*Human Language Technologies – The Baltic Perspective*
*A. Utka et al. (Eds.)*


*doi:10.3233/978-1-61499-442-8-87*

# Tracing Mistakes and Finding Gaps in Automatic Word Alignments for Latvian-English Translation

Valdis GIRGZDIS[a], Maija KALE[a,1], Martins VAICEKAUSKIS[a], Ieva ZARINA[a], and Inguna SKADIŅA[b]
[a] *University of Latvia*
[b] *Tilde, Latvia*

**Abstract.** This paper aims to contribute to an in-depth understanding of computer based word alignment processes in machine translation (MT). The performance of word alignment, based on IBM models and incorporated in *GIZA++,* has been widely discussed in machine translation literature. The debate has lead towards a general consensus that GIZA++ does not provide sufficiently good results for word alignments. In this paper, we analyse the performance of *GIZA++* and *Fast Align* for the Latvian-English pair against the manually aligned Gold Standard. Experiments showed that *Fast Align* proved to be approximately 2-3% more accurate and three times faster than GIZA++ in the alignment task. Where it concerns pre-processing, the removal of articles has a small, but positive, influence on alignment quality and machine translation output. We also present a Word Alignment Visualisation tool for analysis and editing of word alignments.

**Keywords.** word alignment, alignment guidelines, pre-processing, Word Alignment Visualisation tool, Latvian language

## Introduction

The performance of word alignment based on IBM models 1 to 5 [1] and the Hidden Markov Model [2], [1], and incorporated in GIZA++ [3], has been widely discussed in MT literature (e.g., [4], [5]). The debate has lead towards a general consensus that GIZA++ does not provide sufficiently good results for word alignments to be used in further steps of MT. Still, GIZA++ is widely used in various SMT systems. Also, improvements in the quality of word alignment do not necessarily lead to improved translation quality [6], leaving open the question of whether better word alignment would lead to better MT results.

This paper aims to trace the mistakes and gaps in computer based word alignment, contributing to the existing research on alignment mistakes [7], [8], as well as to the question of whether better word alignments would generate better MT output [6], [9].

While *GIZA++* has historically been part of SMT experiments, *Fast Align* [10] is much faster. Therefore, the authors consider both alignment tools (*GIZA++* and *Fast*

[1] Corresponding author: University of Latvia, Raiņa bulvāris 19, Riga, Latvia; E-mail: maija.kale@lu.lv

*Align*) and trace the similarities and differences in terms of mistakes and gaps generated during word alignment.

In this paper, we analyse the performance of *GIZA++* and *Fast Align* for the Latvian-English language pair against the manually aligned Gold Standard. The experiments are performed in the LetsMT platform [11].

## 1. Gold Standard

For evaluation of our experiments, a manually annotated Gold Standard (GS) was created. It contains 512 Latvian and English sentences from various domains, as described in [12].

Before annotation, the annotation guidelines (herein - GS guidelines) were created based on Blinker guidelines [7] and Czech-English language pair annotation guidelines [8]. Special attention was paid to issues related to alignment of articles, prepositions, punctuation marks, double negation, etc.

These guidelines include the following main instructions:

1. The main rule of the GS guidelines was to "Link as many words as necessary, but as few as possible", allowing the many-to-many (or M-to-N) word alignment option in cases when the phrases cannot be further divided without losing their meaning.
2. Non-alignment rules in cases when: a) the translation is simply incorrect, b) it is impossible to directly relate the word in the source language to the translation, c) words are missing (not necessary for keeping the translation correct), or d) punctuation does not match.
3. For special cases (*a, the, a, of, in, by,* etc.), a specific rule for how to align these word classes to the head word was developed; several situations are illustrated in Figure 1.


**Figure 1.** Gold Standard alignment samples for articles and prepositions

4. So that complex grammar construction cases would not lose their meaning, one-to-many or many-to-one word alignments are acceptable. Examples are shown in Figure 2.

