Non-Monotonic Sentence Alignment via Semisupervised Learning

Xiaojun Quan, Chunyu Kit and Yan Song
Department of Chinese, Translation and Linguistics
City University of Hong Kong, HKSAR, China
{xiaoquan,ctckit,[yansong]}@[student.]cityu.edu.hk

Abstract

This paper studies the problem of non-monotonic sentence alignment, motivated by the observation that coupled sentences in real bitexts do not necessarily occur monotonically, and proposes a semisupervised learning approach based on two assumptions: (1) sentences with high affinity in one language tend to have their counterparts with similar relatedness in the other; and (2) initial alignment is readily available with existing alignment techniques. They are incorporated as two constraints into a semisupervised learning framework for optimization to produce a globally optimal solution. The evaluation with real-world legal data from a comprehensive legislation corpus shows that while existing alignment algorithms suffer severely from non-monotonicity, this approach can work effectively on both monotonic and non-monotonic data.

1 Introduction

Bilingual sentence alignment is a fundamental task to undertake for the purpose of facilitating many important natural language processing applications such as statistical machine translation (Brown et al., 1993), bilingual lexicography (Klavans et al., 1990), and cross-language information retrieval (Nie et al., 1999). Its objective is to identify correspondences between bilingual sentences in given bitexts. As summarized by Wu (2010), existing sentence alignment techniques rely mainly on sentence length and bilingual lexical resource. Approaches based on the former perform effectively on cognate languages but not on the others. For instance, the statistical correlation of sentence length between English and Chinese is not as high as that between two Indo-European languages (Wu, 1994). Lexicon-based approaches resort to word correspondences in a bilingual lexicon to match bilingual sentences. A few sentence alignment methods and tools have also been explored to combine the two. Moore (2002) proposes a multi-pass search procedure using both sentence length and an automatically-derived bilingual lexicon. Hunalign (Varga et al., 2005) is another sentence aligner that combines sentence length and a lexicon. Without a lexicon, it backs off to a length-based algorithm and then automatically derives a lexicon from the alignment result. Soon after, Ma (2006) develops the lexicon-based aligner Champollion, assuming that different words have different importance in aligning two sentences.

Nevertheless, most existing approaches to sentence alignment follow the monotonicity assumption that coupled sentences in bitexts appear in a similar sequential order in two languages and crossings are not entertained in general (Langlais et al., 1998; Wu, 2010). Consequently the task of sentence alignment becomes handily solvable by means of such basic techniques as dynamic programming. In many scenarios, however, this prerequisite monotonicity cannot be guaranteed. For example, bilingual clauses in legal bitexts are often coordinated in a way not to keep the same clause order, demanding fully or partially crossing pairings. Figure 1 shows a real excerpt from a legislation corpus. Such monotonicity seriously impairs the existing alignment approaches founded on the monotonicity assumption.

This paper is intended to explore the problem of non-monotonic alignment within the framework of semisupervised learning. Our approach is motivated by the above observation and based on the following two assumptions. First, monolingual sentences with high affinity are likely to have their translations with similar relatedness. Following this assumption, we propose the conception of monolingual consistency which, to the best of

Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, pages 622–630,
Sofia, Bulgaria, August 4-9 2013.

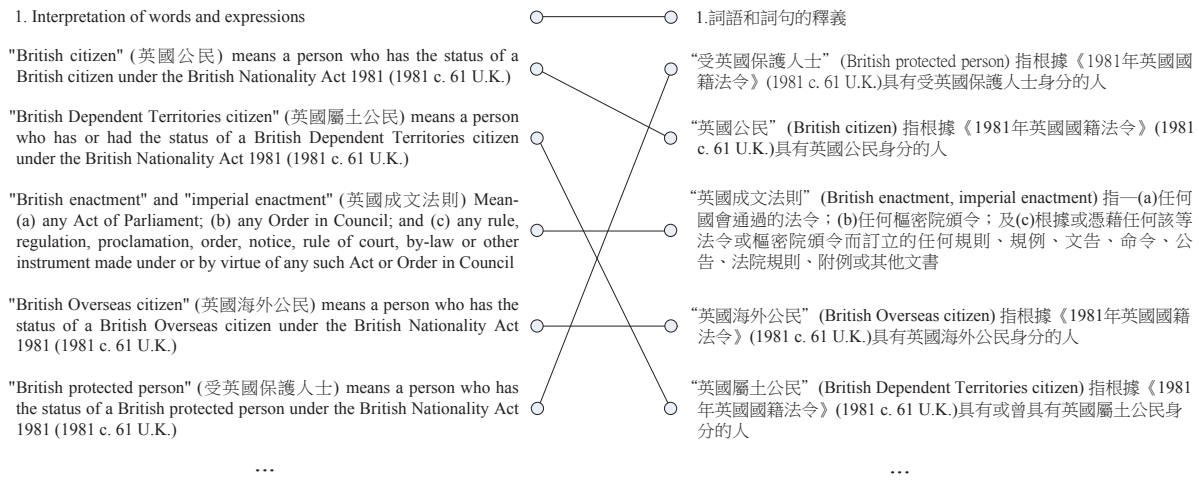

Figure 1: A real example of non-monotonic sentence alignment from BLIS corpus.

