Application of Clause Alignment for Statistical Machine Translation

Svetla Koeva, Borislav Rizov, Ivelina Stoyanova, Svetlozara Leseva, Rositsa Dekova, Angel Genov, Ekaterina Tarpomanova, Tsvetana Dimitrova and **Hristina Kukova**

Department of Computational Linguistics
Institute for Bulgarian Language, Bulgarian Academy of Sciences
Sofia 1113, Bulgaria

{svetla,boby,iva,zarka,rosdek,angel,katja,cvetana,hristina}@dcl.bas.bg

Abstract

The paper presents a new resource light flexible method for clause alignment which combines the Gale-Church algorithm with internally collected textual information. The method does not resort to any pre-developed linguistic resources which makes it very appropriate for resource light clause alignment. We experiment with a combination of the method with the original Gale-Church algorithm (1993) applied for clause alignment. The performance of this flexible method, as it will be referred to hereafter, is measured over a specially designed test corpus.

The clause alignment is explored as means to provide improved training data for the purposes of Statistical Machine Translation (SMT). A series of experiments with Moses demonstrate ways to modify the parallel resource and effects on translation quality: (1) baseline training with a Bulgarian-English parallel corpus aligned at sentence level; (2) training based on parallel clause pairs; (3) training with clause reordering, where clauses in each source language (SL) sentence are reordered according to order of the clauses in the target language (TL) sentence. Evaluation is based on BLEU score and shows small improvement when using the clause aligned corpus.

1 Motivation

Evaluation on the performance of MT systems has shown that a pervasive shortcoming shared by both the phrase-based and the syntax-based SMT systems is translating long and (syntactically) complex sentences (Koehn et al., 2003; Li et al., 2007; Sudoh et al., 2010).

The power of phrase-based SMT lies in local lexical choice and short-distance reordering (Li et al., 2007). Syntax-based SMT is better suited to cope with long-distance dependencies, however there also are problems, some of them originated from the linguistic motivation itself – incorrect parse-trees, or reordering that might involve blocks that are not constituents (Li et al., 2007).

An efficient way to overcome the problem of sentence length and complexity is to process the clauses in a similar way as sentences. This has incited growing interest towards the alignment and processing of clauses – a group of syntactically and semantically related words expressing predicative relation and positioned between sentence borders or clause connectors. (It is known that some predicative relations can be considered complex being saturated with another predicative relation – but with the above given definition this case is simplified).

The differences in word order and phrase structure across languages can be better captured at a clause rather than at a sentence level, therefore, monolingual and parallel text processing in the scope of the clauses may significantly improve syntactic parsing, automatic translation, etc. The sentences can be very long and complex in structure, may consist of a considerable number of clauses which in turn may vary with respect to their relative position to each other in parallel texts both due to linguistic reasons per se and translators' choices.

The flexible order, length and number of clauses

Proceedings of SSST-6, Sixth Workshop on Syntax, Semantics and Structure in Statistical Translation, pages 102–110,
Jeju, Republic of Korea, 12 July 2012.

in sentences, along with the different word order and ways of lexicalisation across languages contribute to the complexity of clause alignment as compared to sentence alignment and call for more sophisticated approaches. These findings have inspired growing research into clause-to-clause machine translation involving clause splitting, alignment and word order restructuring within the clauses (Cowan et al., 2006; Ramanathan et al., 2011; Sudoh et al., 2010; Goh et al., 2011).

A fixed clause order in a language (i.e. relative clauses in Bulgarian, English, French and many other languages follow the head noun, while in Chinese, Japanese, Turkish, etc. they precede it) may correspond to a free order in another (i.e. Bulgarian and English adverbial clauses). The hypothesis is that a SMT model can be improved by inducing a straightforward clause alignment through reordering the clauses of the source language text so as to correspond to the order of the clauses in the target language text.

2 State-of-the-art

The task of clause alignment is closely related to that of sentence alignment (Brown et al., 1990; Gale and Church, 1993; Kay and Roscheisen, 1993) and phrase alignment (DeNero and Klein, 2008; Koehn et al., 2003). There are two main approaches – statistical and lexical, often employed together to produce hybrid methods. Machine learning techniques are applied to extract models from the data and reduce the need of predefined linguistic resources.

