposition allows for *ties* when there are equal times of winnings.

2.3 Acquiring features

In order to obtain features indicating the quality of the MT output, automatic NLP analysis tools are applied on both the source and the two target (MT-generated) sentences of every pairwise comparison. Features considered can be seen in the following categories, according to their origin:

- **Sentence length:** Number of words of source and target sentences, source-length / target-length ratio.
- **Target language model:** Language models provide statistics concerning the correctness of the words' sequence on the target language. Such language model features include:
  - the smoothed n-gram probability of the entire target sentence for a language model of order 5, along with
  - uni-gram, bi-gram, tri-gram probabilities and a
  - count of unknown words
- **Parsing:** Processing features acquired from PCFG parsing (Petrov et al., 2006) for both source and target side include:
  - parse log likelihood,
  - number of n-best trees,
  - confidence for the best parse,
  - average confidence of all trees.

  Ratios of the above target features with their respective source features were included.
- **Shallow grammatical match:** The number of occurences of particular node tags on both the source and the target was counted on the PCFG parses. In particular, NPs, VPs, PPs, NNs and punctuation occurences were counted. Then the ratio of the occurences of each tag in the target sentence by its occurences on the source sentence was also calculated.

2.4 Classifiers

The machine learning core of the system was built supporting two classification approaches.

- **Naive Bayes** allows prediction of a binary class, given the assumption that the features are statistically independent.

  $$p(C, F_1, \ldots, F_n) = p(C) \prod_{n}^{i=1} p(F_i|C)$$

  $p(C)$ is estimated by relative frequencies of the training pairwise examples, while $p(F_i|C)$

features	naive Bayes		knn	
	rho	tau	rho	tau
ngram	0.19	0.05	0.13	0.01
unk, len	0.67	0.20	0.73	0.24
unk, len, bigram	0.61	0.21	0.74	0.21
unk, len, ngram	0.63	0.19	0.59	0.21
unk, len, trigram	0.67	0.20	0.76	0.21
unk, len, $\log_{parse}$	0.75	0.21	0.74	0.25
unk, len, n_{parse},VP	0.67	0.24	0.61	0.20
unk, len, n_{parse},VP, $\text{conf}_{bestparse}$	0.78	0.25	0.75	0.24
unk, len, n_{parse},NP, $\text{conf}_{bestparse}$	0.78	0.23	0.74	0.23
unk, len, n_{parse},VP, conf_{avg}	0.75	0.21	0.78	0.23
unk, len, n_{parse},VP, $\text{conf}_{bestparse}$	0.78	0.25	0.75	0.24
unk, len, n_{parse},VP, $\log_{parse}$	**0.81**	**0.26**	0.75	0.23

Table 1: Spearman rho and Kendall tau correlation coefficients achieved on automatic ranking

for our continuous features are estimated with LOESS (locally weighted linear regression similar to (Cleveland, 1979))

- **k-nearest neighbour** (knn) algorithm allows classifying based on the closest training examples in the feature space.

3 Experiment

3.1 Experiment setup

The classifiers for the task were learnt using the German-English testset of the WMT 2008 and 2010 (about 700 sentences)[1]. For testing, the classifiers were used to perform ranking on a test set of 184 sentences which had been kept apart from the 2010 data, with the criterion that they do not contain contradictions among human judgments. It is due to this strict criterion, considered essential for the credibility of the pilot study, that our results are not comparable with other metrics evaluated against full test sets in previous years.

Tokenization was performed with the PUNKT tokenizer (Kiss et al., 2006; Garrette and Klein,), while n-gram features were generated with the SRILM toolkit (Stolcke, 2002). The language model was relatively big and had been built upon all lowercased monolingual training sets for the WMT 2011 Shared Task, interpolated on the 2007 test set. As a PCFG parser, the Berkeley Parser (Petrov and Klein, 2007) was preferred, due to the possibility of easily obtaining complex internal statistics, including *n*-best trees. Unfortunately, the time required for parsing leads to significant delays at the overall processing. The machine learning algorithms were implemented with the Orange toolkit (Demšar et al., 2004).

[1]data acquired from http://www.statmt.org/wmt11

3.2 Feature Selection

Although the automatic NLP tools provided a lot of features (section 2.3), the classifier methods we used, would be expected to perform better given a smaller group of statistically independent features. Since exhaustive training/testing of all possibile feature subsets was not possible, we performed feature selection based on the Relieff method (Kononenko, 1994; Kira and Rendell, 1992). Automatic ranking was performed based on the most promising feature subsets. The results are examined below.

