Parallel Sentence Discovery for Low-Resource Languages

Jason Smith
Center for Language and Speech Processing
Johns Hopkins University
3400 N. Charles St.
Baltimore, MD 21218, USA
jsmith@cs.jhu.edu
Advisors: Chris Callison-Burch, Adam Lopez

April 2, 2012

Abstract

One of the most important factors in the performance of a statistical machine translation (SMT) system is the amount and quality of parallel data it is trained on. This has motivated work in which parallel sentences are extracted from document pairs that are not necessarily translations of each other. Collections of such semi-parallel documents are called comparable corpora. In this work, we aim to extract parallel sentences from comparable corpora using a minimal amount of supervision.

1 Introduction

Almost all modern SMT systems are trained using a large collection of translated sentence pairs known as a parallel corpus. Sources of parallel data include parliament proceedings, books, and news articles. While this data may be abundant for some language pairs, such as French/English, it is scarce for most others. In addition, even when parallel data is available, it may not match the domain of the data you wish to translate, and this can have a large effect on performance (Munteanu and Marcu, 2005).

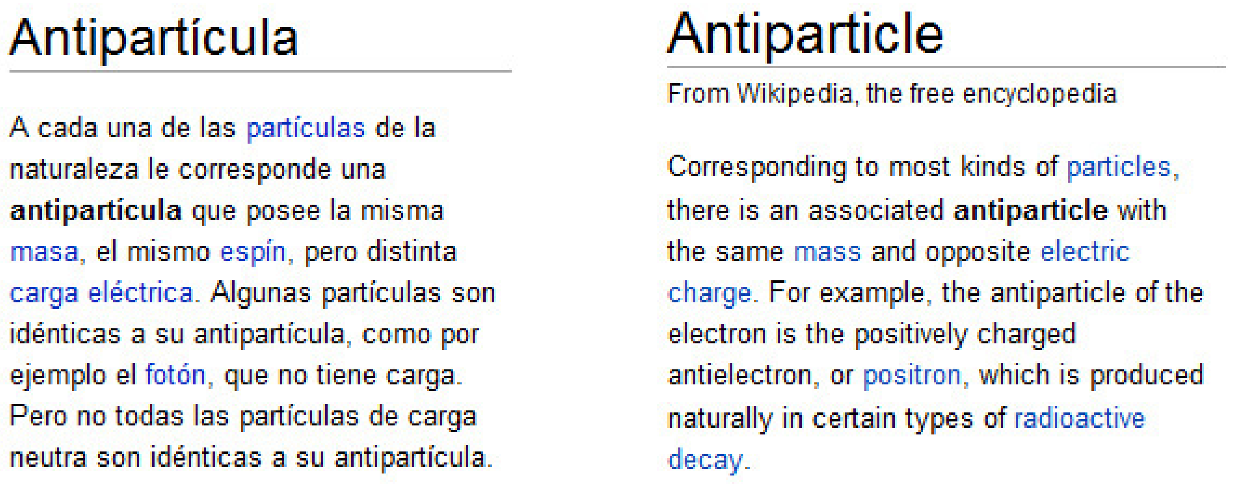
The creation of new parallel corpora can be expensive, especially when bilingual speakers are rare for the language pair of interest. In order to acquire more parallel data without costly human annotation, researchers have looked to corpora which may contain some parallel sentences, but are not completely parallel. Such corpora are referred to as comparable corpora, and examples include multilingual news feeds (Munteanu and Marcu, 2005) and Wikipedia articles (Adafre and de Rijke, 2006; Smith et al., 2010). Most work in extracting parallel sentences from these corpora assumes an initial bilingual dictionary or an existing parallel corpus.