our knowledge, has not been taken into account in any previous work of alignment. Second, initial alignment of certain quality can be obtained by means of existing alignment techniques. Our approach attempts to incorporate both monolingual consistency of sentences and bilingual consistency of initial alignment into a semisupervised learning framework to produce an optimal solution. Extensive evaluations are performed using real-world legislation bitexts from BLIS, a comprehensive legislation database maintained by the Department of Justice, HKSAR. Our experimental results show that the proposed method can work effectively while two representatives of existing aligners suffer severely from the non-monotonicity.

2 Methodology

2.1 The Problem

An alignment algorithm accepts as input a bitext consisting of a set of source-language sentences, $\mathcal{S} = \{s_1, s_2, \ldots, s_m\}$, and a set of target-language sentences, $\mathcal{T} = \{t_1, t_2, \ldots, t_n\}$. Different from previous works relying on the monotonicity assumption, our algorithm is generalized to allow the pairings of sentences in $\mathcal{S}$ and $\mathcal{T}$ to cross arbitrarily. Figure 2(a) illustrates monotonic alignment with no crossing correspondences in a bipartite graph and 2(b) non-monotonic alignment with scrambled pairings. Note that it is relatively straightforward to identify the type of many-to-many alignment in monotonic alignment using techniques such as dynamic programming if there is no scrambled pairing or the scrambled pairings are local, limited to a short distance. However, the situation of non-monotonic alignment is much more complicated. Sentences to be merged into a bundle for matching against another bundle in the other language may occur consecutively or discontinuously. For the sake of simplicity, we will not consider non-monotonic alignment with many-to-many pairings but rather assume that each sentence may align to only one or zero sentence in the other language.

Let $\mathcal{F}$ represent the correspondence relation between $\mathcal{S}$ and $\mathcal{T}$, and therefore $\mathcal{F} \subset \mathcal{S} \times \mathcal{T}$. Let matrix F denote a specific alignment solution of $\mathcal{F}$, where F_{ij} is a real score to measure the likelihood of matching the i-th sentence s_i in $\mathcal{S}$ against the j-th sentence t_j in $\mathcal{T}$. We then define an alignment function $\mathcal{A} : F \rightarrow A$ to produce the final alignment, where A is the alignment matrix for $\mathcal{S}$ and $\mathcal{T}$, with $A_{ij} = 1$ for a correspondence between s_i and t_j and $A_{ij} = 0$ otherwise.

2.2 Semisupervised Learning

A semisupervised learning framework is introduced to incorporate the monolingual and bilingual consistency into alignment scoring

$$\mathcal{Q}(F) = \mathcal{Q}_m(F) + \lambda \mathcal{Q}_b(F), \qquad (1)$$

where $\mathcal{Q}_m(F)$ is the term for monolingual constraint to control the consistency of sentences with high affinities, $\mathcal{Q}_b(F)$ for the constraint of initial alignment obtained with existing techniques, and λ is the weight between them. Then, the optimal alignment solution is to be derived by minimizing the cost function $\mathcal{Q}(F)$, i.e.,

$$F^* = \arg\min_F \mathcal{Q}(F). \qquad (2)$$

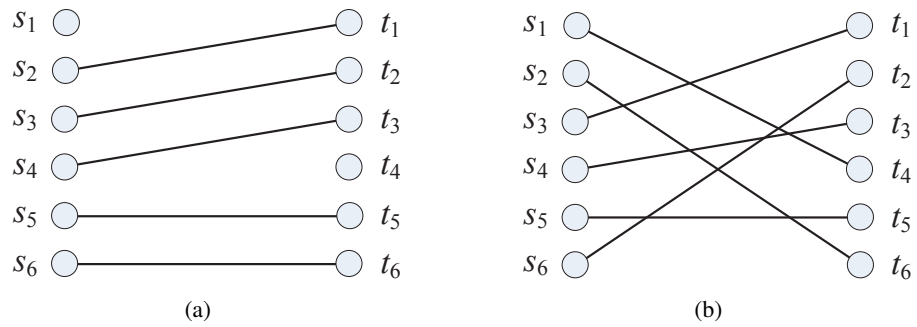

Figure 2: Illustration of monotonic (a) and non-monotonic alignment (b), with a line representing the correspondence of two bilingual sentences.

