EXPERIMENTS WITH RUSSIAN TO KAZAKH SENTENCE ALIGNMENT

Zhenisbek Assylbekov
Nazarbayev University, Astana, Kazakhstan
zhassylbekov@nu.edu.kz
Bagdat Myrzakhmetov, Aibek Makazhanov
National Laboratory Astana, Astana, Kazakhstan
bagdat.myrzakhmetov@nu.edu.kz, aibek.makazhanov@nu.edu.kz

Sentence alignment is the final step in building parallel corpora, which arguably has the greatest impact on the quality of a resulting corpus and the accuracy of machine translation systems that use it for training. However, the quality of sentence alignment itself depends on a number of factors. In this paper we investigate the impact of several data processing techniques on the quality of sentence alignment. We develop and use a number of automatic evaluation metrics, and provide empirical evidence that application of all of the considered data processing techniques yields bitexts with the lowest ratio of noise and the highest ratio of parallel sentences.

Keywords: *sentence alignment, sentence splitting, lemmatization, parallel corpus, Kazakh language*

УДК 81’32

1. Introduction

Sentence alignment (SA) is the problem of identification of parallel sentences (pairs of sentences that constitute translations from a source to a target language) in a given pair of source and target documents, where the target document is assumed to be a translation of the source (mutual translation assumption is also common). More formally, given a source document D_s and a target document D_t represented as lists of sentences S and T respectively, SA is the task of building a list of pairs P, where each pair p contains 0 or more (ideally one) source sentence(s) aligned to 0 or more (ideally 1) target sentence(s). Approaches that consider sentence length correlations [1, 2], bilingual lexicon-based solutions [3], and combinations of the two [4] have been proposed in the past to solve this problem in a sufficiently accurate and efficient manner. In this paper we do not offer a new solution to the problem, nor do we try to improve the existing approaches. Our goal is to investigate what could be done to the input data (not to the methods) to improve the quality of SA.

We begin by asking a few questions, which are inspired directly by the definition of the problem and by ways of solving it. First, a formal definition of SA problem assumes that documents to be aligned are already split into sentences. However, in practice it is almost never the case, and one has to perform splitting before SA. Assuming that one uses for that a statistical approach that requires training, e.g. punkt splitter [5], a question regarding the choice of training data arises: *does it suffice to train the splitter on any data, or would it be beneficial to train on a sample drawn from a target domain?* Second, assuming one uses a lexicon based approach to SA, *should one bother trying to reduce typos and data sparsity of the input, and what lexicon to use automatically induced or handcrafted?* Lastly, *after sentences have been aligned can we still increase the portion of parallel pairs?* In an attempt to answer these questions, we propose to employ the following five data processing techniques: (i) *domain adapted sentence splitting*; (ii) *error correction*; (iii) *lemmatization* (to reduce sparsity); (iv) *use of handcrafted bilingual lexicons*; (v) *junk removal*.

The objective of this work is to assess the impact of the proposed data processing techniques on SA accuracy and find the combination of thereof which maximizes the quality of parallel corpora produced by SA.

2. Data Collection

For our experiments we have crawled three websites, `akorda.kz`, `strategy2050.kz`, `astana.gov.kz`, using our own Python scripts to download only specific branches of these sites - mainly news and

announcements. The choice of these specific websites is motivated by the fact that all of them provide built-in document alignment, i.e. each news article or announcement in Russian contains a link to a corresponding translation into Kazakh (sometimes translation direction may be opposite). Rare exceptions to this behavior include cases where translation link is absent or broken. Such pairs are not included into the data set. The obtained pairs of HTML documents are parsed in a site-specific manner with the help of Python `BeautifulSoup` library to produce raw Russian and Kazakh texts aligned on the document level.

2. Baseline Sentence Alignment and Data Processing Techniques

Let us describe the basic sentence alignment (BSA) procedure that does not assume any of the data processing techniques (DPTs) that we propose. At the sentence splitting stage BSA uses NLTK `punkt` tool [5] trained on approximately 200-250 Mb of plain texts from Russian and Kazakh Wikipedias. Next we tokenize the documents with a Perl-script from an SMT-toolkit Moses [6]. Sentence alignment is performed on tokenized and lowercased texts using `hunalign` [4]. After sentences are aligned we restore their original, non-tokenized and non-lowercased, format. In what follows we describe the implementation of the DPTs that we propose.

Domain adapted sentence splitting. To see whether we can gain any improvements at sentence splitting step, we train `punkt` on 350-370 Mb of text from the news domain rather than on random Wikipedia texts and supply it with a list of abbreviations in Russian and in Kazakh.

