Extending IBM Word Alignment Model 1 with Edit Distance: Results on a Diachronic Parallel Corpus

Siyu Tao
Department of Language Science and Technology
Saarland University
66123 Saarbrücken, Germany
siyutao@coli.uni-saarland.de

Abstract

This paper reports the implementation and results of my final project for the lecture Computational Linguistics (Winter Semester 2020/21) at Saarland University. Two possible extensions to IBM Word Alignment Model 1 are proposed, incorporating edit distance information to utilize known linguistic relatedness when aligning diachronic texts. Experiments and evaluation of these two methods on a diachronic bible corpus show that both achieve better results than the baseline.

1 Introduction

Word alignments form a crucial component in statistical machine translation (SMT) systems. Most influential among the word alignment models are likely the IBM Models, first proposed by Brown et al. (1993). They have not only become the basis of many later works in SMT, but also found use in many other applications.

Given the goal of SMT and the availability of corpora, it comes as no surprise that word alignment models such as the IBM Models are most often used on parallel bi-text corpora of modern languages. There has also been some work exploring the application of multilingual word alignment, most often for the purpose of linguistic and typological comparison of languages (Mayer and Cysouw, 2012; Östling, 2014, *inter alia*). However, very little work has explored the possibility of aligning texts from different diachronic stages of languages, presumably in part due to the lack of diachronic parallel corpora.

The numerous translations and re-translations of the Bible into local languages throughout history make biblical texts one of the few for which the construction of a diachronic parallel corpus are viable. Bouma et al. (2020) has accomplished exactly that in creating the EDGeS Diachronic Bible Corpus, a diachronically and synchronically parallel corpus of Bible translations in four Germanic languages, Dutch, English, German and Swedish, with texts from the 14th century until today. This opens up many opportunities for diachronic linguistic research, especially on structural and syntactic change over time that parallel texts uniquely enable.

Word alignment presents itself as an obvious starting point for research on diachronic parallel texts. While there is little difference between how we formulate the word alignment task on synchronic texts and on diachronic texts, the known relationship between the diachronic stages of languages is something that could conceivably be exploited in the process to our advantage.

In this paper, we preliminarily explore the possibility of extending existing word alignment models to take advantage of such known relationship between these languages or rather stages of languages.

We focus our attention on the first of the IBM models, IBM Model 1, which utilizes only word co-occurrence information from parallel sentences to train the probability of word translation. In other words, each word is treated simply as a token and any linguistic information at a sub-word level, such as cognate relationship, inflection, and word formation, is discarded. A brief definition of Model 1 is provided in §2.

To make use of such information, we propose two methods to extend IBM Model 1 in §3, both using edit distance as a direct measure of orthographic similarity and a shallow proxy for linguistic similarity. In §4, we implement both and evaluate the results on parts of the EDGeS corpus using IBM Model 1 as a baseline. We conclude in §5 and offer an outlook for possible directions of future research.

2 IBM Model 1 Definition

This section only intends to provide a brief summary of the IBM Model 1 definition. Refer to the

original publication (Brown et al., 1993) for details.

In a pair of sentences, **f** and **e**, respectively in the target and source language, each word at position i in the target sentence, f_i, is aligned to at most one and only one word at position j in the source sentence, e_j. Let I, J respectively be the length of the target and source sentences, we have a_i as the alignment variable in the range of $[0, J]$ denoting the index of the word in **e** to which f_i is aligned to. $\mathbf{a} = \langle a_1, \ldots, a_I \rangle$ denotes the array of alignments.

Assuming a simple generative procedure in which the length I is chosen according to the distribution $p(I|J)$, the word position alignments is a uniform distribution $p(a_i = j|J) = \frac{1}{J+1}$, and each target word f_i is translated according to a distribution conditioned on the aligned source word $p(f_i|e_{a_i})$, we then have the product of these probabilities as the joint probability of the target sentence and alignment conditioned on the source sentence:

$$p(\mathbf{f}, \mathbf{a}|\mathbf{e}) = p(I|J) \prod_{i=1}^{I} p\left(a_i|J\right) \cdot p\left(f_i|e_{a_i}\right) \quad (1)$$