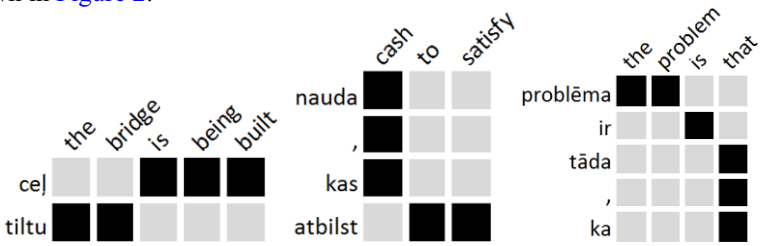


**Figure 2.** Alignment examples for complex syntactic constructions

5. In cases of double negation, all words which are related to negation should be aligned in order to avoid a situation where a negative verb in one language is aligned with a positive one in another language.

Word alignment was performed by two annotators with the UMIACS Word Alignment Interface [13]. Each sentence was first aligned and then cross-checked by another annotator. Consensus on the final alignment was reached via discussion.

Most of the disagreement between the annotators was in cases of many-to-many word alignments versus leaving the words unaligned. Some annotators considered that according to the annotation guidelines, each word of a source language phrase that cannot be divided further should be aligned with each word in the target language phrase. However, in this same situation, other annotators applied the non-alignment rule from the guidelines, which states that if there is no direct translation, then the words should be left unaligned. In most cases, it was decided to make M-to-N word alignments, which leads towards a discussion opened up by Fraser and Marcu [14].

## 2. Application of Alignment Methods

In order to choose the most suitable alignment method for GIZA++, several experiments were carried out. The Gold Standard and the first one million sentences from the DGT-TM 2013 corpus [15] were used for this purpose. As shown in Table 1, the best result in terms of BLEU score [16] was achieved with the default method - grow-diag-final-and[2]. The heuristic "grow-diag", which is the intersection, was a close second. Other alignment options proved to be less efficient.

**Table 1.** Comparison of different Giza++ alignment methods

| Alignment method | BLEU |
|---|---|
| grow diag final and | 0.3893 |
| grow diag | 0.3885 |
| tgttosrc | 0.3322 |
| grow_diag_final | 0.3155 |
| union | 0.3155 |

Upon taking a closer look at the GIZA++ output, one can see that the GIZA++ default alignment method allows many-to-one word alignment in both directions (Figure 3), while intersection (Figure 4) excludes such an option.

[2] The default heuristic grow-diag-final starts with the intersection of the two alignments and then adds additional alignment points.

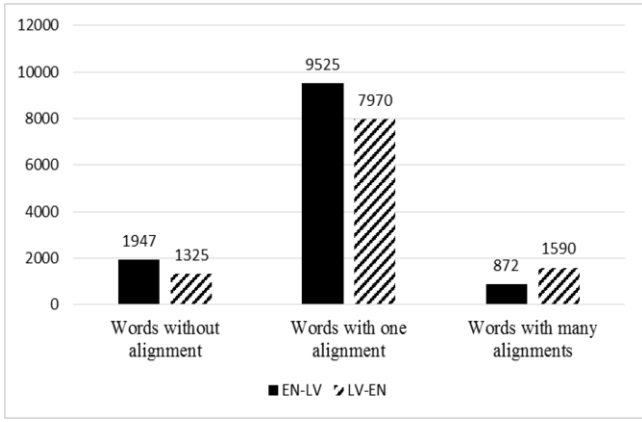


**Figure 3.** GIZA++ Default Heuristic (grow-diag-final-and).

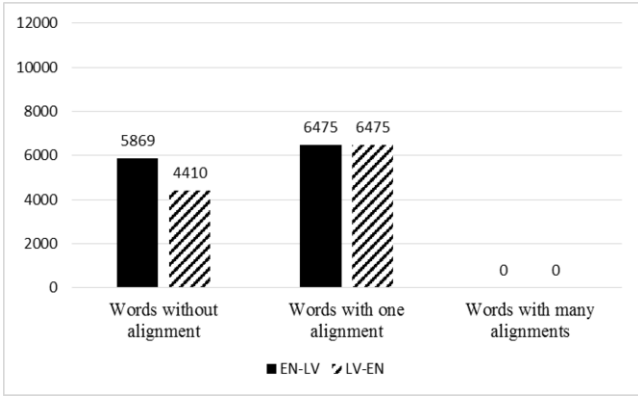


**Figure 4.** GIZA++ Intersection Heuristic (grow-diag).

When analysing the results obtained for two different GIZA++ alignment methods and comparing them with the manually aligned Gold Standard, several 'typical' GIZA++ mistakes were noticed. A summary of these is provided in Table 2.