In this paper, $\mathcal{Q}_m(F)$ is defined as

$$\frac{1}{4}\sum_{i,j=1}^{m} W_{ij}\sum_{k,l=1}^{n} V_{kl}\left(\frac{F_{ik}}{\sqrt{D_{ii}E_{kk}}}-\frac{F_{jl}}{\sqrt{D_{jj}E_{ll}}}\right)^2, \quad (3)$$

where W and V are the symmetric matrices to represent the monolingual sentence affinity matrices in $\mathcal{S}$ and $\mathcal{T}$, respectively, and D and E are the diagonal matrices with entries $D_{ii} = \sum_j W_{ij}$ and $E_{ii} = \sum_j V_{ij}$. The idea behind (3) is that to minimize the cost function, the translations of those monolingual sentences with close relatedness reflected in W and V should also keep similar closeness. The bilingual constraint term $\mathcal{Q}_b(F)$ is defined as

$$\mathcal{Q}_b(F) = \sum_{i=1}^{m}\sum_{j=1}^{n}\left(F_{ij} - \hat{A}_{ij}\right)^2, \quad (4)$$

where $\hat{A}$ is the initial alignment matrix obtained by $\mathcal{A} : \hat{F} \rightarrow \hat{A}$. Note that $\hat{F}$ is the initial relation matrix between $\mathcal{S}$ and $\mathcal{T}$.

The monolingual constraint term $\mathcal{Q}_m(F)$ defined above corresponds to the *smoothness constraint* in the previous semisupervised learning work by Zhou et al. (2004) that assigns higher likelihood to objects with larger similarity to share the same label. On the other hand, $\mathcal{Q}_b(F)$ corresponds to their *fitting constraint*, which requires the final alignment to maintain the maximum consistency with the initial alignment.

Taking the derivative of $\mathcal{Q}(F)$ with respect to F, we have

$$\frac{\partial \mathcal{Q}(F)}{\partial F} = 2F - 2SFT + 2\lambda F - 2\lambda\hat{A}, \quad (5)$$

where S and T are the normalized matrices of W and V, calculated by $S = D^{-1/2}WD^{-1/2}$ and $T = E^{-1/2}VE^{-1/2}$. Then, the optimal F^* is to be found by solving the equation

$$(1+\lambda)F^* - SF^*T = \lambda\hat{A}, \quad (6)$$

which is equivalent to $\alpha F^* - F^*\beta = \gamma$ with $\alpha = (1+\lambda)S^{-1}$, $\beta = T$ and $\gamma = \lambda S^{-1}\hat{A}$. This is in fact a Sylvester equation (Barlow et al., 1992), whose numerical solution can be found by many classical algorithms. In this research, it is solved using LAPACK,[1] a software library for numerical linear algebra. Non-positive entries in F^* indicate unrealistic correspondences of sentences and are thus set to zero before applying the alignment function.

2.3 Alignment Function

Once the optimal F^* is acquired, the remaining task is to design an alignment function $\mathcal{A}$ to convert it into an alignment solution. An intuitive approach is to use a heuristic search for local optimization (Kit et al., 2004), which produces an alignment with respect to the largest scores in each row and each column. However, this does not guarantee a globally optimal solution. Figure 3 illustrates a mapping relation matrix onto an alignment matrix, which also shows that the optimal alignment cannot be achieved by heuristic search.

Banding is another approach frequently used to convert a relation matrix to alignment (Kay and Röscheisen, 1993). It is founded on the observation that true monotonic alignment paths usually lie close to the diagonal of a relation matrix. However, it is not applicable to our task due to the non-monotonicity involved. We opt for converting a relation matrix into specific alignment by solving

[1] http://www.netlib.org/lapack/

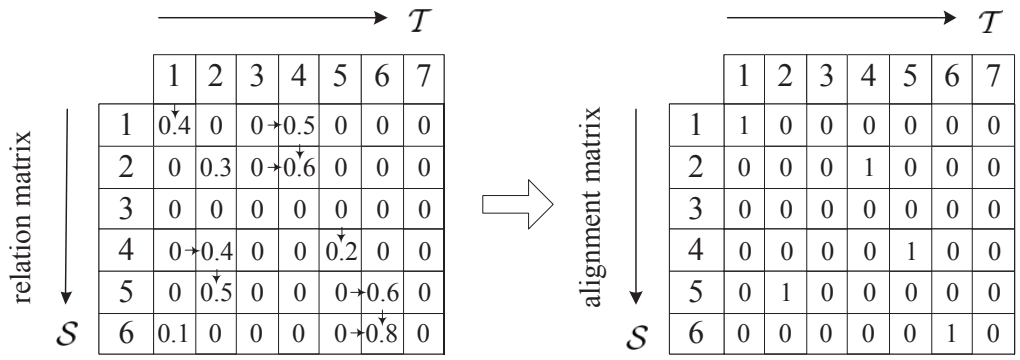

Figure 3: Illustration of sentence alignment from relation matrix to alignment matrix. The scores marked with arrows are the best in each row/column to be used by the heuristic search. The right matrix represents the corresponding alignment matrix by our algorithm.

the following optimization

$$A = \arg\max_{X} \sum_{i=1}^{m} \sum_{j=1}^{n} X_{ij} F_{ij} \tag{7}$$

$$s.t. \quad \sum_{i=1}^{m} X_{ij} \leq 1, \sum_{j=1}^{n} X_{ij} \leq 1, X_{ij} \in \{0, 1\}$$

This turns sentence alignment into a problem to be resolved by binary linear programming (BIP), which has been successfully applied to word alignment (Taskar et al., 2005). Given a scoring matrix, it guarantees an optimal solution.