Error correction. In this work we consider a light-weight error correction procedure, which involves normalization of scripts (alphabets) used in a given text. Electronic texts written in Cyrillic (Russian and Kazakh alike), especially those which were digitized in 1990s, sometimes suffer from mixed scripts, i.e. when Latin letters are used instead of Cyrillic ones and vice versa: e.g. in a Kazakh word "ecірткі" it is possible to replace the letters 'e' (u+0435), 'c' (u+0441), 'і' (u+0456), 'p' (u+0440) with their Latin homoglyphs, 'e' (u+0065), 'c' (u+0063), 'i' (u+0069) and 'p' (u+0070), which allows a total of 2^5=32 spelling variations. To reduce data sparseness that may result from this, we developed a tool which tries to resolve unambiguous cases.

Lemmatization. Another possible way to improve sentence alignment is a prior lemmatization of texts. Theoretically this should decrease data sparseness and be helpful when combined together with automatic construction of a bi-dictionary. Also, one can try to align sentences when both texts and handcrafted dictionary are lemmatized. For Russian-side lemmatization we use an open source tool Mystem [7], and for Kazakh – morphological disambiguation tool developed by Makhambetov et al. [8].

Adding bilingual dictionaries for sentence alignment. In the baseline approach no bilingual dictionaries are provided to `hunalign`, and very often such dictionaries are not available, especially for low-resourced languages. In such cases one can construct and exploit rough bi-dictionaries in three steps: (1) apply the baseline sentence alignment; (2) use `hunalign` again to align already aligned texts with the `-autodict` option - the byproduct of this step is a bi-dictionary; (3) finally, apply `hunalign` to the original non-aligned texts with the obtained bi-dictionary. We experiment with both options, using as a handcrafted dictionary a compilation of resources obtained from `Bitextor` [9], `Apertium-kaz` [10], and `www.mtdi.kz`.

Junk removal. Finally, we believe that removing the following sentence pairs should benefit the final corpora (hereinafter such pairs are called "junk"):

- at least one of the sides (Kazakh or Russian) is empty;
- at least one of the sides does not contain any letters (Latin and Cyrillic);
- both sides are identical after tokenization and lowercasing.

3. Evaluation Metrics

The most reliable way to evaluate the quality of SA is to perform human evaluation by checking the output of an automatic SA method, and calculating its accuracy, i.e. percentage of correct alignments in the total number of aligned pairs. To evaluate the baseline SA in this fashion, we ran the baseline on our data set and on the data crawled from an additional page-aligned website (`adilet.zan.kz`). We then randomly sampled 800 sentence pairs (including null alignments produced by `hunalign`) and asked three annotators to label each pair

as parallel or not. The inconsistencies were resolved by the fourth annotator. In Table 1 we present the results of this procedure (averages are calculated excluding the results for `adilet.zan.kz` for comparison purposes).

Table 1. Accuracy of the baseline SA, per website and per annotator

Web-site	Annotator 1	Annotator 2	Annotator 3	Annotator 4
`adilet.zan.kz`	0.9375	0.9425	0.8825	**0.9075**
`akorda.kz`	0.9525	0.9450	0.8675	**0.9050**
`astana.gov.kz`	0.7925	0.7950	0.7325	**0.7400**
`strategy2050.kz`	0.7700	0.7525	0.6575	**0.6900**
Average	0.8383	0.8308	0.7525	**0.7783**

As we can see, on a sample of our data set (the latter three websites) the baseline SA method achieves the average accuracy of ~78%. As we will show later application of the DPTs can increase the accuracy. But to show that, we need to develop a more efficient way of computing SA, because to test all configurations of the DPTs would require us to perform expensive human evaluation procedure up to 10 times.

Table 2. Features used in a learning-based SA accuracy estimator

#	DC	Feature	---	#	DC	Feature
1,2	S,T	length in characters		19,20	S,T	count of personal initials
3	ST	MMR(F1,F2)		21	ST	COS(F19*,F20*)
4,5	S,T	length in tokens		22,23	S,T	ratio of alphanumerics
6	ST	MMR(F4,F5)		24	ST	MMR(F22,F23)
7,8	S,T	count of symbols		25,26	S,T	count of words in quotes
9	ST	COS(F7*,F8*)		27	ST	MMR(F25,F26)
10,11	S,T	count of numerals		28,29	S,T	count of words in parenthesis
12	ST	COS(F10*,F11*)		30	ST	MMR(F25,F26)
13,14	S,T	count of digits		31	ST	num. of tokens in identical pairs
15	ST	COS(F13*,F14*)		32	ST	min-max ratio between unique tokens in source and target sentences
16,17	S,T	count of latin alphanumerics				
18	ST	COS(F16*,F17*)		33-35	ST	`Hunalign` score: absolute, relative, min-max scaled.

