PBML

The Prague Bulletin of Mathematical Linguistics
NUMBER 99 APRIL 2013 5–16

High-Precision Sentence Alignment by Bootstrapping from Wood Standard Annotations

Éva Mújdricza-Maydt, Huiqin Körkel-Qu, Stefan Riezler, Sebastian Padó

Department of Computational Linguistics, Heidelberg University, Germany

Abstract

We present a semi-supervised, language- and domain-independent approach to high precision sentence alignment. The key idea is to bootstrap a supervised discriminative learner from wood-standard alignments, i.e. alignments that have been automatically generated by state-of-the-art sentence alignment tools. We deploy 3 different unsupervised sentence aligners (Opus, Hunalign, Gargantua) and 2 different datasets (movie subtitles and novels) and show experimentally that bootstrapping consistently improves precision significantly such that, with one exception, we obtain an overall gain in F-score.

1. Introduction

Parallel text is a crucial resource for current approaches to statistical machine translation (Koehn, 2010), statistical models of cross-language information retrieval (Xu et al., 2001; Kraaij et al., 2003; Gao et al., 2006), or other natural language processing tasks that deploy bilingual texts, e.g. monolingual paraphrasing (Bannard and Callison-Burch, 2005).

However, one-to-one sentence parallelism, as found in parliament proceedings, is not the rule but an exception. Most naturally occurring bilingual texts contain roughly corresponding descriptions of the same or overlapping topics. They exhibit parallelism at the level of documents, sentences, or sentence fragments. The challenge to employ such not strictly parallel texts has led to a surge of research on specialized models to extract parallel sentences from sources such as the web (Resnik and Smith, 2003), Wikipedia (Smith et al., 2010), newswire (Munteanu and Marcu, 2005),

© 2013 PBML. All rights reserved.
Corresponding author: riezler@cl.uni-heidelberg.de
Cite as: Éva Mújdricza-Maydt, Huiqin Körkel-Qu, Stefan Riezler, Sebastian Padó. High-Precision Sentence Alignment by Bootstrapping from Wood Standard Annotations . The Prague Bulletin of Mathematical Linguistics No. 99, 2013, pp. 5–16. doi: 10.2478/pralin-2013-0001.

or patents (Utiyama and Isahara, 2007; Lu et al., 2009).

Instead of devising another specialized method for sentence alignment on noisy data, we propose a general two-step model in which a discriminative learner is bootstrapped from data generated by state-of-the-art unsupervised sentence alignment tools. This architecture makes our approach semi-supervised, language- and domain-independent, and applicable to data of various degrees of parallelism. Our bootstrapping methods works as follows: In a first step, we produce large amounts of machine alignments using state-of-the-art sentence aligners. In a second step, we train a discriminative learner on the "wood standard" annotations created in the first step. This combination of arbitrary amounts of machine aligned data and an expressive discriminative learner provides a boost in precision. We evaluate our approach on two datasets: movie subtitles and novels. The efficiency of our approach is ensured by using a moving window of 50 sentence pairs above and below a diagonal of 1-to-1 alignments to break down the huge number of possible alignments (especially in the absence of paragraph breaks, as is the case for movie subtitles). We deploy 3 different unsupervised sentence aligners (Hunalign (Varga et al., 2005), Opus (Tiedemann, 2007), Gargantua (Braune and Fraser, 2010)) for machine labeling. Our experiments show that bootstrapping a discriminative learner significantly improves precision in all cases and, with one exception, also results in an overall gain in F-score.

2. Related work

Most approaches break the sentence alignment problem down into document alignment, e.g. using IR techniques, and a procedure for extracting parallel sentence pairs, e.g., by length-based alignment (Gale and Church, 1993). Typically, sentence pairs are filtered further in a second step on the basis of word alignment scores. These word alignments can be obtained from dictionaries (Utiyama and Isahara, 2007; Lu et al., 2009), external sources (Munteanu and Marcu, 2005; Smith et al., 2010), or from the preliminary sentence pairs obtained in the first step (Braune and Fraser, 2010; Moore, 2002). As an alternative to filtering, word alignments can be integrated as features in a maximum-entropy or CRF model (Munteanu and Marcu, 2005; Smith et al., 2010).

Our approach uses a different kind of two-step approach where a discriminative learner is trained on data that has been machine labeled by state-of-the-art sentence aligners. While our discriminative learner is based on offline computable word-level features, more complex features are hidden in the sentence aligners used in the first step. These may include a word alignment model (e.g., Gargantua (Braune and Fraser, 2010)) or use features that are available only for particular data domains, such as time stamp information (Tiedemann, 2007).