2.4 Alignment Initialization

Once the above alignment function is available, the initial alignment matrix $\hat{A}$ can be derived from an initial relation matrix $\hat{F}$ obtained by an available alignment method. This work resorts to another approach to initializing the relation matrix. In many genres of bitexts, such as government transcripts or legal documents, there are a certain number of common strings on the two sides of bitexts. In legal documents, for example, translations of many key terms are usually accompanied with their source terms. Also, common numberings can be found in enumerated lists in bitexts. These kinds of *anchor strings* provide quite reliable information to link bilingual sentences into pairs, and thus can serve as useful cues for sentence alignment. In fact, they can be treated as a special type of highly reliable "bilexicon".

The anchor strings used in this work are derived by searching the bitexts using word-level inverted indexing, a basic technique widely used in information retrieval (Baeza-Yates and Ribeiro-Neto, 2011). For each index term, a list of postings is created. Each posting includes a sentence identifier, the in-sentence frequency and positions of this term. The positions of terms are intersected to find common anchor strings. The anchor strings, once found, are used to calculate the initial affinity $\hat{F}_{ij}$ of two sentences using Dice's coefficient

$$\hat{F}_{ij} = \frac{2|\mathcal{C}_{1i} \cap \mathcal{C}_{2j}|}{|\mathcal{C}_{1i}| + |\mathcal{C}_{2j}|} \tag{8}$$

where $\mathcal{C}_{1i}$ and $\mathcal{C}_{2j}$ are the anchor sets in s_i and t_j, respectively, and $|\cdot|$ is the cardinality of a set.

Apart from using anchor strings, other avenues for the initialization are studied in the evaluation section below, i.e., using another aligner and an existing lexicon.

2.5 Monolingual Affinity

Although various kinds of information from a monolingual corpus have been exploited to boost statistical machine translation models (Liu et al., 2010; Su et al., 2012), we have not yet been exposed to any attempt to leverage monolingual sentence affinity for sentence alignment. In our framework, an attempt to this can be made through the computation of W and V. Let us take W as an example, where the entry W_{ij} represents the affinity of sentence s_i and sentence s_j, and it is set to 0 for $i = j$ in order to avoid self-reinforcement during optimization (Zhou et al., 2004).

When two sentences in $\mathcal{S}$ or $\mathcal{T}$ are not too short, or their content is not divergent in meaning, their semantic similarity can be estimated in terms of common words. Motivated by this, we define W_{ij} (for $i \neq j$) based on the Gaussian kernel as

$$W_{ij} = \exp\left(-\frac{1}{2\sigma^2}\left(1 - \frac{v_i^T v_j}{\|v_i\| \, \|v_j\|}\right)^2\right) \tag{9}$$

where σ is the standard deviation parameter, v_i and v_j are vectors of s_i and s_j with each component corresponding to the *tf-idf* value of a particular term in $\mathcal{S}$ (or $\mathcal{T}$), and $\|\cdot\|$ is the norm of a vector. The underlying assumption here is that words appearing frequently in a small number of sentences but rarely in the others are more significant in measuring sentence affinity.

Although semantic similarity estimation is a straightforward approach to deriving the two affinity matrices, other approaches are also feasible. An alternative approach can be based on sentence length under the assumption that two sentences with close lengths in one language tend to have their translations also with close lengths.

2.6 Discussion

The proposed semisupervised framework for non-monotonic alignment is in fact generalized beyond, and can also be applied to, monotonic alignment. Towards this, we need to make use of sentence sequence information. One way to do it is to incorporate sentence positions into Equation (1) by introducing a position constraint $\mathcal{Q}_p(F)$ to enforce that bilingual sentences in closer positions should have a higher chance to match one another. For example, the new constraint can be defined as

$$\mathcal{Q}_p(F) = \sum_{i=1}^{m}\sum_{j=1}^{n} |p_i - q_j| F_{ij}^2,$$

where p_i and q_j are the absolute (or relative) positions of two bilingual sentences in their respective sequences. Another way follows the banding assumption that the actual couplings only appear in a narrow band along the main diagonal of relation matrix. Accordingly, all entries of F^* outside this band are set to zero before the alignment function is applied. Kay and Röscheisen (1993) illustrate that this can be done by modeling the maximum deviation of true couplings from the diagonal as $O(\sqrt{n})$.