The parameter of the model, θ, is defined as two tables of probabilities: $p(I|J)$, $p(f|e)$. Given a dataset $\mathcal{D}$ of N sentence pairs $\mathcal{D} = \left\{\left(\mathbf{f}^{(1)}, \mathbf{e}^{(1)}\right) \ldots \left(\mathbf{f}^{(N)}, \mathbf{e}^{(N)}\right)\right\}$, the Expectation–Maximization (EM) algorithm is then applied to a corpus of parallel source-target sentence pairs, with the goal of getting the parameter $\hat{\theta}$ that maximizes the likelihood of the corpus:

$$\hat{\theta} = \arg\max_{\theta} \prod_{n=1}^{N} p_\theta \left(\mathbf{f}^{(n)}, \mathbf{a}^{(n)} \mid \mathbf{e}^{(n)}\right) \quad (2)$$

3 Extending Model 1 with Edit Distance

We explore two possible methods to integrate edit distance information into IBM Model 1. In §3.1, we use edit distance to heuristically initialize the parameters before the EM training. In §3.2, edit distance is only used in the decoding process after the EM training is completed.

Definition. The **Levenshtein edit distance** (Levenshtein, 1966), **edit distance** for short, considers three possible *edit* operations that can be performed on a string: INSERTION, DELETION, and SUBSTITUTION. The edit distance between two strings is therefore the minimum number of edit operations needed to change one from another. This can be calculated by a simple recursive algorithm but more efficient algorithms have also been introduced (Hyyrö, 2001).

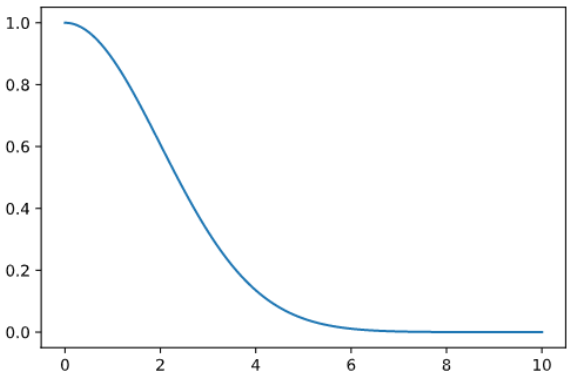

Figure 1: Gaussian function over $[0, 10]$

3.1 Method 1: Heuristic Initialization

Commonly, the initial parameters θ_0, i.e., the translation probability distribution of a target language word given a source language word, is often initialized uniformly. To utilize edit distance, we propose using it as a heuristic to initialize the parameters.

To achieve that, we go over the vocabularies of the target and source languages and calculate the edit distance between each target language word f and each source language word e. We then scale the edit distance using a Gaussian function (3), where we set $\mu = 0, \sigma = 2$. For visual reference, this function is also plotted in Figure 1, where it is shown over the range $[0, 10]$.

$$gaussian(x) = \exp\left(\frac{-(x-\mu)^2}{2\sigma^2}\right) \quad (3)$$

In other words, for all f, e, $f \in F$, $e \in E$, we have the heuristic function $h(f, e)$:

$$h(f, e) = \exp\left(\frac{-edit\_dist(f, e)^2}{8}\right) \quad (4)$$

Because $\forall_{e \in E} \sum_{f \in F} p(f|e) = 1$, we must normalize the heuristic to get a probability distribution. We divide the heuristic for each pair of f_i, e_j by the sum of the heuristics of f with all $e \in E$:

$$p\left(f_i \mid e_j\right) = \frac{h\left(f_i, e_j\right)}{\sum_{e \in E} h\left(f_i, e\right)} \quad (5)$$

We then use the heuristically initialized probability distribution for the EM training and the rest of the standard Model 1 remains unchanged.

An obvious drawback of this method is that the algorithm goes over the entire vocabularies of target and source languages and calculates the edit distance between each word pair. This is clearly inefficient, as the edit distances between words that never co-occur in a sentence pair are presumably of little significance. We expect long computation times as a result of this inefficiency.

3.2 Method 2: Modified Decoding Algorithm

In part answering to the likely inefficiency of the heuristic initialization method, we attempt to be more parsimonious in performing edit distance calculations in the second method. Namely, we propose modifying the decoding algorithm in order to take the edit distance into account, but only at the final alignment stage.