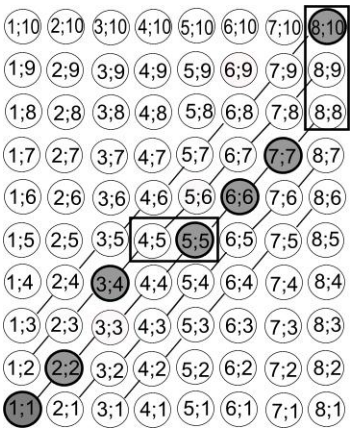

*Figure 1. Example of alignment matrix. Circles represent sentence pairs and are labeled with (source sentence ID; target sentence ID). Bold circles stand for 1:1 gold alignments, bold squares for 1:n and n:1 alignments. Note that target sentence 3 remains unaligned; target sentence 5 participates in a 2:1 alignment (label **S2**); and source sentence 8 takes place in a 1:3 alignment (label **T3**). Filled circles denote model predictions.*

3. Implementation of discriminative sentence alignment

Our toolkit, called *CRFalign*, is implemented in Java and relies heavily on the fast conditional random field sequence classifier *Wapiti* (`wapiti.limsi.fr`) (Lavergne et al., 2010), deploying the expressiveness and flexibility of supervised learning with linear classifiers. Our code is available at `www.cl.uni-heidelberg.de/~mujdricz/software/CRFalign/` and includes the following:

- functions for encoding alignment as sequence labeling problem (see Section 3.1),
- feature functions and interface to *Wapiti* training (see Section 3.2),
- beam search and pruning functions,
- scripts for evaluation and significance testing,
- example corpora and detailed usage instructions.

3.1. Problem encoding

In contrast to simpler sequence labeling problems like part of speech tagging, where we have a sequence of observations to each of which a single label is assigned, the sentence alignment task involves *two* independent sequences, the source and the target corpus. We turn the sentence alignment problem into a set of sequence labeling problems using a *diagonalization* strategy in an alignment matrix, as shown in Figure 1.

This strategy exploits the observation that sentence alignments (like word alignments) tend to follow diagonals in the alignment matrix (Gale and Church, 1993). In other words, the diagonals represent the direction of the most important statistical dependencies: An alignment of sentence pair (n;m) is strong evidence for an alignment at (n+1;m+1). To exploit these dependencies in a linear-chain CRF, we encode each diagonal[1] from an alignment matrix as one sequence labeling problem. For example, one labeling problem would consist of the observation sequence $\langle(1;1),(2;2),\ldots\rangle$ and another one of the observation sequence $\langle(2;1),(3;2),\ldots\rangle$. These sequences capture dependencies between adjacent 1:1 alignments. However, they are unable to directly express the dependencies in 1:n and n:1 alignments, and we express them through labels. We use a set of six labels that express different alignment configurations, defined as follows:

1:1 alignments. If (p;q) is an alignment and no other sentences are aligned with either p or q, the observation (p;q) is labeled with **T**.

2:1 alignments. If (p-1;q) and (p;q) are alignments and no other sentences are aligned with either p or q, then the observation (p;q) is labeled with **S2**.

3:1 alignments. If (p-2;q) (p-1;q) and (p;q) are alignments and no other sentences are aligned with either p or q, then the observation (p;q) is labeled with **S3**.

1:2 alignments. If (p;q-1) and (p;q) are alignments and no other sentences are aligned with either p or q, then the observation (p;q) is labeled with **T2**.

1:3 alignments. If (p;q-2), (p;q-1) and (p;q) are alignments and no other sentences are aligned with either p or q, then the observation (p;q) is labeled with **T3**.

Incomplete alignments and unaligned sentences. All other observations (p;q) are labeled with **F**. This case applies both if p or q are unaligned or if they are part of a larger alignment block.

Therefore, the label sequence in Figure 1 for the observation sequence starting with (1;1) is **T T F F S2 T T F**, and the label sequence for the observation sequence starting with (1;2) is **F F T F F F F F**. This label set is unable to model m:n alignments with $m, n > 1$, or 1:n or n:1 alignments with $n > 3$, but these alignments typically only make up a small fraction of the data, and the cost incurred from introducing more labels exceeds possible benefits.