3 Evaluation

3.1 Data Set

Our data set is acquired from the Bilingual Laws Information System (BLIS),[2] an electronic database of Hong Kong legislation maintained by the Department of Justice, HKSAR. BLIS provides Chinese-English bilingual texts of ordinances and subsidiary legislation in effect on or after 30 June 1997. It organizes the legal texts into a hierarchy of chapters, sections, subsections, paragraphs and subparagraphs, and displays the content of a such hierarchical construct (usually a section) on a single web page.

By web crawling, we have collected in total 31,516 English and 31,405 Chinese web pages, forming a bilingual corpus of 31,401 bitexts after filtering out null pages. A text contains several to two hundred sentences. Many bitexts exhibit partially non-monotonic order of sentences. Among them, 175 bitexts are randomly selected for manual alignment. Sentences are identified based on punctuations. OpenNLP Tokenizer[3] is applied to segment English sentences into tokens. For Chinese, since there is no reliable segmenter for this genre of text, we have to treat each Chinese character as a single token. In addition, to calculate the monolingual sentence affinity, stemming of English words is performed with the Porter Stemmer (Porter, 1980) after anchor string mining.

The manual alignment of the evaluation data set is performed upon the initial alignment by Hunalign (Varga et al., 2005), an effective sentence aligner that uses both sentence length and a bilexicon (if available). For this work, Hunalign relies solely on sentence length. Its output is then double-checked and corrected by two experts in bilingual studies, resulting in a data set of 1747 1-1 and 70 1-0 or 0-1 sentence pairs.

The standard deviation σ in (9) is an important parameter for the Gaussian kernel that has to be determined empirically (Zhu et al., 2003; Zhou et al., 2004). In addition, the $\mathcal{Q}$ function also involves another parameter λ to adjust the weight of the bilingual constraint. This work seeks an approach to deriving the optimal parameters without any external training data beyond the initial alignment. A three-fold cross-validation is thus performed on the initial 1-1 alignment and the parameters that give the best average performance are chosen.

3.2 Monolingual Consistency

To demonstrate the validity of the monolingual consistency, the semantic similarity defined by $\frac{v_i^T v_j}{\|v_i\|\|v_j\|}$ is evaluated as follows. 500 pairs of English sentences with the highest similarities are selected, excluding *null* pairings (1-0 or 0-1 type).

[2] http://www.legislation.gov.hk

[3] http://opennlp.apache.org/

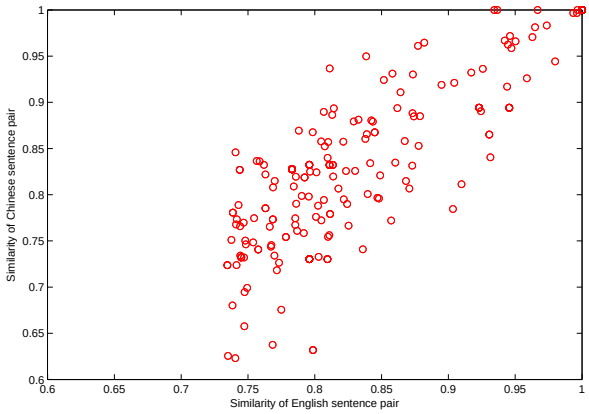

Figure 4: Demonstration of monolingual consistency. The horizontal axis is the similarity of English sentence pairs and the vertical is the similarity of the corresponding pairs in Chinese.

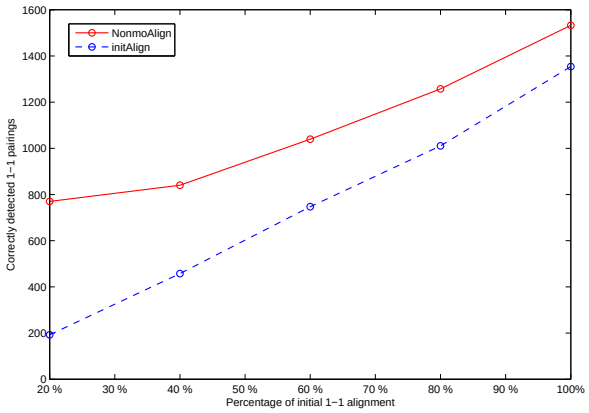

Figure 5: Performance of non-monotonic alignment along the percentage of initial 1-1 alignment.

Type	Total	initAlign		NonmoAlign	
		Pred	*Corr*	*Pred*	*Corr*
1-0	70	662	66	70	50
1-1	1747	1451	1354	1747	1533

Table 1: Performance of the initial alignment and our aligner, where the *Pred* and *Corr* columns are the numbers of predicted and correct pairings.