Boutsis, Piperidis and others (Boutsis and Piperidis, 1998; Boutsis and Piperidis, 1998; Piperidis et al., 2000) employ a method combining statistical techniques and shallow linguistic processing applied on a bilingual parallel corpus of software documentation which is sentence-aligned, POS-tagged and shallow parsed. The combined task of clause borders identification uses linguistic information (POS tagging and shallow parsing) and clause alignment based on pure statistical analysis. The reported precision is 85.7%. Kit et al. (2004) propose a method for aligning clauses in Hong Kong legal texts to English which relies on linguistic information derived from a glossary of bilingual legal terms and a large-scale bilingual dictionary. The algorithm selects a minimal optimal set of scores in the similarity matrix that covers all clauses in both languages. The authors report 94.60% alignment accuracy of the clauses, corresponding to 88.64% of the words.

The quality of the parallel resources is of crucial importance to the performance of SMT systems and substantial research is focused on developing good parallel corpora of high standard. Most clause alignment methods are applied on domain specific corpora, in particular administrative corpora and are not extensively tested and evaluated on general corpora or on texts of other domains. Although clause segmentation is often performed together with clause alignment (Papageorgiou, 1997) the former tends to be more language-specific and therefore clause alignment is performed and evaluated independently. The majority of the available comparative analyses discuss modifications of one method rather than the performance of different methods. Moreover, the performance of resource-free against resource-rich methods has been poorly explored. To the best of our knowledge, there is no purely resource-free method for clause alignment offered so far.

In recent years, handling machine translation at the clause level has been found to overcome some of the limitations of phrase-based SMT. Clause aligned corpora have been successfully employed in the training of models for clause-to-clause translation, reordering and subsequent sentence reconstruction in SMT – Cowan et al. (2006) for syntax-based German-to-English SMT, Sudoh et al. (2010) for English-to-Japanese phrase-based SMT, among others.

Cowan et al. (2006) discuss an approach for tree-to-tree SMT using Tree Adjoining Grammars. Clause alignment is performed on a corpus (Europarl) which is then used in the training of a model for mapping parse trees in the source language to parse trees in the target language. The performance of this syntax-based method is similar to the phrase-based model of Koehn et al. (2003).

Sudoh et al. (2010) propose a method for clause-to-clause translation by means of a standard SMT method. The clauses may contain non-terminals as placeholders for embedded clauses. After translation is performed, the non-terminals are replaced

by their clause translations. The model for clause translation is trained using a clause-aligned bilingual corpus of research paper abstract. The proposed improvement by using Moses is 1.4% in BLEU (33.19% to 34.60%), and 1.3% in TER (57.83% to 56.50%) and 2.2% in BLEU (32.39% to 34.55%) and 3.5% in TER (58.36% to 54.87%) using a hierarchical phrase-based SMT system.

The potential of clause alignment along with other sub-sentence levels of alignment in extracting matching translation equivalents from translation archives has been recognised within the EBMT framework, as well (Piperidis et al., 2000).

3 Bootstrapping clause alignment

The clause alignment is modelled as a bipartite graph. Each node in the graph corresponds to a clause in either the source or the target language. A pair of clauses that are fully or partially translational equivalents is connected by an edge in the graph. The connected components of the graph are beads (the smallest group of aligned clauses). In these terms, the task of clause alignment is the task of the identification of the edges in a bipartite graph, where the nodes are the clauses (Brown et al., 1990).

A bootstrapping method for clause alignment that does not exploit any pre-developed linguistic resources is elaborated. The method uses length-balance based alignment algorithm – i.e. Gale-Church (Gale and Church, 1993), for the data collecting. The bootstrapping algorithm attains high precision and relatively good recall. In order to improve the recall while preserving the precision the method is combined with the Gale-Church algorithm applied to clause alignment.

The proposed method consists of the following stages:

1. Initial clause alignment that serves as training data.
2. Identifying similarities between clauses in different languages.
3. Building the clause alignment.