We apply two optimizations to this process concerning the selection of training and test data to address the predominance of the label **F** in the entirety of alignment matrices. First, we restrict our attention to a subset of all diagonals, exploiting the observation that true alignments usually appear near the first diagonal (the observation sequence starting with (1;1)). Therefore, we only consider diagonals (n;m) where the difference between n and m is smaller than or equal to 50. If the lengths of the texts are very different, we furthermore extend our diagonal set on the axis of the longer

[1] We define a diagonal in a matrix of sentence numbers of two parallel texts as a chain of sentence pairs (*source sentence number*; *target sentence number*) in which the sentence numbers of each next pair are incremented by 1 on both sides.

text. If the source text is x sentences longer than the target text, then we take all diagonals with starting point from $(1;51)$ up to $(51+x;1)$. In the inverse case, we take the diagonals with starting point from $(1;51+x)$ up to $(51;1)$.

The second optimization performs further pruning within this diagonal window: We remove any subsequences of the diagonals that contain more than 15 **F** labels in a row if no other labels follows. This second optimization applies to the training data only, since the test data do not have any access to the labels.

3.2. Features

We use four types of features, all of which are inspired by work on word alignment (Blunsom and Cohn, 2006): (1), length features; (2), position features; (3), similarity features; (4), sequence features. We lift these features to the sentence level. All features, with the exception of the POS agreement feature, are language-independent and can be precomputed from raw text, without the need for linguistic preprocessing.[2] Consequently, our model carries over to other language pairs and to other corpora.

Length ratio. In true alignments, the ratios of source and target sentence lengths can be assumed to be normally distributed (Gale and Church, 1993). The length ratio is an important indicator if a source sentence and a target sentence are a true alignment. We use one feature that captures this intuition. For source and target sentences with m and n words respectively, it is defined as follows:

$$\mathrm{lengthRatio} = \min(\frac{m}{n}, \frac{n}{m}).$$

Position ratio. Sentences that are at similar (relative) positions in source and target files are more likely to be aligned than those which are far away from each other. To describe this characteristic of the sentence alignment, we calculate the position ratio. For source sentences and target sentences at positions p and q, and source and target corpora with s and t sentences, the position ratio is defined as

$$\mathrm{positionRatio} = |\frac{p}{s} - \frac{q}{t}|.$$

POS similarity. Grammatical agreement between sentences can be evidence that they are aligned. This intuition can be operationalized at the part of speech level. For example, if there are two nouns in the source sentence, there is an increased probability to see two nouns in the target sentence. We obtain POS tags from the TreeTagger (Schmid, 1994) and define a simple cross-lingual mapping. The POS agreement

[2] As we will see in the experimental evaluation, the contribution of the POS similarity feature is marginal, thus vindicating our claim of language independence.

feature is defined as the normalized (by sentence length) overlap between the POS tags of a source and target sentence.

Orthographic similarity. The agreement of named entities and internationally used words in a sentence pair is another strong signal for true alignment. We compute an orthographic similarity-based feature which counts matching words in source and target sentence that contain non-lower-case characters, such as *IBM*, *Oct*, or 2.

Punctuation similarity. Similarly, shared punctuation between sentences is also evidence for alignment. Here we only compare the last tokens in the current sentence pairs. If they are matching punctuation marks, we assign a weighted similarity value (by inverse corpus frequency) to the sentence pair.

Word edit similarity. Many language pairs share cognates, that is, words with similar spelling. Our *goodEdit* feature captures this observation by counting the relative frequency of similar words:

$$\mathtt{goodEdit}(p, q) = \frac{2 * \mathtt{goodEditCount}}{wordLen(p) + wordLen(q)}.$$

where `goodEditCount` is defined as the number of word pairs with an edit distance lower than one fifth of their length. This feature can find slightly different spellings of a word in different languages like names such as "Erik" and "Eric", or cognate pairs like "wonder" and "Wunder". This similarity is computed without considering capitalization.

Dice lexical similarity. The Dice coefficient measures the amount of correlated words in two sentences, that is, to what extent the sentences' lexical material co-occurs frequently throughout the corpus. The Dice coefficient for sentences p and q is defined as:

$$\mathrm{Dice}(p, q) = \frac{\sum_i max_j C(p_i, q_j)}{wordLen(p) + wordLen(q)},$$

where the co-occurrence score of two words $C(p_i, q_j)$ is defined as

$$C(p_i, q_j) = \frac{2 * f(p_i, q_j)}{f(p_i) + f(q_j)}.$$

C is highest for word pairs all of whose instances occur in parallel. The Dice coefficient sums the maximum co-occurrence scores for all words in the source sentence that can be obtained by pairing them with the most strongly co-occurring word in the target sentence. The sum is normalized by the lengths of the two sentences. In our computation, we exclude function words from consideration, since they frequently co-occur in unaligned sentences.