All of these high-affinity pairs have a similarity score higher than 0.72. A number of duplicate sentences (e.g., date) with exceptionally high similarity 1.0 are dropped. Also, the similarity of the corresponding translations of each selected pair is calculated. These two sets of similarity scores are then plotted in a scatter plot, as in Figure 4. If the monolingual consistency assumption holds, the plotted points would appear nearby the diagonal. Figure 4 confirms this, indicating that sentence pairs with high affinity in one language do have their counterparts with similarly high affinity in the other language.

3.3 Impact of Initial Alignment

The 1-1 initial alignment plays the role of labeled instances for the semisupervised learning. It is of critical importance to the learning performance. As shown in Table 1, our alignment function predicts 1451 1-1 pairings by virtue of anchor strings, among which 1354 pairings are correct, yielding a relatively high precision in the non-monotonic circumstance. It also predicts *null* alignment for many sentences that contain no anchor. This explains why it outputs 662 1-0 pairings when there are only 70 1-0 true ones. Starting from this initial alignment, our aligner (let us call it *NonmoAlign*) discovers 179 more 1-1 pairings.

A question here is concerned with how the scale of initial alignment affects the final alignment. To examine this, we randomly select 20%, 40%, 60% and 80% of the 1451 1-1 detected pairings as the initial alignments for a series of experiments. The random selection for each proportion is performed ten times and their average alignment performance is taken as the final result and plotted in Figure 5. An observation from this figure is that the aligner consistently discovers significantly more 1-1 pairings on top of an initial 1-1 alignment, which has to be accounted for by the monolingual consistency. Another observation is that the alignment performance goes up along the increase of the percentage of initial alignment while performance gain slows down gradually. When the percentage is very low, the aligner still works quite effectively.

3.4 Non-Monotonic Alignment

To test our aligner with non-monotonic sequences of sentences, we have them randomly scrambled in our experimental data. This undoubtedly increases the difficulty of sentence alignment, especially for the traditional approaches critically relying on monotonicity.

The baseline methods used for comparison are Moore's aligner (Moore, 2002) and Hunalign (Varga et al., 2005). Hunalign is configured with the option [-realign], which triggers a three-step procedure: after an initial alignment, Hunalign heuristically enriches its dictionary using word co-occurrences in identified sentence pairs; then, it re-runs the alignment process using the updated

Type	Moore			Hunalign			NonmoAlign		
	P	R	F_1	P	R	F_1	P	R	F_1
1-1	0.104	0.104	0.104	0.407	0.229	0.293	0.878	0.878	0.878
1-0	0.288	0.243	0.264	0.033	0.671	0.062	0.714	0.714	0.714
Micro	0.110	0.110	0.110	0.184	0.246	0.210	0.871	0.871	0.871

Table 2: Performance comparison with the baseline methods.

dictionary. According to Varga et al (2005), this setting gives a higher alignment quality than otherwise. In addition, Hunalign can use an external bilexicon. For a fair comparison, the identified anchor set is fed to Hunalign as a special bilexicon. The performance of alignment is measured by precision (P), recall (R) and F-measure (F_1). Micro-averaged performance scores of precision, recall and F-measure are also computed to measure the overall performance on 1-1 and 1-0 alignment. The final results are presented in Table 2, showing that both Moore's aligner and Hunalign underperform ours on non-monotonic alignment. The particularly poor performance of Moore's aligner has to be accounted for by its requirement of more than thousands of sentences in bitext input for reliable estimation of its parameters. Unfortunately, our available data has not reached that scale yet.

3.5 Partially Non-Monotonic Alignment

Full non-monotonic bitexts are rare in practice. But partial non-monotonic ones are not. Unlike traditional alignment approaches, ours does not found its performance on the degree of monotonicity. To test this, we construct five new versions of the data set for a series of experiments by randomly choosing and scrambling 0%, 10%, 20%, 40%, 60% and 80% sentence parings. In theory, partial non-monotonicity of various degrees should have no impact on the performance of our aligner. It is thus not surprised that it achieves the same result as reported in last subsection. NonmoAlign initialized with Hunalign (marked as NonmoAlign_Hun) is also tested. The experimental results are presented in Figure 6. It shows that both Moore's aligner and Hunalign work relatively well on bitexts with a low degree of non-monotonicity, but their performance drops dramatically when the non-monotonicity is increased. Despite the improvement at low non-monotonicity by seeding our aligner with Hunalign, its performance decreases likewise when the degree of non-monotonicity increases, due to the quality decrease of the initial alignment by Hunalign.