**Table 2.** Comparison of GIZA++ alignment methods

| Intersection (grow-diag) | Default alignment (grow-diag-final-and) |
|---|---|
| In situations when GS contains many-to-one, one-to-many, or many-to-many alignment structures, intersection provides only one-to-one alignments. | In situations when GS contains many-to-one, one-to-many, or many-to-many alignment structures, default heuristics provide a maximum of one-to-many or many-to-one alignments. |
| No articles and prepositions are aligned. | Articles and prepositions are aligned, but the alignment structure does not follow any logistics and is sporadic. |
| Long distance reordering causes difficulties for the intersection heuristic; word alignment structures that are more linear are performed better (alignment near the matrix's diagonal works better than in cases when it deviates away from diagonal word alignment). | Long distance reordering causes difficulties for the default heuristic; word alignment structures that are more linear are performed better (alignment near the matrix's diagonal works better than in cases when it deviates away from diagonal word alignment). |
| When a sentence in active voice is translated as a sentence in passive voice, or vice versa, there are difficulties for GIZA++ to pursue alignments. | When a sentence in active voice is translated as a sentence in passive voice, or vice versa, there are difficulties for GIZA++ to pursue alignments. |

## 3. Analysis of Mistakes and Gaps

### *3.1. Experiment Overview*

The experiments on word alignment using *GIZA++* and *Fast Align* were conducted on a parallel corpora containing one, two, three, and four million sentences from the DGT-TM 2013 corpus together with the sentences from GS. The maximum size of corpora (four million) was determined by the memory restrictions of *Fast Align*.

The F-measure was calculated on GS for each corpus (Table 3). We found that *Fast Align* proved to be approximately 2-3% more accurate and was consistently faster to align (on average three times faster, although the alignment speed may be affected by the server memory availability during heavy workflow).

**Table 3**. F-measure against GS

| Corpus size / Aligner | 1 million | 2 million | 3 million | 4 million |
|---|---|---|---|---|
| Giza++ | 59.9 | 62.1 | 63.5 | 63.5 |
| Fast Align | 63.3 | 64.4 | 65.1 | 67.4 |

By splitting up the results into precision and recall, one can see that with alpha being 50%, GIZA++ performs better in precision, while *Fast Align* provides better results in recall (Table 4).

**Table 4.** F-measure for GIZA++ and *Fast Align* with alpha 50%

| Corpus size/Aligner | F-measure | 1 million | 2 million | 3 million | 4 million |
|---|---|---|---|---|---|
| Giza++ | Precision | 0.933 | 0.939 | 0.943 | 0.944 |
| | Recall | 0.441 | 0.464 | 0.479 | 0.479 |
| Fast Align | Precision | 0.879 | 0.885 | 0.888 | 0.899 |
| | Recall | 0.536 | 0.546 | 0.682 | 0.578 |

Benchmarking *GIZA++* and *Fast Align* alignments against the GS showed that *Fast Align* aligns more words and, at the same time, makes more mistakes than GIZA++, which nevertheless results in better word alignment performance (Figure 5).

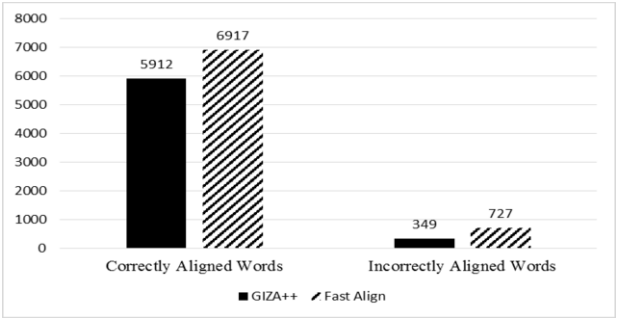


**Figure 5.** Alignment results for GS on a corpus of 4 million sentences.

### *3.2. Analysis of Mistakes and Gaps*

The following situations were observed, revealing discrepancies between *GIZA++* and *Fast Align*.