3.3 Results

The performance of the classifier is measured after the classifier output has been converted back to rank lists, similar to the WMT 2010 evaluation. We therefore calculated two types of rank coefficients: averaged Kendall's tau on a segment level, and Spearman's rho on a system level, based on the percentage that the each system's translations performed better than or equal to the translations of any other system.

The results for the various combinations of features and classifiers are depicted on Table 1. Naive Bayes provides the best score on the test set, with $\rho = 0.81$ on a system level and $\tau = 0.26$, trained with features including the number of the unknown words, the source-length by target-length ratio, the VP count ratio and the source-target ratio of the parsing log-likelihood. The number of unknown words particularly appears to be a strong indicator for the quality of the sentence. On the first part of the table we can also observe that language model features do not perform as well as the features deriving from the processing information delivered by the parser. On the second part of the table we compare the use of various grammatical combinations. Finally, the last part contains the correlation obtained by various similar internal parsing-related features.

4 Conclusion and Further work

The experiment presented in this article indicates that confidence metrics trained over human rankings can be possibly used for several tasks of evaluation, given particular conditions, where e.g. there is no reference translation given. Features obtained from a PCFG parser seem to be leading to good correlations, given our test set.

Nevertheless this is still a small-scale experiment, given the restricted data size and the single translation direction. The performance of the system on broader training and test sets will be evaluated in the future. Feature selection is also expected to change if other language pairs are introduced, while more sophisticated machine learning algorithms may also lead to better results.

Acknowledgments

This work was done with the support of the TaraXŰ Project[2], financed by TSB Technologiestiftung Berlin–Zukunftsfonds Berlin, co-financed by the European Union–European fund for regional development.

References

John Blatz, Erin Fitzgerald, George Foster, Simona Gandrabur, Cyril Goutte, Alex Kulesza, Alberto Sanchis, and Nicola Ueffing. 2004. Confidence estimation for machine translation. In *IN M. ROLLINS (ED.), MENTAL IMAGERY*. Yale University Press.

Chris Callison-Burch, 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.

Chris Callison-Burch, 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.

Chris Callison-Burch, 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.

W.S. Cleveland. 1979. Robust locally weighted regression and smoothing scatterplots. *Journal of the American statistical association*, 74(368):829–836.

J. Demšar, B. Zupan, G. Leban, and T. Curk. 2004. Orange: From experimental machine learning to interactive data mining. *Knowledge discovery in databases: PKDD 2004*, pages 537–539.

Dan Garrette and Ewan Klein. An extensible toolkit for computational semantics.

K. Kira and L.A. Rendell. 1992. The feature selection problem: Traditional methods and a new algorithm. In *Proceedings of the National Conference on Artificial Intelligence*, pages 129–129. John Wiley & Sons Ltd.

Tibor Kiss, Jan Strunk, Ruhr universität Bochum, and Ruhr universität Bochum. 2006. Unsupervised multilingual sentence boundary detection. In *Proceedings of IICS-04, Guadalajara, Mexico and Springer LNCS 3473*.

Igor Kononenko. 1994. Estimating attributes: analysis and extensions of relief. In *Proceedings of the European conference on machine learning on Machine Learning*, pages 171–182, Secaucus, NJ, USA. Springer-Verlag New York, Inc.

Slav Petrov and Dan Klein. 2007. Improved inference for unlexicalized parsing.

Slav Petrov, Leon Barrett, Romain Thibaux, and Dan Klein. 2006. Learning accurate, compact, and interpretable tree annotation. In *In ACL '06*, pages 433–440.

S. Raybaud, C. Lavecchia, D. Langlois, and K. Smaili. 2009. Word-and sentence-level confidence measures

[2] http://taraxu.dfki.de

for machine translation. In *European Association of Machine Translation 2009*.

Antti-Veikko Rosti, Necip Fazil Ayan, Bing Xiang, Spyros Matsoukas, Richard Schwartz, and Bonnie J. Dorr. 2007. Combining outputs from multiple machine translation systems. In *In Proceedings of the North American Chapter of the Association for Computational Linguistics Human Language Technologies*, pages 228–235.

Lucia Specia, Marco Turchi, Zhuoran Wang, John Shawe-Taylor, and Craig Saunders. 2009. Improving the confidence of machine translation quality estimates. In *Machine Translation Summit XII*, Ottawa, Canada.

Andreas Stolcke. 2002. Srilm—an extensible language modeling toolkit. In *In Proceedings of the 7th International Conference on Spoken Language Processing (ICSLP 2002*, pages 901–904.

Nicola Ueffing and Hermann Ney. 2005. Word-level confidence estimation for machine translation using phrase-based translation models. *Computational Linguistics*, pages 763–770.