On the other hand, there has also been work on aligning sentences in parallel corpora where the documents may contain $2 : 1$ or $1 : 2$ sentence alignments, or there may be large insertions or deletions of sentences (Gale and Church, 1993; Chen, 1993; Moore, 2002). This work, by contrast, does not require existing parallel data or a bilingual dictionary for the language pair of interest. Instead, the structure of the documents and the lengths of the sentences are used to determine the sentence alignment. Any information about bilingual word correspondence comes from the parallel data that is being aligned.

In this work, we aim to combine techniques from both parallel and comparable sentence alignment to improve the state of the art for parallel sentence extraction from comparable corpora. We will introduce a model for aligning comparable documents which needs only a minimal amount of supervision. Similar to how unsupervised word alignment models can learn their parameters from unlabeled data, we aim to learn parameters for a sentence alignment model from comparable unaligned documents.

2 Sentence Alignment

In this section, we will describe our task and notation. We will view both parallel corpora alignment and the extraction of parallel sentences from comparable corpora as an alignment task. In either type of alignment we are given a set of bilingual document pairs in *source* and *target* languages. When performing parallel corpora alignment, these document pairs will correspond to each other very strongly, while in the case of comparable corpora, some these document pairs may contain no parallel sentences. Munteanu and Marcu (2005) take their document pairs from news stories published at roughly the same time, while Adafre and de Rijke (2006; Smith et al. (2010) use entries from Wikipedia that are on the same topic (Figure 1 gives and example). The task of finding comparable document pairs is not addressed in this work.



Antipartícula

A cada una de las partículas de la naturaleza le corresponde una **antipartícula** que posee la misma masa, el mismo espín, pero distinta carga eléctrica. Algunas partículas son idénticas a su antipartícula, como por ejemplo el fotón, que no tiene carga. Pero no todas las partículas de carga neutra son idénticas a su antipartícula.

Antiparticle

From Wikipedia, the free encyclopedia

Corresponding to most kinds of particles, there is an associated **antiparticle** with the same mass and opposite electric charge. For example, the antiparticle of the electron is the positively charged antielectron, or positron, which is produced naturally in certain types of radioactive decay.

Figure 1: An example of a Spanish/English document pair from Wikipedia.

Each document pair contains a sequence of source sentences (denoted by $\mathbf{S}$) and target sentences (denoted by $\mathbf{T}$). Individual source and target sentences are referred to by S and T respectively. Similarly, we refer to the words within source and target sentences with the lowercase s and t. We borrow the notation of (Och and Ney, 2003) for describing alignments between sentences as subsets of the Cartesian product of sentence positions. Sentence alignments are referred to with the uppercase A, and word alignments with the lowercase a.

The goal of sentence alignment is to identify which sentence pairs in the bilingual document pairs are parallel. We view this as a retrieval task for parallel sentence pairs, and so when annotated sentence alignments are present, we can compute precision, recall, and F-measure.

3 Unsupervised Sentence Alignment

In most previous work on finding parallel sentences in comparable corpora, some initial parallel data (parallel sentences or bilingual dictionary entries) is used as a starting point. This data is used to extract parallel sentences, with the hope that the bilingual word correspondences from the initial data are enough to determine whether or not two sentences are parallel. The obvious drawback is the reliance on the initial data, which may be small. Ideally, one would learn additional word correspondences from parallel sentences that were extracted, and this information could be used to find more parallel sentences. In fact, this bootstrapping method has been used in previous work (Fung and Cheung, 2004b; Fung and Cheung, 2004a; Wu and Fung, 2005).

We will explore a novel way of using semi-supervised learning to find parallel sentences: by including sentence and word alignment in a single model. Much like the IBM word alignment models (Brown et al., 1993) which can be trained on sentence pairs without word alignment data, our model can be trained on document pairs without sentence or word alignment data, and can similarly be trained using the expectation-maximization (EM) algorithm (Dempster et al., 1977).