3.1 The Gale and Church algorithm

Gale and Church (1993) describe a method for aligning sentences based on a simple statistical model of sentence lengths measured in number of characters. It relies on the fact that longer sentences in one language tend to be translated into longer sentences in the other language, and vice versa. A probabilistic score is assigned to each proposed correspondence of sentences, based on the scaled difference and the variance of the lengths of the two sentences. The method is reported to give less than 4% error in terms of alignment and is probably the most widely used sentence alignment method.

The extended version of the Gale-Church aligner from the Natural Language Toolkit[1] is applied for clause alignment. The original Gale-Church method applies the 1:1, 0:1, 1:0, 1:2, 2:1 and 2:2 bead models; in the extended version – the 1:3, 3:1, 2:3, 3:2, 3:3 models are added.

3.2 Clause alignment training data

The clause beads are identified by applying the Gale-Church algorithm. The aim is to select a set of aligned beads which are to serve as a training set for the subsequent stages. Only beads showing high probability of correctness are used. For any probability p we could find δ so that for the Gale-Church measure within $[-\delta, \delta]$ the corresponding bead is correct with probability p.

3.3 Clause similarity

Clause similarity is measured by means of: a) partial word alignment, b) length similarity, and c) weighted punctuation similarity.

3.3.1 Word alignment

To align words in the scope of parallel clauses, word-to-word connections (weighted links between two words based on word similarity) are calculated using several methods given below:

- Vector space model

  A given word is assigned a vector

  $$< x_1,\ x_2,\ \cdots,\ x_n >$$

  in an n-dimensional vector space, where each dimension represents a bead in the preliminary clause alignment and x_i is the number of the occurrences of the word in the bead. The set of these vectors is a matrix.

[1] http://nltk.googlecode.com

The vector space word similarity is the cosine of the angle between the vectors of the words (Ruge, 1992; Schütze, 1992). Two words are similar if the cosine is above a specified threshold. The observations over the training and test data show that the translation equivalents are identified best when the cosine is higher than 0.7. However, the word-to-word alignment reduces some of the errors which increase in number when lowering the threshold. Therefore, the threshold is set at 0.4 acquiring a good balance between the number of the connections obtained and the error rate.

A second vector space matrix is built using the first two words in each clause on the assumption that clause-introducing words may express stronger word-to-word connections.

Some experiments with word similarity association measures e.g. the chi-square measure (Evert, 2005) failed to show any improvements.

Word forms are treated as instances of one and the same word if either their actual or normalised forms are equal (Kay and Roscheisen, 1993). The normalised forms cover correspondences between grammatically and semantically related words in languages with rich inflectional and derivational morphology. The morphology algorithm proposed by Kay and Roscheisen (1993) is applied for splitting potential suffixes and prefixes and for obtaining the normalised word forms. The vector space word-to-word connections are calculated for both actual and normalised forms and the obtained similarity measures are summed up.

- Levenshtein measure (Levenshtein, 1966)

  Church (1993) employs a method that induces sentence alignment by employing cognates (words that are spelled similarly across languages). Instead the standard Levenshtein distance (the number of edits required to transform a string A into another string B) is applied. The non-Latin characters are transliterated into Latin ones. The distance is calculated within a tolerance different for a different word length. The distance is then transformed into similarity by means of the tolerance.

$$\sqrt{1 - \frac{\text{levenshtein}}{\text{tolerance} + 1}}.$$

- Punctuation

  Similarity is calculated also if two words contain identical prefixes or suffixes which are punctuation marks or special characters. Punctuation and special characters are not all equal. Some of them are more robust, e.g. marks for currency and measurement, or mathematical symbols ($, , , %, +, <, >, =) or the different types of brackets. Others (e.g. comma, hyphen, colon, semi-colon) may be governed by language specific rules and may lead to improvement only for those pairs of languages that employ similar rules.

  The word-to-word similarity measure is the weighted sum of the above measures where the Levenshtein similarity is multiplied by 3, the punctuation similarity by 0.4 and the vector space similarity measure by 1, which is defined as a base.

  The similarity connections are sorted descendingly and sequentially processed. At each iteration only connections between dangling words are stored. Thus there is only one connection left for each word resulting in partial word alignment. The weights of all obtained word-to-word connections are summed up to produce the weight of the clause association that is propagated to the clause similarity calculation stage.