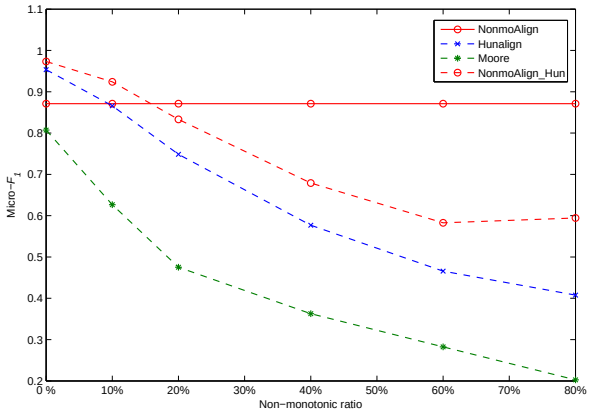

Figure 6: Performance of alignment approaches at different degrees of non-monotonicity.

3.6 Monotonic Alignment

The proposed alignment approach is also expected to work well on monotonic sentence alignment. An evaluation is conducted for this using a monotonic data set constructed from our data set by discarding all its 126 crossed pairings. Of the two strategies discussed above, banding is used to help our aligner incorporate the sequence information. The initial relation matrix is built with the aid of a dictionary automatically derived by Hunalign. Entries of the matrix are derived by employing a similar strategy as in Varga et al. (2005). The evaluation results are presented in Table 3, which shows that NonmoAlign still achieves very competitive performance on monotonic sentence alignment.

4 Related Work

The research of sentence alignment originates in the early 1990s. Gale and Church (1991) and Brown (1991) report the early works using length statistics of bilingual sentences. The general idea is that the closer two sentences are in length, the more likely they are to align. A notable difference of their methods is that the former uses sentence

Type	Moore			Hunalign			NonmoAlign		
	P	*R*	F_1	*P*	*R*	F_1	*P*	*R*	F_1
1-1	0.827	0.828	0.827	0.999	0.972	0.986	0.987	0.987	0.987
1-0	0.359	0.329	0.343	0.330	0.457	0.383	0.729	0.729	0.729
Micro	0.809	0.807	0.808	0.961	0.951	0.956	0.976	0.976	0.976

Table 3: Performance of monotonic alignment in comparison with the baseline methods.

length in number of characters while the latter in number of tokens. Both use dynamic programming to search for the best alignment. As shown in Chen (1993) and Wu (1994), however, sentence-length based methods suffer when the texts to be aligned contain small passages, or the languages involved share few cognates. The subsequent stage of sentence alignment research is accompanied by the advent of a handful of well-designed alignment tools. Moore (2002) proposes a three-pass procedure to find final alignment. Its bitext input is initially aligned based on sentence length. This step generates a set of strictly-selected sentence pairs for use to train an IBM translation model 1 (Brown et al., 1993). Its final step realigns the bitext using both sentence length and the discovered word correspondences. Hunalign (Varga et al., 2005), originally proposed as an ingredient for building parallel corpora, has demonstrated an outstanding performance on sentence alignment. Like many other aligners, it employs a similar strategy of combining sentence length and lexical data. In the absence of a lexicon, it first performs an initial alignment wholly relying on sentence length and then automatically builds a lexicon based on this alignment. Using an available lexicon, it produces a rough translation of the source text by converting each token to the one of its possible counterparts that has the highest frequency in the target corpus. Then, the relation matrix of a bitext is built of similarity scores for the rough translation and the actual translation at sentence level. The similarity of two sentences is calculated in terms of their common pairs and length ratio.

To deal with noisy input, Ma (2006) proposes a lexicon-based sentence aligner - Champollion. Its distinctive feature is that it assigns different weights to words in terms of their *tf-idf* scores, assuming that words with low sentence frequencies in a text but high occurrences in some local sentences are more indicative of alignment. Under this assumption, the similarity of any two sentences is calculated accordingly and then a dynamic programming algorithm is applied to produce final alignment. Following this work, Li et al. (2010) propose a revised version of Champollion, attempting to improve its speed without performance loss. For this purpose, the input bitexts are first divided into smaller aligned fragments before applying Champollion to derive finer-grained sentence pairs. In another related work by Deng et al. (2007), a generative model is proposed, accompanied by two specific alignment strategies, i.e., dynamic programming and divisive clustering. Although a non-monotonic search process that tolerates two successive chunks in reverse order is involved, their work is essentially targeted at monotonic alignment.

5 Conclusion

In this paper we have proposed and tested a semisupervised learning approach to non-monotonic sentence alignment by incorporating both monolingual and bilingual consistency. The utility of monolingual consistency in maintaining the consonance of high-affinity monolingual sentences with their translations has been demonstrated. This work also exhibits that bilingual consistency of initial alignment of certain quality is useful to boost alignment performance. Our evaluation using real-world data from a legislation corpus shows that the proposed approach outperforms the baseline methods significantly when the bitext input is composed of non-monotonic sentences. Working on partially non-monotonic data, this approach also demonstrates a superior performance. Although initially proposed for non-monotonic alignment, it works well on monotonic alignment by incorporating the constraint of sentence sequence.