Markov sequence feature. Our last feature expresses the first-order Markov dependency between subsequent sentence pairs in a corpus. It is defined to be the label of the preceding pair, i.e. the sequence feature of (n;m) is equal to the label of the preceding pair (n-1;m-1). This Markov feature is crucial for our model, since it captures the regularity that subsequent observations are likely to share the same label.

Feature computation. As described in Section 3, our model assigns six labels to observations that represent different sentence alignment configurations. Obviously, decisions among labels like **T** and **S2** (1:1 vs. 2:1 alignment) require access to features not only of the current observation but also of adjacent observations.

For this reason, the feature set for an observation (p;q) consists not only of the feature values for (p;q) itself, but also of the feature values for the four observations that concern the two previous sentences among the source and target sentences and which can have an impact on the label choice: (p-1;q), (p-2;q), (p;q-1) and (p;q-2). When choosing a label for an observation, the model takes all of these feature "groups" into account. We see that in practice, every feature groups indeed correlates strongly with one label. For example, the model learns that the label **S2** is tightly related with the feature group for (p-1;q).

4. Experiments

4.1. Data

We evaluate our work on two datasets. The first one is Tiedemann's (2009) OpenSubtitles corpus (`opus.lingfil.uu.se/OpenSubtitles_v2.php`) which consists of parallel subtitles extracted from the on-line subtitle provider `www.opensubtitles.org/`. The parallel data contain over a million translations of movie files in 54 languages. For the language pair German-English, there are 3.4 million sentence pairs comprising 42.8 million tokens. At first glance, the alignment of movie subtitles appears to be a simple problem, since subtitle files contain time stamps that indicate the time of the acoustic appearance of each sentence. (Tiedemann, 2007) has used this information to automatically align sentences for the OpenSubtitles corpus. However, this time information is imperfect, and movie subtitles exhibit other kinds of non-parallelism that make alignment more difficult. Non-parallelism arises from insertions (e.g. of scene descriptions), omissions (e.g., due to compression, language differences, or cultural differences), or other complex mappings (e.g., due to subtitling traditions or special application areas such as subtitles for the hearing impaired that need extra information about sounds).

The second dataset is a parallel corpus of 115 19-th century novels and stories in English and German. The novels are part of the Project Gutenberg (`www.gutenberg.org`) (English) and Projekt Gutenberg-DE (`gutenberg.spiegel.de`) (German) and are available from `www.nlpado.de/~sebastian/data/tv_data.shtml`. As the texts are lit-

erary translations, they show a lot of freedom in verbalization. It is also the case that the sentences are on average much longer in novels than in movie subtitles. Sentences in the Gutenberg corpus are on average 25.2 words long. This is more than three times longer than in the OpenSubtitles texts with an average of 7.5 words per sentence. Another source of variability is automatic sentence boundary detection, which leads to typical mistakes in sentence detection which produce $n : m$ alignments ($n, m > 1$). The percentage of such $n : m$ alignments is higher than in the OpenSubtitles corpus.

4.2. Experiment design

For training, *CRFalign* uses *Wapiti* with default meta-parameter settings, except for posterior decoding at labeling time. On the movie subtitle dataset, we use 309 English–German movie pairs from the OpenSubtitles corpus as training set. For evaluation, we manually aligned the German and English subtitles for 6 movies. The annotation guidelines allowed 1:1, 1:n, n:1 and m:n alignments, stating that sentences, or sentence sequences, respectively, should only be aligned if the passages expressed exactly or almost exactly the same content. This guideline was mostly uncontroversial. The train and test data for the Gutenberg dataset consist of 112 novels for training and 3 for testing. The test data set was manually annotated following the OpenSubtitles annotation guidelines.