3.3.2 Length similarity

Zero-weighted similarity connections between clauses are collected using Gale-Church's distance measure. Thus connections are added without increasing the weight of the existing ones.

3.3.3 Weighted punctuation similarity

This similarity is calculated by the following formula

$$\sum_{Z \in PU} \min(\text{count}(Z \in cl_1),\ \text{count}(Z \in cl_2)),$$

where PU is the set of the punctuation marks and special symbols being prefixes and suffixes of words in the clauses processed.

3.4 Clause alignment with the bootstrapping method

The bipartite graph is built by filtering the set of the calculated clause similarity connections. The connected components of this graph form the clause beads. A conservative fallback strategy is applied to add the dangling clauses to the most appropriate bead. The filtering process starts by defining a threshold for grouping (1,2) and every clause similarity connection with weight above it is considered strong. In a way similar to word alignment, the remaining (weak) connections are sorted descendingly and processed one by one. If the processed connection relates clauses that are not attached to any bead, it passes the filter. In other words these two clauses form a 1:1 bead.

The bootstrapping method evaluated on the test corpus has precision above 94% and recall of 77%. To overcome this low recall we combine the Gale-Church algorithm with the core method.

3.5 Combined clause alignment

The combined method also distinguishes strong and weak clause connections by means of a threshold constant. At the beginning the Gale-Church results in clause alignment are compared with the strong connections. If they comply with the Gale-Church's beads, the weak connections are processed. The weak connections are added to the final graph if they do not contradict Gale-Church's output, i.e. when they do not connect clauses from two different beads.

In case of a strong connection the Gale-Church's alignment is discarded, assuming that the semantic and the syntactic similarities between clauses are more significant than the length.

4 Clause alignment evaluation

4.1 Test corpus

A test corpus was constructed for the purposes of method evaluation. It consists of 363,402 tokens altogether (174,790 for Bulgarian and 188,612 for English) distributed over five thematic domains: Fiction (21.4%), News (37.1%), Administrative (20.5%), Science (11.2%) and Subtitles (9.8%). The purpose of using a general testing corpus with texts from a variety of domains is to investigate method performance in a wider range of contexts.

Both Bulgarian and English parts of the corpus are first automatically segmented and then aligned at sentence level. The task of sentence detection in Bulgarian is carried out using a Bulgarian sentence splitter (Koeva and Genov, 2011). For sentence splitting of the English texts a pre-trained OpenNLP[2] model is used. Sentence alignment is produced using HunAlign[3] (Varga et al., 2005), with the alignment manually verified by human experts.

Clause splitting is considered a highly language dependent task and separate linguistic models need to be developed for each language. For the purposes of the present study, Bulgarian sentences are manually or semiautomatically split into clauses and for the English texts a pre-trained OpenNLP parser is used to determine clause boundaries followed by manual expert verification and post-editing (the task of automatic clause splitting falls outside the scope of the present study).

Subsequently, manual clause alignment is performed. Tables 1 and 2 present the number of sentences and clauses, respectively, in Bulgarian and English with their average length in tokens ($L_S(t)$) and in characters ($L_S(ch)$).

Language	**Sentences**		
	number	$L_S(t)$	$L_S(ch)$
Bulgarian	13,213	13.23	73.04
English	13,896	13.57	69.21
Total	**27,109**	–	–

Table 1: Number of sentences and their length.

Different models of clause alignment reflect interlingual symmetry or assymetry, such as: 1:1 for equivalent clauses in both languages; 0:1 or 1:0 if a clause in one of the languages is missing in the other; $1:N$ and $N:1$ ($N>1$) in the cases of different clause segmentation, when clauses contain the same information; $N:M$ ($N,M>1$) in relatively rare cases when the information is crossed among

[2] http://opennlp.apache.org/index.html

[3] http://mokk.bme.hu/resources/hunalign/

Language	Clauses		
	number	$L_S(t)$	$L_S(ch)$
Bulgarian	24,409	7.20	39.54
English	28,949	6.57	33.22
Total	**53,358**	–	–

Table 2: Number of clauses and their length.

clauses. The distribution of the models is given in Table 3.