3.1 Model

First we must define a generative model of a bilingual (possibly) parallel document pair. We will use a joint model of the source and target documents based on stochastic edit distance (Ristad and Yianilos, 1998). Document pairs are generated by a memoryless transducer which generates substitution pairs (S, T), insertion pairs (ϵ, T), deletion pairs (S, ϵ), and the termination pair (ϵ, ϵ), borrowing the convention used by (Oncina and Sebban, 2006) for simplicity. Substitution pairs correspond to parallel source and target sentences, while the insertion and deletion pairs are monolingually generated. For this model to be properly defined, the probability of generating all pairs must sum to one:

$$\sum_{x \in S \cup \{\epsilon\}, y \in T \cup \{\epsilon\}} p(x, y) = 1 \tag{1}$$

Since the insertion and deletion operations are monolingual generation of sentences, we use a standard n-gram language model for their probabilities. For the probability of a substitution pair, we decompose $p(S, T)$ into $p(T|S)p(S)$. $p(T|S)$ is defined by an IBM word alignment model (Brown et al., 1993) (Model 1 in this preliminary work), and $p(S)$ is given by the same language model used to generate deletion pairs ((S, ϵ)). Since $p(S, T)$, $p(S, \epsilon)$ and $p(\epsilon, T)$ all individually sum to one, they must be weighted to ensure that $p(\mathbf{S}, \mathbf{T})$ is properly normalized.[1] In this work, we will use a single parameter to weight these pairs:

$$\begin{aligned} p(S, T) = &\quad \lambda p_{Model1}(T|S) p_{LM}(S) \\ p(S, \epsilon) = &\quad \frac{1-\lambda}{2} p_{LM}(S) \\ p(\epsilon, T) = &\quad \frac{1-\lambda}{2} p_{LM}(T) \end{aligned}$$

p_{Model1} and p_{LM} refer to the IBM Model 1 and a unigram language model, respectively. The parameter λ roughly controls how eager the model is to label sentence pairs as parallel. This can be set based on some prior knowledge about the corpus. p_{Model1} is given by the following equation from (Brown et al., 1993):

$$p(T|S) = p\left(|T| \big| |S|\right) \frac{1}{|S|^{|T|}} \prod_{j=1}^{|T|} \sum_{i=1}^{|S|} p(t_j | s_i) \tag{2}$$

For simplicity, we assume the source sentence S contains the null word. The term $\frac{1}{|S|^{|T|}}$ is the uniform alignment probability. The length distribution, $p\left(|T| \big| |S|\right)$, was originally described as a uniform distribution over a large finite set of lengths. Since Model 1 is usually applied to parallel corpora with observed sentence alignments, and the goal of using Model 1 is to find word translation probabilities ($p(t|s)$), it is unnecessary to find an accurate model of sentence length. However, when the sentence alignments are being learned, it is important to have an accurate model of the length of the target sentence given the source

[1] Since our document pairs are always observed, we can safely ignore the stopping cost $p(\epsilon, \epsilon)$ by assuming it to be some small constant.

sentence. In this work, we use a Poisson distribution to model the target sentence length, following Moore (2002).

The probability for generating sentences monolingually, $p_{LM}(S)$, is a unigram model estimated from the source language documents in the corpus. Similarly, $p_{LM}(T)$ is estimated form the target language documents. While a higher order language model could be learned, we use a unigram model to more closely match IBM Model 1, which can be thought of as a mixture of unigram models (one for each source word and one for the null word) that generate the target sentence. We also use a Poisson distribution to model the lengths of monolingually generated sentences, rather than generating a special end-of-sentence token.

4 Data Collection

In order to evaluate the unsupervised sentence alignment model that we are proposing, we must have bilingual document pairs with an annotated sentence alignment. While existing parallel corpora may be used for this, the document pairs in these corpora are highly parallel and would not resemble the alignments found in Wikipedia articles on the same topic, or comparable news articles. We will instead annotate comparable document pairs with their sentence alignment using Amazon's Mechanical Turk (MTurk).