To evaluate the predictions of our *CRFalign* system for each of the 6 and 3 manually aligned file pairs for OpenSubtitles and Gutenberg data respectively, we compute the standard evaluation measures precision (P), recall (R) and F_1-measure (F). We use the following state-of-the-art sentence alignment tools. On the OpenSubtitles corpus, we deploy the original OpenSubtitles alignments (`opus.lingfil.uu.se/OpenSubtitles_v2.php`) (Tiedemann, 2007) (*OPUS*). This dataset was also sentence-aligned with *Hunalign* (`mokk.bme.hu/resources/hunalign`) (Varga et al., 2005), an unsupervised alignment system that combines length-based alignment with word-alignment filtering. On the Gutenberg corpus, *OPUS* is not available since it relies on time stamp information. Therefore we aligned this data, in addition to *Hunalign*, with *Gargantua* (`sourceforge.net/projects/gargantua/`) (Braune and Fraser, 2010), another state-of-the-art unsupervised sentence alignment tool. Our own system, which we call *CRFalign*, makes its predictions on the basis of these systems for each sentence-pair within the generated diagonals. Recall that for test files we do not prune the diagonals at all, which leads to (a priori) independent predictions for each diagonal. In the case of conflicts (about 5% of predictions), conflicts are resolved by preferring longer alignment chains over shorter ones.

4.3. Experimental results

Table 1 shows a comparative evaluation of state-of-the-art sentence aligners with our discriminative learner trained on machine labeled output of the respective systems.

OpenSubtitles corpus					
OPUS			*Hunalign*		
P	R	F	P	R	F
74.64	73.47	74.05	92.27	91.48	91.87
OPUS + CRFalign			*Hunalign + CRFalign*		
97.59*	85.69*	91.26*	95.51*	87.95*	91.58
Gutenberg corpus					
Gargantua			*Hunalign*		
P	R	F	P	R	F
90.70	89.86	90.28*	74.64	77.76	76.17
Gargantua + CRFalign			*Hunalign + CRFalign*		
91.94	76.64*	83.60*	91.08*	72.22*	80.56*

*Table 1. Precision (P), Recall (R), and F_1-score (F) on OpenSubtitles (top) and Gutenberg (bottom) data for state-of-the-art sentence aligners OPUS, Gargantua, and Hunalign, compared to our CRFalign discriminative sentence aligner trained on the machine labeled output of the respective systems. A statistically significant difference between systems is indicated by * ($p < 0.05$).*

We find that in every single case, precision is significantly improved by bootstrapping the discriminative learner *CRFalign* compared to the original machine-labeled data. Thus, CRFalign consistently acts like a filter that learns to recognize reliable sentence alignment pattern in the output of other aligners. The impact on Recall is more varied: it rises significantly for *OPUS*, drops somewhat for *Hunalign*, and decreases substantially for *Gargantua*. This indicates that the filter is not able to recognize all valid alignment pattern. In the case of Gargantua, F-Score decreases over the initial alignment, however, it increases significantly in most other cases.

We also compared CRFalign against *Hunalign*'s capability to refine its initial alignments in a realignment step (option -realign). *Hunalign*'s realignment results in the following scores (Precision / Recall / F1-measure): 91.14% / 90.73% / 90.93%, and 75.21% / 78.08% / 76.62% on OpenSubtitles and Gutenberg data respectively. Results on OpenSubtitles are slightly lower than the original *Hunalign* alignment scores; the differences on Gutenberg are not statistically significant. These results are lower than the results obtained by realignment via bootstrapping with *CRFalign*.

	OpenSubtitles corpus		Gutenberg corpus	
CRFalign +	*OPUS*	*Hunalign*	*Gargantua*	*Hunalign*
all	92.17	92.22	84.46	79.93
–Markov sequence feature	71.83	60.40	55.07	54.28
–Dice lexical similarity	86.61	86.78	77.98	74.43
–length ratio	88.83	90.11	77.31	76.00
–word edit similarity	91.50	91.85	83.40	80.09
–punctuation similarity	91.81	91.64	83.73	79.48
–position ratio	92.48	92.45	84.04	80.05
–orthographic similarity	91.51	92.49	84.59	80.42
–POS similarity	91.55	92.69	84.75	81.35

Table 2. F-Scores after removing different features the CRFalign feature set.

Furthermore, we investigate the contribution of each feature by ablation, i.e. by leaving out each feature in turn. Table 2 shows the results. Removed features are listed in descending order of their influence on F-score. The analysis shows that the most important individual feature in our feature set is the Markov feature. This feature represents the essential sequential characteristic of our data. Other features contribute to the performance to various, but much smaller, degrees. Most features complement each other so that a cumulative improvement is generally achieved by using all features in the model.