To compute the accuracy estimate of SA more efficiently we cast the SA problem as a classification task, where given a pair of source and target sentences, a supervised learning algorithm estimates the probability of the pair being parallel. We design a set of 35 features listed in Table 2, where each feature has a domain of calculation (DC), and can be calculated for the (S)ource or the (T)arget sentence, or for both (ST). MMR refers to min-max ratio, e.g. MMR(F1,F2) means that the minimum of features 1 and 2 is divided by the maximum of the two. Similarly, COS refers to cosine similarity calculated for the count-vectors of a given pair of features.

We extract these features from the annotated sample that was used for human evaluation and perform a five-fold cross-validation using a range of classifiers implemented in Python `scikit-learn` library. Gradient Boosting classifier achieved the highest F-measure of 0.94 (per-fold average) and the lowest variance of 0.08. Therefore, we use this classifier as a rough estimator of SA accuracy as follows. Given the alignment pairs produced by SA, the estimator classifies each pair as parallel or not. The ratio of pairs classified as parallel to the total number of pairs provides the accuracy estimate.

4. Experiments and Results

We compare different configurations of DPTs. To refer to a specific technique we use the following abbreviations: adapted splitting - AS, error correction - EC, automatically obtained dictionary - AD, handcrafted dictionary - HD, lemmatized handcrafted dictionary - LHD, lemmatization - L, junk removal - JR.

We measure the quality of produced bitexts in the total number of parallel pairs (P) and the automatic accuracy estimation (P/T). As it can be seen from Table 3, combined application of all DPTs (AS+EC+L+LHD+JR) achieves the highest accuracy per-site and on average improves ~6% over the baseline. It also produces about 5.5K more parallel sentences (total) than the baseline, and only 40 pairs less than the same configuration shy of JR. However, notice how drastically junk removal increases the accuracy of SA, more than 5%. Hence, we indeed can increase the portion of parallel sentences after SA has been performed, through the removal of the pairs which are very unlikely to be parallel. We also notice that text lemmatization applied without the use of a handcrafted dictionary (AS+EC+L) produces far less parallel sentences than the baseline and is only 0.28% more accurate. Perhaps more surprisingly adding an automatically obtained dictionary to this configurations (AS+EC+L+AD) makes matters even worse.

Table 3. Qualities of bitexts produced using various processing techniques

Method	**akorda.kz**		**astana.gov.kz**		**strategy2050.kz**		**total**	**average**
	P	**P/T**	**P**	**P/T**	**P**	**P/T**	**P**	**P/T**
Baseline	70,956	0.9116	63,731	0.7215	201,678	0.6650	336,365	0.7660
AS+EC	70,860	0.9171	63,777	0.7310	202,354	0.6740	336,991	0.7740
AS+EC+AD	71,002	0.9193	63,298	0.7266	200,319	0.6681	334,619	0.7713
AS+EC+HD	71,062	0.9199	**64,210**	0.7365	204,716	0.6818	339,988	0.7794
AS+EC+LHD	71,089	0.9203	64,159	0.7356	204,857	0.6822	340,105	0.7794
AS+EC+L	70,605	0.9138	63,260	0.7246	199,675	0.6661	333,540	0.7682
AS+EC+L+AD	70,862	0.9178	62,929	0.7213	198,500	0.6638	332,291	0.7676
AS+EC+L+HD	71,114	0.9204	64,119	0.7345	206,225	0.6880	341,458	0.7810
AS+EC+L+LHD	**71,129**	0.9208	64,029	0.7333	206,797	0.6899	**341,955**	0.7813
AS+EC+L+LHD+JR	71,115	**0.9488**	64,014	**0.7823**	**206,786**	**0.7667**	341,915	**0.8326**

To measure the level of noise (the lower the better) in the produced bitexts, we calculate proportion of short pairs[1] among parallel pairs (S/P), and proportion of junk pairs among all pairs (J/T). From Table 3 we notice that a complete set of DPTs achieves the second lowest S/P ratio after the AS+EC+L+AD configuration, which also achieves the second lowest J/T ratio. Thus, using auto-induced dictionary on lemmatized text produces least amount of parallel sentences and the lowest ratio of thereof, but resulting bitexts actually come out less noisy than in other DPT configurations. We will study this strange behavior in the future.