4.1 Mechanical Turk

MTurk is an online marketplace where people may post collections of tasks that workers may choose to complete for small amounts of money. These tasks are referred to as Human Intelligence Tasks (or HITs) because they are intended to be easy for humans to complete but difficult to automate. Examples of HITs include the identification of offensive images, moderation of forum posts or blog comments, and finding the contact information of a business. The workers on MTurk are referred to as "Turkers". MTurk has also been used for several natural language tasks (Snow et al., 2008), including the evaluation of machine translation output (Callison-Burch, 2009) and even translation itself (Zaidan and Callison-Burch, 2011). The greatest concern when using MTurk for annotation is ensuring that the results are reliable.

There are many ways in which sentence alignment of bilingual comparable documents could be organized into HITs on MTurk. The simplest way would be to take all possible sentence pairs in the document pair, and ask the Turkers to decide whether or not they are parallel. Unfortunately, this will result in far too many tasks to be affordable, as some Wikipedia articles have over a thousand sentences. In order to cut down on the number of tasks, we applied pruning to the candidate sentence pairs.

4.2 Pruning and Data Selection

Our pruning strategy is roughly based on that of Munteanu and Marcu (2005). Sentence pairs are filtered by two criteria. **Length ratio:** The ratio between the lengths (in words) of the two sentences must be below a threshold in each direction. **Coverage:** The percentage of target words t which either have an exact string match with a source word, or have $p(t|s)$ (under IBM Model 1) greater than a threshold for some s in the source sentence. We obtain the Model 1 probabilities by training on existing parallel data and bilingual dictionary entries for the language pair. Coverage is computed on both the source and target sentences, and a sentence pair is filtered if the average coverage falls below a threshold.

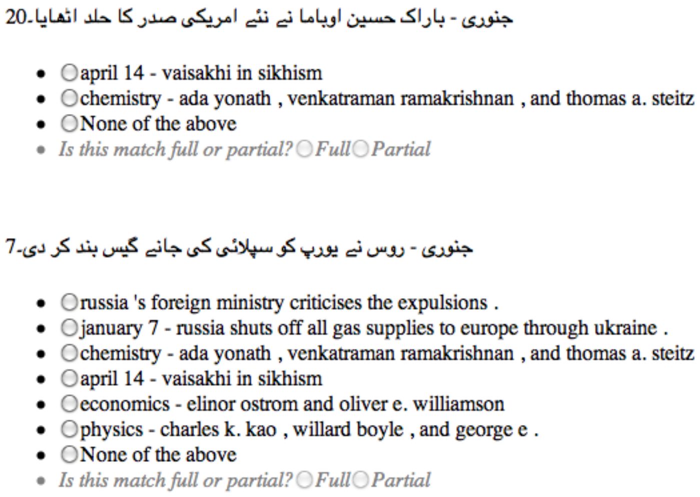
This pruning strategy requires three thresholds to be set: a maximum length ratio, a minimum average source/target coverage, and a minimum Model 1 probability for determining whether or not a word is covered. We tune these thresholds on existing parallel data to ensure that the filter has high recall (90%) while still removing many non-parallel sentence pairs. For our Urdu/English experiments, the thresholds we used were 2.5 for the maximum length ratio, 0.01 for the minimum average coverage, and 0.575 for the Model 1 word coverage threshold. We take our parallel data for training Model 1 parameters from the NIST MT09 Urdu-English training set and the bilingual dictionaries and sentences gathered by Post et al. (2012).

In addition to pruning sentence pairs which are not likely parallel, we also remove any pairs containing sentences with less than five tokens. Wikipedia articles include section headings lists of names (such as an actor's filmography), and links to other articles or external websites. Since our goal is to find parallel sentences, we do not ask Turkers to annotate these very short segments.

Since we are not asking Turkers to annotate all possible sentence pairs from an article pair, evaluation becomes more difficult. We will discuss how we use our partial annotation in Section 4.5.