5. Conclusion

This paper has presented an approach sentence alignment that piggybacks on the output of state-of-the-art sentence aligners for bootstrapping a discriminative sentence aligner from machine labeled data. The semi-supervised nature of our approach allows us to aim for high precision alignments while still obtaining improved F-score without the need for manual alignment. Our approach is language- and domain independent and even applicable to datasets with varying degree of parallelism. As shown in the feature ablation experiment, the only language-dependent feature (POS agreement) contributes nearly nothing to overall quality. Our approach addresses the problem of searching a large space of possible sentence alignments by employing a moving window of 50 sentences above and 50 sentences below a diagonal of 1-to-1 alignments. This makes our approach feasible for large datasets even in the absence of paragraph breaks. Finally, the features used in our approach are efficiently computable offline, so that the additional burden of discriminative re-alignment becomes worthwhile if high precision alignments are desired.

Bibliography

Bannard, Colin and Chris Callison-Burch. Paraphrasing with bilingual parallel corpora. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 597–604, Ann Arbor, MI, 2005.

Blunsom, Phil and Trevor Cohn. Discriminative word alignment with conditional random fields. In *Proceedings of the joint conference of the International Committee on Computational Linguistics and the Association for Computational Linguistics (COLING-ACL'06)*, pages 65–72, Sydney, Australia, 2006.

Braune, Fabienne and Alexander Fraser. Improved unsupervised sentence alignment for symmetrical and asymmetrical parallel corpora. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING'10)*, pages 81–89, Beijing, China, 2010.

Gale, William A. and Kenneth W. Church. A program for aligning sentences in bilingual corpora. *Computational Linguistics*, 19(1):75–102, 1993.

Gao, Jianfeng, Jian-Yun Nie, and Ming Zhou. Statistical query translation models for cross language information retrieval. *ACM Transactions on Asian Language Information Processing*, 5(4):323–359, 2006.

Koehn, Philipp. *Statistical Machine Translation*. Cambridge University Press, 2010.

Kraaij, Wessel, Jian-Yun Nie, and Michel Simard. Embedding web-based statistical translation models in cross-language information retrieval. *Computational Linguistics*, 29(3):381–419, 2003.

Lavergne, Thomas, Olivier Chappé, and François Yvon. Practical very large scale CRFs. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL'10)*, pages 504–513, Uppsala, Sweden, 2010.

Lu, Bin, Banjamin K. Tsou, Jingbo Zhu, Tao Jiang, and Oi Yee Kwong. The construction of a chinese-english patent parallel corpus. In *Proceedings of the MT Summit XII*, pages 17–24, Ottawa, Canada, 2009.

Moore, Robert. Fast and accurate sentence alignment of bilingual corpora. In *Proceedings of the 5th Conference of the Association for Machine Translation in the Americas (AMTA'02)*, pages 135–144, Tiburon, CA, 2002.

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

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

Schmid, Helmut. Probabilistic part-of-speech tagging using decision trees. In *Proceedings of the International Conference on New Methods in Language Processing*, Manchester, UK, 1994.

Smith, Jason R., Chris Quirk, and Kristina Toutanova. Extracting parallel sentences from comparable corpora using document level alignment. In *Proceedings of Human Language Technologies: The 11th Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT'10)*, pages 403–411, Los Angeles, CA, 2010.

Tiedemann, Jörg. Improved sentence alignment for movie subtitles. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP'07)*, pages 582–588, Borovets, Bulgaria, 2007.

Tiedemann, Jörg. News from OPUS - a collection of multilingual parallel corpora with tools and interfaces. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP'09)*, pages 1–12, Borovets, Bulgaria, 2009.

Utiyama, Masao and Hitoshi Isahara. A Japanese-English patent parallel corpus. In *Proceedings of MT Summit XI*, pages 475–482, Copenhagen, Denmark, 2007.

Varga, Dániel, László Németh, Péter Halácsy, András Kornai, Viktor Trón, and Viktor Nagy. Parallel corpora for medium density languages. In *Proceedings of the Recent Advances in Natural Language Processing 2005 Conference*, pages 590–596, Borovets, Bulgaria, 2005.

Xu, Jinxi, Ralph Weischedel, and Chanh Nguyen. Evaluating a probabilistic model for cross-lingual information retrieval. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 105–110, New Orleans, LA, 2001.

Address for correspondence:
Stefan Riezler
riezler@cl.uni-heidelberg.de
Department of Computational Linguistics,
Heidelberg University,
Im Neuenheimer Feld 325,
69120 Heidelberg, Germany