- No articles and propositions are aligned by *GIZA++* and *Fast Align*.
- In situations when GS contains many-to-one, one-to-many, or many-to-many alignment structures, *GIZA++* and *Fast Align* provide either no or one-to-one alignment.
- Long distance reordering causes difficulties for *GIZA++* and *Fast Align* to align words, since alignment structures that are more linear are handled better (alignment near the matrix's diagonal works better than in cases when it deviates away from diagonal word alignment). Here, however, *Fast Align* performs better than *GIZA++*.
- Taking into account that the Latvian language is an inflected language with many inflected forms, *GIZA++* and *Fast Align* do not recognise words in some cases. Here, however, *Fast Align* performs better than *GIZA++*.

### *3.3. Pre-processing Experiments*

Taking into account that *Fast Align* provided better results in terms of F-measure in previous experiments, several experiments were carried out to identify which corpus pre-processing operations can result in better word alignment (Table 5). Methods included experiments with the removal of articles and commas, as well as adding the article to the following word.

**Table 5.** Influence of pre-processing on F-measure and BLEU

| Method | Baseline | Before symmetrisation | | | | After symmetrisation | |
|---|---|---|---|---|---|---|---|
| | | Removing 'the', 'a', 'an' | Removing commas | Removing 'the', 'a', 'an', 'of', 'by' | Joining 'the' with the next word | Removing 'the', 'a', 'an' | Removing 'the', 'a', 'an', 'of', 'by' |
| BLEU | 0.5331 | **0.5402** | 0.5275 | 0.5352 | 0.5265 | 0.5343 | 0.5316 |
| F-measure | 0.6021 | **0.6182** | 0.4254 | 0.4371 | | | |

To evaluate the influence of pre-processing on MT, an SMT system was trained on the aligned corpus and evaluated. As shown in Table 5, some improvements in terms of F-measure and BLEU score were achieved when articles were removed from the training data.

## 4. Word Alignment Visualisation tool

In order to make the analysis of GIZA++ and *Fast Align* output convenient, the authors have developed a Word Alignment Visualisation (WAV) tool. WAV can be used as an editor to prepare and manually edit word alignments. It also allows to visualise and edit the results of GS, *GIZA++,* and *Fast Align* in one sentence based matrix (Figure 6).

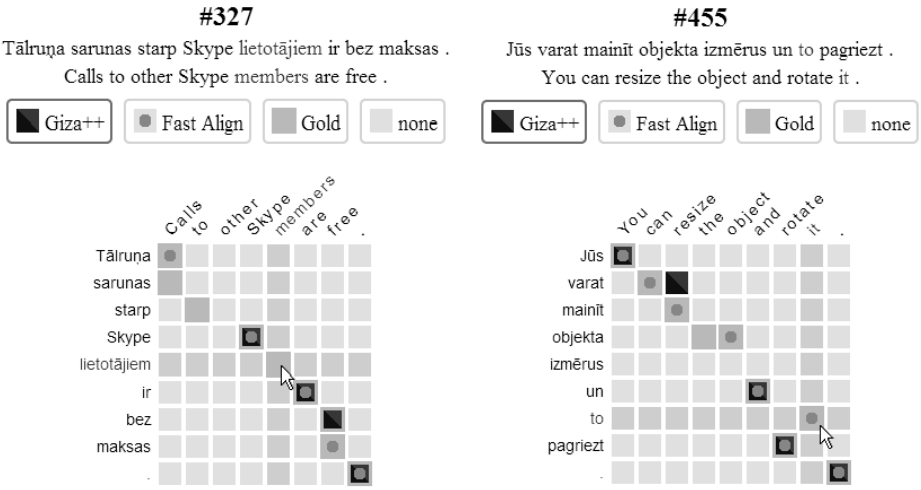


**Figure 6.** Giza++, Fast Align, and GS comparisons using WAV.

WAV consists of data generator and visualizer. The generator, written in Python, creates *.htm* file from the alignment files (in *GIZA++* or *Fast Align* format). Alignment format used for WAV tool is shown in Figure 7.

```
0-0 1-0 2-1 3-3 4-4 5-5 6-6 7-6 8-7
```

**Figure 7.** Alignment format used for WAV tool

The visualizer (browser) then generates an interactive visualization of alignments using HTML, CSS and JS technologies.