We do so by defining a revised translation probability p' of a target word f given a source word e: the probability $p(f, e)$ as estimated by the EM procedures divided by the edit distance $edit\_dist(f, e)$ add 1 (to avoid division by zero when $edit\_dist(f, e) = 0$).

$$p'(f|e) = \frac{p(f|e)}{edit\_dist(f, e) + 1} \tag{6}$$

The calculations of this revised probability and therefore the calculations of the edit distances are performed in the decoding process. **Algorithm 1** shows the most probable alignment algorithm modified with the revised translation probability.

Algorithm 1: Most Probable Alignment with Edit Distance

for each n in $[1, ..., N]$ **do**
  for each i in $[1, ..., I^{(n)}]$ **do**
    $best\_prob = 0$
    $best\_j = 0$
    for each j in $[1, ..., J^{(n)}]$ **do**
      $p'(f_i^{(n)}|e_j^{(n)}) = \frac{p(f_i^{(n)}|e_j^{(n)})}{e\_d(f_i^{(n)}, e_j^{(n)})+1}$
      if $p'(f_i^{(n)}|e_j^{(n)}) > best\_prob$ **then**
        $best\_prob = p'(f_i^{(n)}|e_j^{(n)})$
        $best\_j = j$
    $align(n, i, best_j)$

This method is significantly more efficient than the heuristic initialization method, as any calculation of edit distance is done *à la carte* when both words appear in one sentence pair being aligned. On the other hand, this also means that the information is not available at the EM training stage. We will see how these two methods compare in the next section.

4 Experiments

4.1 Dataset

We implement the two proposed methods and evaluate them on OpneEDGeS, the open sub-part of the EDGeS Diachronic Bible Coprus. The alignments in EDGeS is done at verse level with the *Nieuwe Bijbel Vertaling* (Dutch, 2004) as the pivot. See Bouma et al. (2020) for a detailed description of their dataset.

Preprocessing. OpenEDGeS provides texts and the bi-text verse-level alignments separately. Our preprocessing starts from an alignment file and retrieves the corresponding lines from the texts and perform a simple regex tokenization. In one-to-many or many-to-many correspondences, the lines are appended together. Notably, we replace non-verse lines such as chapter titles that are handled variably in different translations with a uniform `'<NON-VERSE>'` token. Refer to the accompanying code for details of the preprocessing steps if needed.

4.2 Evaluation

For the purpose of evaluation, given the constraints of time and language competency, we select three pairs of texts with bi-text verse-level alignments, one pair of German texts, one pair of English texts, and one pair of a German text and an English text. The translation editions are listed below:

f-text (target)	e-text (source)
1781 *Rosalino* (de)	1871 *Elberfelder* (de)
1611 *KJV* (en)	1890 *Darby* (en)
1871 *Elberfelder* (de)	1890 *Darby* (en)

Table 1: Text pairs used for evaluation

For each of the text pair, we create a small gold set of alignments as the benchmark, consisting of 15 aligned sentence pairs for each of the text pairs. Samples of the alignments are provided in Table 2.

target sentence	source sentence	alignments
Daß ihr durch ihn in allen Dingen , in allen Worten , und in aller Erkenntniß reich geworden seyd ,	daß ihr in ihm in allem reich gemacht worden seid , in allem Wort der Lehre und aller Erkenntniß ,	0-0 1-1 2-2 3-3 4-4 5-5 6-6 7-10 8-11 9-12 10-13 12-16 13-11 14-17 15-18 16-6 17-8 18-9 19-19
And vnto Eber were borne two sonnes : the name of the one was Peleg , (because in his dayes the earth was diuided) and his brothers name was Ioktan . And Ioktan begate Almodad , and Sheleph , and Hazermaueth , and Ierah ,	And to Eber were born two sons : the name of the one was Peleg , for in his days was the earth divided ; and his brother ' s name was Joktan . And Joktan begot Almodad , and Sheleph , and Hazarmaveth , and Jerah ,	0-0 1-1 2-2 3-3 4-4 5-5 6-6 7-7 8-8 9-9 10-10 11-11 12-12 13-13 14-14 15-15 17-16 18-17 19-18 20-19 21-21 22-22 23-20 24-23 26-25 27-26 28-27 29-30 30-31 31-32 32-33 33-34 34-35 35-36 36-37 37-38 38-39 39-40 40-41 41-42 42-43 43-44 44-45 45-46 46-47
Und Kusch zeugte Nimrod , selbiger fing an , gewaltig zu werden auf der Erde .	And Cush begot Nimrod : he began to be mighty on the earth .	0-0 1-1 2-2 3-3 4-4 5?5 6-6 7-6 9-9 10-7 11-8 12-10 13-11 14-12 15-13