Acknowledgments

The research described in this paper was substantially supported by the Research Grants Council (RGC) of Hong Kong SAR, China, through the GRF grant 9041597 (CityU 144410).

References

Ricardo Baeza-Yates and Berthier Ribeiro-Neto. 2011. *Modern Information Retrieval: The Concepts and Technology Behind Search*, 2nd ed., Harlow: Addison-Wesley.

Jewel B. Barlow, Moghen M. Monahemi, and Dianne P. O'Leary. 1992. Constrained matrix Sylvester equations. In *SIAM Journal on Matrix Analysis and Applications*, 13(1):1-9.

Peter F. Brown, Jennifer C. Lai, Robert L. Mercer. 1991. Aligning sentences in parallel corpora. In *Proceedings of ACL'91*, pages 169-176.

Peter F. Brown, Vincent J. Della Pietra, Stephen A. Della Pietra and Robert L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263-311.

Stanley F. Chen. 1993. Aligning sentences in bilingual corpora using lexical information. In *Proceedings of ACL'93*, pages 9-16.

Yonggang Deng, Shankar Kumar, and William Byrne. 2007. Segmentation and alignment of parallel text for statistical machine translation. *Natural Language Engineering*, 13(3): 235-260.

William A. Gale, Kenneth Ward Church. 1991. A Program for aligning sentences in bilingual corpora. In *Proceedings of ACL'91*, pages 177-184.

Martin Kay and Martin Röscheisen. 1993. Text-translation alignment. *Computational Linguistics*, 19(1):121-142.

Chunyu Kit, Jonathan J. Webster, King Kui Sin, Haihua Pan, and Heng Li. 2004. Clause alignment for bilingual HK legal texts: A lexical-based approach. *International Journal of Corpus Linguistics*, 9(1):29-51.

Chunyu Kit, Xiaoyue Liu, King Kui Sin, and Jonathan J. Webster. 2005. Harvesting the bitexts of the laws of Hong Kong from the Web. In *The 5th Workshop on Asian Language Resources*, pages 71-78.

Judith L. Klavans and Evelyne Tzoukermann. 1990. The bicord system: Combining lexical information from bilingual corpora and machine readable dictionaries. In *Proceedings of COLING'90*, pages 174-179.

Philippe Langlais, Michel Simard, and Jean Véronis. 1998. Methods and practical issues in evaluating alignment techniques. In *Proceedings of COLING-ACL'98*, pages 711-717.

Zhanyi Liu, Haifeng Wang, Hua Wu, and Sheng Li. 2010. Improving statistical machine translation with monolingual collocation. In *Proceedings of ACL 2010*, pages 825-833.

Xiaoyi Ma. 2006. Champollion: A robust parallel text sentence aligner. In *LREC 2006*, pages 489-492.

Peng Li, Maosong Sun, Ping Xue. 2010. Fast-Champollion: a fast and robust sentence alignment algorithm. In *Proceedings of ACL 2010: Posters*, pages 710-718.

Robert C. Moore. 2002. Fast and accurate sentence alignment of bilingual corpora. In *Proceedings of AMTA 2002*, page 135-144.

Jian-Yun Nie, Michel Simard, Pierre Isabelle and Richard Durand. 1999. Cross-language information retrieval based on parallel texts and automatic mining of parallel texts from the Web. In *Proceedings of SIGIR'99*, pages 74-81.

Martin F. Porter. 1980. An algorithm for suffix stripping. *Program*, 14(3): 130-137.

Jinsong Su, Hua Wu, Haifeng Wang, Yidong Chen, Xiaodong Shi, Huailin Dong, Qun Liu. 2012. Translation model adaptation for statistical machine translation with monolingual topic information. In *Proceedings of ACL 2012*, Vol. 1, pages 459-468.

Ben Taskar, Simon Lacoste-Julien and Dan Klein. 2005. A discriminative matching approach to word alignment. In *Proceedings of HLT/EMNLP 2005*, pages 73-80.

Dániel Varga, Péter Halácsy, András Kornai, Viktor Nagy, László Németh, Viktor Trón. 2005. Parallel corpora for medium density languages. In *Proceedings of RANLP 2005*, pages 590-596.

Dekai Wu. 1994. Aligning a parallel English-Chinese corpus statistically with lexical criteria. In *Proceedings of ACL'94*, pages 80-87.

Dekai Wu. 2010. Alignment. *Handbook of Natural Language Processing*, 2nd ed., CRC Press.

Dengyong Zhou, Olivier Bousquet, Thomas N. Lal, Jason Weston, Bernhard Schlkopf. 2004. Learning with local and global consistency. *Advances in Neural Information Processing Systems*, 16:321-328.

Xiaojin Zhu, Zoubin Ghahramani and John Lafferty. 2003. Semi-supervised learning using Gaussian fields and harmonic functions. In *Proceedings of ICML 2003*, pages 912-919.