Model	Frequency	% of all
0:1	553	2.53
1:0	412	1.88
1:1	17,708	80.88
1:2	2,055	9.39
1:3	309	1.41
1:4	98	0.45
2:1	588	2.69
2:2	81	0.37
2:3	15	0.07
3:1	31	0.14
3:2	7	0.03

Table 3: Distribution of bead models in the manually aligned corpus.

4.2 Evaluation

The precision is calculated as the number of true connections (between clauses in the two languages) divided by the number of the proposed connections, while the recall is the proportion of true connections to all connections in the corpus. The connections in a bead are the Cartesian product of the clauses in the first and the second language. The $K:0$ and $0:K$ bead models are considered as $K:1$ and $1:K$ by adding a fake clause.

The evaluation is performed both over the corpus as a whole and on each of the domain specific subcorpora included in it.

The evaluation of the clause alignment implementation of the Gale-Church algorithm on the same corpus shows overall precision of 0.902, recall – 0.891 and F_1 measure – 0.897. Although the original Gale-Church method performs very well in terms of both precision and recall, sentence alignment poses a greater challenge. The explanation for this fact lies in the broader scope of variations of clause correspondences as compared to sentences.

Domain	Precision	Recall	F_1
Total	**0.910**	**0.911**	**0.911**
Administrative	0.865	0.857	0.861
Fiction	0.899	0.902	0.901
News	0.933	0.946	0.940
Science	0.874	0.852	0.862
Subtitles	0.934	0.934	0.934

Table 4: Performance of the flexible method.

The bootstrapping method performs better in the translations with clause reordering. An example is the administrative subcorpus where Gale-Church gives precision/recall – 81.5%/79.7% compared to 86.6%/85.8% shown by the bootstrapping method. In the texts with less clause order asymmetries the results are close.

5 Application of clause alignment in SMT

Typical Moses[4] (Koehn et al., 2007) models are built on a large amount of parallel data aligned at the sentence level. For the purposes of the present study a specially designed parallel corpus is used. The aim is to demonstrate the effect of using syntactically enhanced parallel data (clause segmentation and alignment, reordering of clauses, etc.).

A series of experiments with Moses is designed to demonstrate the effect of training data modification on the performance of the SMT system. The different training datasets comprise the same sentences but differ in their syntactic representation. The baseline model is constructed on the basis of aligned sentence pairs. The first experiment is based on aligned clauses rather than sentences. The second experiment demonstrates the effect of reordering of the clauses within the source language sentences. The main purpose of the experiments is to demonstrate possible applications of the clause alignment method for training an SMT system, enhanced with linguistic information.

5.1 Training corpus

For the demonstration purposes of the present study we apply a small corpus of 27,408 aligned sen-

[4] http://www.statmt.org/moses/

tence pairs (comprising 382,950 tokens in Bulgarian and 409,757 tokens in English) which is semi-automatically split into clauses and automatically aligned at clause level. The current purposes of the research do not include the development of a full SMT model but focus on the demonstration of the effect of syntactical information on the performance of the SMT system. Thus, the size of the training corpus is considered sufficient for demonstration purposes. The parallel texts are extracted from several domains – Administrative, Fiction, News, Science, Subtitles.

5.2 Test corpus

The test corpus compiled for the purposes of evaluation of the SMT performance is independently derived from the Bulgarian-English parallel corpus and does not overlap with the training corpus. It however, resembles its structure and contains texts from the same domains as the training data. Table 5 gives the number of tokens in the Bulgarian and in the English part of the test corpus, with percent of tokens in the Bulgarian texts.

Domain	BG	ENl	% (BG)
Administrative	36,042	35,185	21.10
Fiction	34,518	38,723	20.21
News	64,169	62,848	37.57
Science	18,912	19,856	11.07
Subtitles	17,147	18,951	10.04
Total	**170,788**	**175,563**	

Table 5: Number of tokens in the test corpus.

5.3 Baseline model

The baseline model corresponds to the traditional Moses trained models and is constructed from aligned sentences in Bulgarian and English. The BLEU score for translation from Bulgarian into English is 16.99 while for the reverse it is substantially lower – 15.23. In the subsequent tests we observe the results for the Bulgarian-to-English translation only.