Table 2: Sample of gold alignment labels from each of the three pairs of texts used in evaluation

4.3 Results

The results of our two modified models along with the baseline are reported in terms of precision, recall, and alignment error rate (AER), as defined by Och and Ney (2003). Their definitions are as follows:

$$\text{recall} = \frac{|A \cap S|}{|S|} \tag{7}$$

$$\text{precision} = \frac{|A \cap P|}{|A|} \tag{8}$$

$$\text{AER} = 1 - \frac{|A \cap S| + |A \cap P|}{|A| + |S|} \tag{9}$$

where S denotes the set of sure alignments in the annotation, P denotes the set of possible alignments in the annotation, and A denotes the set of alignments produced by the model. The results are presented in Table 3.

We observe that both methods consistently though modestly outperform the baseline, with the heuristic method performing better in both text pairs involving German, and the modified decoding method performing better in the English pair. All results are consistent across the three metrics used.

However, in line with our expectation, due to the initialization process, the heuristic method takes significantly longer than the modified decoding to train (30+ minutes vs. ∼3 minutes in our test).

		precision	recall	AER
Baseline	de-de	0.541	0.613	0.425
	en-en	0.541	0.567	0.446
	de-en	0.597	0.611	0.396
ED Heuristic	de-de	**0.579**	**0.657**	**0.384**
	en-en	0.591	0.619	0.395
	de-en	**0.658**	**0.673**	**0.334**
ED Decoding	de-de	0.550	0.623	0.416
	en-en	**0.607**	**0.637**	**0.378**
	de-en	0.646	0.661	0.346

Table 3: Results (precision, recall, AER) of the two modified models and baseline on the evaluation set

The modest improvement it provides over the modified decoding method in some cases is clearly not enough to compensate the significant longer training time.

5 Conclusion and Future Work

In conclusion, we report modest but positive results from our preliminary experiments with extending IBM Model 1 with edit distance information. This shows that existing statistical word alignment models could benefit from integrating this orthographic similarity measure and possibly other measures of linguistic similarity.

There are several directions for possible future

research. Most directly, less naïve methods to integrate edit distance in the heuristic initialization should be considered, as well as more linguistically informed measures of sequence similarity such as that proposed by List et al. (2018). Edit distance and other possible orthographic measures could also conceivably be utilized on their own to study orthographic change and standardization. Being preliminary in nature, our experiments only focus on Model 1 but it should be possible to extend the methods to other IBM Models, among others. The alignments may also potentially be used to enable certain annotation transfers like in Östling (2015), allowing for more in-depth study of structural change in the languages.

References

Gerlof Bouma, Evie Coussé, Trude Dijkstra, and Nicoline van der Sijs. 2020. The EDGeS Diachronic Bible Corpus. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 5232–5239, Marseille, France. European Language Resources Association.

Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. 1993. The Mathematics of Statistical Machine Translation: Parameter Estimation. *Computational Linguistics*, 19(2):263–311.

Heikki Hyyrö. 2001. Explaining and extending the bit-parallel approximate string matching algorithm of Myers. Technical report, Citeseer.

Vladimir I Levenshtein. 1966. Binary codes capable of correcting deletions, insertions, and reversals. In *Soviet Physics Doklady*, volume 10, pages 707–710. Soviet Union.

Johann-Mattis List, Mary Walworth, Simon J Greenhill, Tiago Tresoldi, and Robert Forkel. 2018. Sequence comparison in computational historical linguistics. *Journal of Language Evolution*, 3(2):130–144.

Thomas Mayer and Michael Cysouw. 2012. Language comparison through sparse multilingual word alignment. In *Proceedings of the EACL 2012 Joint Workshop of LINGVIS & UNCLH*, pages 54–62, Avignon, France. Association for Computational Linguistics.

Franz Josef Och and Hermann Ney. 2003. A Systematic Comparison of Various Statistical Alignment Models. *Computational Linguistics*, 29(1):19–51.

Robert Östling. 2014. Bayesian Word Alignment for Massively Parallel Texts. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, Volume 2: Short Papers*, pages 123–127, Gothenburg, Sweden. Association for Computational Linguistics.

Robert Östling. 2015. Word Order Typology through Multilingual Word Alignment. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 205–211, Beijing, China. Association for Computational Linguistics.