4.3 Task Design

Our strategy for designing the HITs on MTurk was to give the user an Urdu sentence and a list of up to ten English sentences. The Turker is asked to select which of the English sentences is parallel to the Urdu sentence, or select "None of the above" if none of the English sentences are parallel. We also ask if the sentence pair they find is a partial or full match, and give some examples of each in the instructions. Figure 2 shows an example of one of these questions.



جنوری - باراک حسین اوباما نے نئے امریکی صدر کا حلد اٹھایا-20

- ○april 14 - vaisakhi in sikhism
- ○chemistry - ada yonath , venkatraman ramakrishnan , and thomas a. steitz
- ○None of the above
- *Is this match full or partial?*○*Full*○*Partial*

جنوری - روس نے یورپ کو سپلائی کی جانے گیس بند کر دی-7

- ○russia 's foreign ministry criticises the expulsions .
- ○january 7 - russia shuts off all gas supplies to europe through ukraine .
- ○chemistry - ada yonath , venkatraman ramakrishnan , and thomas a. steitz
- ○april 14 - vaisakhi in sikhism
- ○economics - elinor ostrom and oliver e. williamson
- ○physics - charles k. kao , willard boyle , and george e .
- ○None of the above
- *Is this match full or partial?*○*Full*○*Partial*

Figure 2: The MTurk annotation interface for finding Urdu-English parallel sentences.

Our method of pruning potential sentence pairs may leave us with more than ten candidate English sentences for some Urdu sentences. When this happens, we make additional questions about these Urdu sentences to ensure all candidate pairs are accounted for in the annotation.

In each HIT, we ask the Turkers to annotate up to ten Urdu sentences with their English counterpart (if any), including two control questions with sentences taken from the parallel data described in Section 4.2. There is one positive and one negative control in each HIT. We also request that each HIT be done by three Turkers.

4.4 Data Collection Results

In our first large-scale experiment, we took 92 Urdu-English article pairs, applied our filters as described in Section 4.2, and uploaded our task to MTurk. While there were over 8 million possible sentence pairs in these articles before pruning, we ended up with $785,000$ sentence pairs to be annotated at a total cost of $\$726.80$ (this cost includes the duplicate annotations).

Agreement among the Turkers was high ($\kappa = 0.84$). While the most common answer was "None of the above", there were a substantial number of Urdu sentences which the Turkers found some English counterpart for. For 21.4% of Urdu sentences, at least one Turker found one of the English sentences to be parallel, and in 44.8% of Urdu sentences, at least two Turkers identified a match.

4.5 Evaluation Using Partial Alignments

When we evaluate our sentence pair alignment model, we would like to compute the precision and recall of the proposed sentence alignments. However, since we prune many possible sentence pairs before asking the Turkers for annotation, we cannot be sure whether or not some sentence pairs are parallel. In this section, we will outline a scheme for evaluating sentence alignments using our partially annotated data.

Our primary intrinsic evalutaion metric is alignment F-measure on sentence alignments. This metric could also be seen as F-measure on a parallel sentence pair retrieval task. Let T be the set of true positives (sentence pairs that are truly parallel), and P be the set of predicted positives (sentence pairs identified by our model as parallel). Precision, recall, and F-measure are defined as follows:

$$\begin{aligned} \text{Precison} &= \frac{|T \cup P|}{|P|} \\ \text{Recall} &= \frac{|T \cup P|}{|T|} \\ \text{F-measure} &= \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \end{aligned}$$

When our document pairs are only partially annotated, we will used modified definitions of precision, recall, and F-measure. Let U be the set of sentence pairs which were not annotated as parallel or non-parallel.

$$\begin{aligned} \text{Precison} &= \frac{|T \cup P|}{|P \setminus U|} \\ \text{Recall} &= \frac{|T \cup P|}{|T|} \\ \text{F-measure} &= \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \end{aligned}$$

Since T and U are disjoint, only the definition of precision needs to be modified.