5.4 Clause level trained model

The first experiment aims to demonstrate that training of the model based on aligned clauses rather than sentences yields improvement. The assumption is that alignment at a sub-sentential level would improve word and phrase alignment precision by limiting the scope of occurrence of translational equivalents. On the other hand, however, lower level alignment reduces the number of aligned phrases. For this purpose clauses are the optimal scope for alignment as phrases rarely cross clause boundaries.

The results of the clause level training show small improvement of 0.11 in the BLEU score from 16.99 (baseline) to 17.10 for the Bulgarian-to-English translation.

5.5 Reordering of clauses

The second experiment relies on reordering of clauses within aligned sentences. The experiment aims at showing that reordering improves performance of SMT system.

A simple clause reordering task was carried out within the sentences on the parallel training corpus. Clause reordering involves linear reordering of clauses in the source language sentences to match the linear order of corresponding clauses in the target language sentences.

Reordering applies to cases where asymmetries are present in the alignment i.e. crossed connections between clauses, which is expected to vary across languages and domains. This suggests that the proportion of the corpus affected by reordering also depends on the language and on the domain. Based on an experiment with a smaller corpus, approximately 7% of the Bulgarian sentences are affected by reordering when adjusted to the English sentences.

The result is BLEU score of 17.12 compared to 16.99 (baseline) which yields an improvement of 0.13.

5.6 Analysis

The results obtained from the above two experiments show a small yet consistent improvement in the BLEU score. It shows a possibility to improve the results by applying parallel data enhanced by syntactic information, namely, aligned pairs at clause level, or sentences with reordered clauses.

The data, however, are not sufficient to draw a definite conclusion both on whether the improvement is stable and on which of the two methods –

using clause aligned pairs or reordered sentences – performs better.

6 Conclusions

The research done in the scope of this paper has shown that, on the one hand, the Gale-Church algorithm is applicable for clause alignment. The results achieved by the bootstrapping method, on the other hand, show that clause alignment may be appropriately improved by means of similarity measurement especially for the domain dependent tasks – particularly for the domains for which non-linear order of the translated clauses is typical. Experiments showed that especially for texts exhibiting alignment asymmetries our method for clause alignment outperforms Gale-Church considerably.

We applied automatic clause alignment for building a Moses training dataset enhanced with syntactic information. Two experiments were performed – first, involving aligned clause pairs, and the second using clause reordering in the source language assuming that the order of clauses in the target language defines relations specific for the particular language. The experiments suggest that the clause reordering might improve translation models.

The series of experiments conducted with Moses showed possible applications of the clause alignment method for training an SMT system, enhanced with linguistic information.

References

Sotiris Boutsis and Stelios Piperidis. 1998. OK with alignment of sentences. What about clauses? *Proceedings of the Panhellenic Conference on New Information Technology (NIT98)*. pp. 288–297.

Sotiris Boutsis and Stelios Piperidis. 1998. Aligning clauses in parallel texts. *Proceedings of the 3rd Conference on Empirical Methods in Natural Language Processing (EMNLP 1998)*. pp. 17–26.

Peter F. Brown, John Cocke, Stephen A. Della Pietra, Vincent J. Della Pietra, Frederick Jelinek, Robert L. Mercer and Paul S. Roossin. 1990. A statistical approach to language translation. *Computational Linguistics*, 16(2): 79–85.

Kenneth Church. 1993. Charalign: A program for aligning parallel texts at the character level. *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics (ACL 1993)*. pp. 1–8.

Brooke Cowan, Ivona Kucerová and Michael Collins. 2006. A Discriminative Model for Tree-to-Tree Translation. *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP 2006)*. pp. 232–241.

John DeNero and Dan Klein. 2008. The Complexity of Phrase Alignment Models. *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics (ACL 2008)*, Short Paper Track.

Stefan Evert. 2005. *The Statistics of Word Cooccurrences: Word Pairs and Collocations. Ph.D. thesis*. Institut fur maschinelle Sprachverarbeitung, University of Stuttgart.

William A. Gale and Kenneth. W. Church. 1993. A Program for Aligning Sentences in Bilingual Corpora. *Computational Linguistics*, 19(1): 75–102. URL: `http://acl.ldc.upenn.edu/J/J93/J93-1004.pdf`.