## 5. Conclusion

This paper provides an insight into computer based word alignment; it traces the performance of *GIZA++* and *Fast Align* and benchmarks it against the manually aligned Gold Standard for the Latvian-English language pair. Experiments illustrate the potential changes in *GIZA++* and *Fast Align* settings and/or word alignment files, which can lead to a better quality of word alignment.

To measure the influence of different settings on the quality of word alignment and machine translation, F-Score and BLEU score have been calculated. Our experiments show that *Fast Align* has proved to be approximately 2-3% more accurate (in terms of F-measure) and approximately three times faster than *GIZA++* in the alignment task.

Where it concerns pre-processing, the removal of articles has had a small, but positive influence on alignment quality and BLEU score.

This paper has also introduced word alignment visualisation tool, which provides a convenient environment for analysis and comparison of GIZA++, *Fast Align*, and GS word alignments.

## 6. Acknowledgements

The research leading to these results has received funding from the research project "Optimization methods of large scale statistical models for innovative machine translation technologies", project financed by The State Education Development Agency (Latvia) and European Regional Development Fund, contract nr. 2013/0038/2DP/2.1.1.1.0/13/APIA/VIAA/029. We would like to thank Pēteris Ņikiforovs, Raivis Skadiņš and Valters Šics for their advices and contributions.

## References

[1] Brown, P., F., Della Pietra, S., A., Della Pietra, V., J. and Mercer, R., L. (1993). The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2), 263–311.

[2] Vogel, S., Ney, H., Tillmann, Ch. (1996). Hmm-based word alignment in statistical translation. *COLING-96*, 836–841.

[3] Och, F., J. and Ney, H. (2003). A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1),19–51.

[4] Fishel, M. (2010). Simpler Is Better: Re-evaluation of Default Word Alignment Models in Statistical MT. *Proceedings of PACLIC 2010*.

[5] Riley, D. and Gildea, D. (2010). Improving the performance of GIZA++ using variational Bayes. *The University of Rochester, Computer Science Department, Tech. Rep.* 963.

[6] Schoenemann, Th. (2013). Training Nondeficient Variants of IBM-3 and IBM-4 for Word Alignment. *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, 22–31.

[7] Melamed, D. (1998). Annotation Style Guide for the Blinker Project. Version 1.0.4. Philadelphia, University of Pennsylvania.

[8] Kruijff-Korbayová, I., Chvátalová, K., Postolache, O. (2006). Annotation guidelines for Czech-English word alignment. *Proceedings of LREC 2006*.

[9] Ganchev, K., Graca, J., Tasker, B. (2008). Better Alignments = Better Translations? *Proceedings of ACL-08: HLT, Association for Computational Linguistics, Columbus, Ohio*, 986-993.

[10] Dyer, C., Chahuneau, V., and Smith, N., A. (2013). A Simple, Fast, and Effective Reparameterization of IBM Model 2. *Proceedings of NAACL*.

[11] Vasiļjevs, A., Skadiņš, R., Tiedemann, J. (2012). LetsMT!: a cloud-based platform for do-it-yourself machine translation. *Proceedings of ACL2012*, 43-48.

[12] Skadiņš, R., Goba, K., Šics, V. (2010). Improving SMT for Baltic Languages with Factored Models. *Proceedings of the Fourth International Conference Baltic HLT 2010, Frontiers in Artificial Intelligence and Applications*, Vol. 2192, 125-132.

[13] Madnani, N., Hwa, R. (2004). UMIACS Word Alignment Interface, http://www.umiacs.umd.edu/~nmadnani/alignment/forclip.htm.

[14] Fraser, A., Marcu, D. (2007). Getting the structure right for word alignment: LEAF. *Conference on Empirical Methods in Natural Language Processing and Conference on Computational Natural Language Learning*, 51–60.

[15] Steinberger, R., Eisele, A., Klocek, S., Pilos, S., and Schlüter, P. (2012). DGT-TM: A freely Available Translation Memory in 22 Languages. *Proceedings of the 8th international conference on Language Resources and Evaluation (LREC'2012). Istanbul, Turkey*, 454-459.

[16] Papineni, K., Roukos, S., Ward, T., Zhu, W. (2002). BLEU: a method for automatic evaluation of machine translation. *Proceedings of the 40th Annual Meeting of the Association of Computational Linguistics.*