Given the annotations we gathered from MTurk, it is possible to define U in multiple ways. The most conservative method would be to take U to be all sentence pairs not presented to the Turkers. However, if we make the assumption that sentence alignments of the document pairs are $1:1$, then when a Turker annotates a sentence pair (S, T) as parallel, it follows that all (S, T') pairs with $T' \neq T$ and (S', T) pairs with $S' \neq S$ are not parallel. Since the alignments we found were mostly $1:1$, we decided to go with this option.[2] Figure 3 illustrates this method.

4.6 Alternate Evaluation Strategies

Section 4.5 describes a method for using Turkers' partial annotation of a document pair's sentence alignment for intrinsic evaluation of a sentence alignment model. In this section, we will explore other strategies for using the Turkers' output for evaluation.

[2] There were a small number of alignments which were not $1:1$, most of which were image captions.

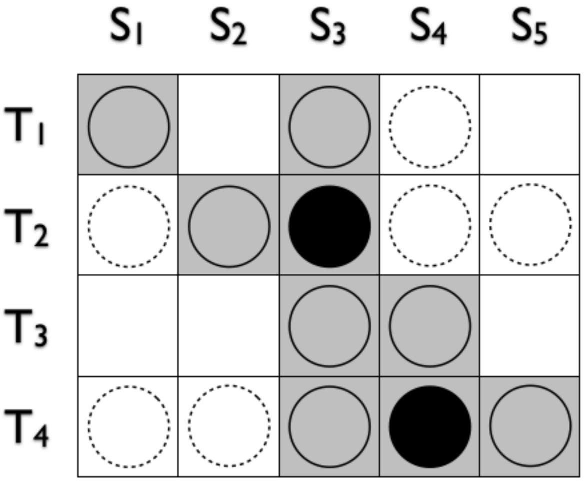


Figure 3: A partial alignment grid for a comparable document pair. The shaded cells in the grid represent the sentence pairs which were presented to the Turkers for annotation. A filled circle indicates the Turker found the sentence pair to be parallel, and an empty circle means the pair is not parallel. The dashed circles represent the sentence pairs we infer to be non-parallel by assuming the sentence alignments are $1:1$.

In our MTurk task setup (see Section 4.3), we collect redundant annotations for each HIT. While this was done primarily for quality control, and it is more convenient to use a single judgement for each sentence pair, we can perform a more fine-grained analysis by looking at the individual Turkers' judgements. Also, we gave the option of labeling a parallel sentence pair as a "partial" or "full" match.

[TODO: For the semi-supervised experiments, we want to treat sentence pairs that any annotator marked as a match as a true positive. We could also measure the inter-annotator agreement of our system against the Turkers. –JS]

5 Experiments

Our first set of experiments uses a semi-supervised setting. We have both parallel sentences (labeled data) and comparable document pairs (unlabeled data), and learn our model's parameters from both of these resources.

Our parallel corpus is taken from the NIST MT09 Urdu-English training set and the bilingual dictionaries and sentences gathered by Post et al. (2012).[3] The parallel sentences from this corpus are treated as single sentence document pairs. Alternatively, the entire training set could be seen as a single document pair whose sentence alignment lies completely on the diagonal. The model described in 3 does not differentiate between these two ways of viewing the corpus. In either case, learning from the parallel sentences is identical to IBM Model 1 training.

The comparable document pairs are a subset of the Wikipedia article pairs that we annotated using MTurk as described in Section 4. 60% of this data was taken as a development set. The remaining 40% of the annotated document pairs was split into two equal sized test sets.[4]

In the following experiments the setup is as follows: We initialize our parameters by running five iterations of EM on the parallel sentences from our labeled data. Then we run several iterations of EM on both

[3] This is the same parallel corpus used to create the sentence pair filters used in collecting the annotated sentence alignments.