Chooi-Ling Goh, Takashi Onishi and Eiichiro Sumita. 2011. Rule-based Reordering Constraints for Phrase-based SMT. Mikel L. Forcada, Heidi Depraetere, Vincent Vandeghinste (eds.) *Proceedings of the 15th Conference of the European Association for Machine Translation (EAMT 2011)*. pp. 113–120.

Mridul Gupta, Sanjika Hewavitharana and Stephan Vogel. 2011. Extending a probabilistic phrase alignment approach for SMT. *Proceedings of the International Workshop on Spoken Language Translation (IWSLT 2011)*. pp. 175-182.

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

Chunyu Kit, Jonathan J. Webster, King Kui Sin, Haihua Pan, Heng Li. 2004. Clause Alignment for Hong Kong Legal Texts: A Lexical-based Approach. *International Journal of Corpus Linguistics*, 9(1): 29–51.

Philipp Koehn, Franz J. Och and Daniel Marcu. 2003. Statistical phrase-based translation. *Proceedings of the North American Chapter of the Association for Computational Linguistics (NAACL 2003)*. pp. 48–54.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, Evan Herbst. 2007. Moses: Open Source Toolkit for Statistical Machine Translation. *Annual Meeting of the Association for Computational Linguistics (ACL), demonstration session*. Prague, Czech Republic, June 2007.

Svetla Koeva, Diana Blagoeva and Siya Kolkovska. 2010. Bulgarian National Corpus Project. Nicoletta Calzolari, Khalid Choukri, Bente Maegaard, Joseph Mariani, Jan Odijk, Stelios Piperidis, Mike Rosner,

Daniel Tapias (eds.) *Proceedings of the 7th conference on International Language Resources and Evaluation (LREC 2010)*. pp. 3678–3684.

Svetla Koeva and Angel Genov. 2011. Bulgarian language processing chain. *Proceedings of Integration of multilingual resources and tools in Web applications. Workshop in conjunction with GSCL 2011, 26 September 2011, University of Hamburg*. (to appear)

Vladimir Levenshtein 1966. Binary codes capable of correcting deletions, insertions, and reversals. *Soviet Physics Doklady*, 10. pp. 707–710.

Chi-Ho Li, Minghui Li, Dongdong Zhang, Mu Li, Ming Zhou and Yi Guan. 2007. A probabilistic approach to syntax-based reordering for statistical machine translation. *Proceedings of the 45rd Annual Meeting of the Association for Computational Linguistics (ACL 2007)*. pp. 720–727.

Harris Papageorgiou 1997. Clause recognition in the framework of alignment. Ruslan Mitkov and Nicolas Nicolov, N. (eds.) *Current Issues in Linguistic Theory*, 136: 417–425. John Benjamins B.V.

Stelios Piperidis, Harris Papageorgiou and Sotiris Boutsis. 2000. From sentences to words and clauses. Chapter 6. Jean Veronis and Nancy Ide (eds.) *Parallel Text Processing: Alignment and Use of Translation Corpora. Text, Speech and Language Technology series*, 13: 117–137.

Ananthakrishnan Ramanathan, Pushpak Bhattacharyya, Karthik Visweswariah, Kushal Ladha and Ankur Gandhe. 2011. Clause-Based Reordering Constraints to Improve Statistical Machine Translation. *Proceedings of the 5th International Joint Conference on Natural Language Processing (IJCNLP 2011)*. pp. 1351–1355.

Gerda Ruge. 1992. Experiments on linguistically based term associations. *Information Processing & Management*. 28(3):317-332.

Hinrich Schütze. 1992. Context Space. *Working Notes of the AAAI Fall Symposium on Probabilistic Approaches to Natural Language*. pp. 113-120

Katsuhito Sudoh, Kevin Duh, Hajime Tsukada, Tsutomu Hirao, Masaaki Nagata. 2010. Divide and Translate: Improving Long Distance Reordering in Statistical Machine Translation. *Proceedings of the Joint 5th Workshop on Statistical Machine Translation and MetricsMATR*. pp. 418–427.

Daniel Varga, Laszlo Nemeth, Peter Halacsy, Andras Kornai, Viktor Tron, Viktor Nagy. 2005. Parallel corpora for medium density languages. *Proceedings of the RANLP 2005*. pp. 590–596.