Table 4. Noise level in bitexts produced using various processing techniques

Method	**akorda.kz**		**astana.gov.kz**		**strategy2050.kz**		**average**	
	S/P	**J/T**	**S/P**	**J/T**	**S/P**	**J/T**	**S/P**	**J/T**
Baseline	0.0190	0.0294	0.0098	0.0610	0.0202	0.0982	0.0163	0.0629
AS+EC	0.0172	0.0291	0.0084	0.0586	0.0195	0.0969	0.0150	0.0615
AS+EC+AD	0.0169	0.0294	0.0078	0.0588	0.0193	0.0974	0.0147	0.0619
AS+EC+HD	0.0172	0.0292	0.0084	0.0589	0.0193	0.0980	0.0150	0.0620
AS+EC+LHD	0.0171	0.0294	0.0084	0.0591	0.0193	0.0981	0.0149	0.0622
AS+EC+L	0.0171	0.0296	0.0084	0.0613	0.0198	0.0979	0.0151	0.0630
AS+EC+L+AD	**0.0167**	0.0297	**0.0072**	0.0620	**0.0191**	0.0950	**0.0143**	0.0622
AS+EC+L+HD	0.0170	0.0295	0.0084	0.0627	0.0192	0.0997	0.0149	0.0640
AS+EC+L+LHD	0.0170	0.0294	0.0085	0.0629	0.0192	0.1002	0.0149	0.0642
AS+EC+L+LHD+JR	0.0168	**0.0000**	0.0082	**0.0000**	0.0192	**0.0000**	0.0147	**0.0000**

[1] Short pairs are defined as those where both sides, Kazak and Russian, contain *three* or less words. Usually such text chunks are dates, titles, enumerations, etc., and they do not qualify as full sentences.

5. Conclusion

In this work we have shown that various techniques of data processing can increase the accuracy of sentence alignment and reduce the level of noise in the resulting bitexts. We provided empirical evidence that combined application of five simple data processing techniques before and after sentence alignment results in production of parallel corpora with the lowest ratio of noise and the highest ratio of parallel sentences.

Acknowledgements. This work has been funded by the Nazarbayev University under the research grant №064-2016/013-2016, and by the Committee of Science of the Ministry of Education and Science of the Republic of Kazakhstan under the targeted program O.0743 (0115PK02473).

ЭКСПЕРИМЕНТЫ ПО ВЫРАВНИВАНИЮ ПАРАЛЛЕЛЬНЫХ ТЕКСТОВ ПО ПРЕДЛОЖЕНИЯМ ДЛЯ РУССКО-КАЗАХСКОЙ ПАРЫ

Асылбеков Женисбек Аманбаевич
Назарбаев Университет, Астана, Казахстан
zhassylbekov@nu.edu.kz
Мырзахметов Багдат Омаралиевич, Макажанов Айбек Омиржанович
Национальная Лаборатория, Астана, Казахстан
bagdat.myrzakhmetov@nu.edu.kz, aibek.makazhanov@nu.edu.kz

Выравнивание параллельных текстов по предложениям является заключительным этапом построения параллельного корпуса и возможно оказывает наибольшее влияние качество конечного продукта и на точность систем машинного перевода, использующих этот корпус для обучения. Качество же выравнивания по предложениям, в свою очередь, также зависит от ряда факторов. В данной статье мы исследуем влияние некоторых способов обработки данных на качество выравнивания по предложениям. Мы разрабатываем и используем несколько автоматических метрик оценки качества, и приводим эмпирические доказательства того, что совокупное использование всех рассмотренных способов обработки данных приводит к получению параллельных корпусов с наименьшей долей шума и наибольшей долей параллельных предложений.

Ключевые слова: *выравнивание по предложениям, разбивка по предложениям, лемматизация, параллельный корпус, казахский язык*

References

1 Peter F. Brown, Jennifer C. Lai, and Robert L. Mercer. “Aligning sentences in parallel corpora”. ACL, 1991.

2 William A. Gale and Kenneth Ward Church. “A program for aligning sentences in bilingual corpora”. ACL, 1991.

3 Martin Kay and Martin Roscheisen. “Text-translation alignment”. Computational Linguistics, 19(1), 1993.

4 Varga, Daniel et al. “Parallel corpora for medium density languages”. AMSTERDAM STUDIES IN THE THEORY AND HISTORY OF LINGUISTIC SCIENCE SERIES 4 292 (2007): 247.

5 Kiss, Tibor, and Jan Strunk. “Unsupervised multilingual sentence boundary detection”. CL 32.4 (2006): 485-525.

6 Koehn, Philipp et al. “Moses: Open source toolkit for statistical machine translation”. Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions 25 Jun. 2007: 177-180.

7 Segalovich, I. “A fast morphological algorithm with unknown word guessing induced by a dictionary for a web search engine”. MLMTA. (2003)

8 Makhambetov, O., Makazhanov, A., Sabyrgaliyev, I., Yessenbayev, Z. “Data-driven morphological analysis and disambiguation for Kazakh”. CICLing 2015, 151–163.

9 Espla-Gomis, Miquel, and Mikel Forcada. “Combining content-based and url-based heuristics to harvest aligned bitexts from multilingual sites with bitextor”. The Prague Bulletin of Mathematical Linguistics 93 (2010): 77-86.

10 Washington, Jonathan, Ilnar Salimzyanov, and Francis M Tyers. “Finite-state morphological transducers for three Kypchak languages”. LREC 2014: 3378-3385.