[4] This split was done in order to have training, development, and test sets for supervised sentence aligment models.

the labeled data and unlabeled data, measuring performance after each iteration.

5.1 Results

6 Conclusions

References

[Adafre and de Rijke2006] Sisay F. Adafre and Maarten de Rijke. 2006. Finding Similar Sentences Across Multiple Languages in Wikipedia. In *EACL 2006 Workshop on New Text, Wikis and Blogs and Other Dynamic Text Sources*.

[Brown et al.1993] Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Comput. Linguist.*

[Callison-Burch2009] Chris Callison-Burch. 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: Volume 1 - Volume 1*, EMNLP '09, pages 286–295, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Chen1993] Stanley F. Chen. 1993. Aligning sentences in bilingual corpora using lexical information. In *Proceedings of the 31st annual meeting on Association for Computational Linguistics*, ACL '93, pages 9–16, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Dempster et al.1977] A.P. Dempster, N.M. Laird, D.B. Rubin, et al. 1977. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38.

[Fung and Cheung2004a] Pascale Fung and Percy Cheung. 2004a. Mining Very-Non-Parallel Corpora: Parallel Sentence and Lexicon Extraction via Bootstrapping and EM. In *EMNLP 04*.

[Fung and Cheung2004b] Pascale Fung and Percy Cheung. 2004b. Multi-level bootstrapping for extracting parallel sentences from a quasi-comparable corpus. In *Proceedings of the 20th international conference on Computational Linguistics*, COLING '04, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Gale and Church1993] William A. Gale and Kenneth W. Church. 1993. A program for aligning sentences in bilingual corpora. *Comput. Linguist.*, 19:75–102, March.

[Moore2002] Robert Moore. 2002. Fast and Accurate Sentence Alignment of Bilingual Corpora. In Stephen Richardson, editor, *Machine Translation: From Research to Real Users*, volume 2499 of *Lecture Notes in Computer Science*, pages 135–144. Springer Berlin / Heidelberg.

[Munteanu and Marcu2005] Dragos Stefan Munteanu and Daniel Marcu. 2005. Improving Machine Translation Performance by Exploiting Non-Parallel Corpora. *Comput. Linguist.*, 31:477–504, December.

[Och and Ney2003] Franz Josef Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Comput. Linguist.*, 29:19–51, March.

[Oncina and Sebban2006] Jose Oncina and Marc Sebban. 2006. Learning stochastic edit distance: Application in handwritten character recognition. *Pattern Recogn.*, 39:1575–1587, September.

[Post et al.2012] Matt Post, Chris Callison-Burch, and Miles Osborne. 2012. Constructing parallel corpora for six indian languages via crowdsourcing. In *In submission*.

[Ristad and Yianilos1998] Eric Sven Ristad and Peter N. Yianilos. 1998. Learning String-Edit Distance. *IEEE Trans. Pattern Anal. Mach. Intell.*, 20:522–532, May.

[Smith et al.2010] Jason R. Smith, Chris Quirk, and Kristina Toutanova. 2010. Extracting Parallel Sentences from Comparable Corpora using Document Level Alignment. In *NAACL 2010*.

[Snow et al.2008] Rion Snow, Brendan O'Connor, Daniel Jurafsky, and Andrew Y. Ng. 2008. Cheap and fast—but is it good?: evaluating non-expert annotations for natural language tasks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '08, pages 254–263, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Wu and Fung2005] Dekai Wu and Pascale Fung. 2005. Inversion Transduction Grammar Constraints for Mining Parallel Sentences from Quasi-Comparable Corpora. In Robert Dale, Kam-Fai Wong, Jian Su, and Oi Kwong, editors, *Natural Language Processing IJCNLP 2005*, volume 3651 of *Lecture Notes in Computer Science*, pages 257–268. Springer Berlin / Heidelberg.

[Zaidan and Callison-Burch2011] Omar F. Zaidan 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, Stroudsburg, PA, USA. Association for Computational Linguistics.