# Combining Word-Level and Character-Level Models for Machine Translation Between Closely-Related Languages

**Preslav Nakov**
Qatar Computing Research Institute
Qatar Foundation, P.O. box 5825
Doha, Qatar
pnakov@qf.org.qa

**Jörg Tiedemann**
Department of Linguistics and Philology
Uppsala University
Uppsala, Sweden
jorg.tiedemann@lingfil.uu.se

## Abstract

We propose several techniques for improving statistical machine translation between closely-related languages with scarce resources. We use character-level translation trained on $n$-gram-character-aligned bitexts and tuned using word-level BLEU, which we further augment with character-based transliteration at the word level and combine with a word-level translation model. The evaluation on Macedonian-Bulgarian movie subtitles shows an improvement of 2.84 BLEU points over a phrase-based word-level baseline.

## 1 Introduction

Statistical machine translation (SMT) systems, require parallel corpora of sentences and their translations, called *bitexts*, which are often not sufficiently large. However, for many *closely-related* languages, SMT can be carried out even with small bitexts by exploring relations below the word level.

Closely-related languages such as Macedonian and Bulgarian exhibit a large overlap in their vocabulary and strong syntactic and lexical similarities. Spelling conventions in such related languages can still be different, and they may diverge more substantially at the level of morphology. However, the differences often constitute consistent regularities that can be generalized when translating.

The language similarities and the regularities in morphological variation and spelling motivate the use of character-level translation models, which were applied to translation (Vilar et al., 2007; Tiedemann, 2009a) and transliteration (Matthews, 2007).

| Macedonian | Bulgarian |
|---|---|
| а в м е _ | а х м е _ |
| а в м е _ д а _ | а х м е _ д а _ |
| _ в е р у в а м _ | _ в я р в а м _ |
| _ д е к а _ т о ј _ | _ , ч е _ т о й _ |

Table 1: Examples from a character-level phrase table (without scores): mappings can cover words and phrases.

Certainly, translation cannot be adequately modeled as simple transliteration, even for closely-related languages. However, the strength of phrase-based SMT (Koehn et al., 2003) is that it can support rather large sequences (phrases) that capture translations of entire chunks. This makes it possible to include mappings that go far beyond the edit-distance-based string operations usually modeled in transliteration. Table 1 shows how character-level phrase tables can cover mappings spanning over multi-word units. Thus, character-level phrase-based SMT models combine the generality of character-by-character transliteration and lexical mappings of larger units that could possibly refer to morphemes, words or phrases, as well as to various combinations thereof.

## 2 Training Character-level SMT Models

We treat sentences as sequences of characters instead of words, as shown in Figure 1. Due to the reduced vocabulary, we can use higher-order models, which is necessary in order to avoid the generation of non-word sequences. In our case, we opted for a 10-character language model and a maximum phrase length of 10 (based on initial experiments).

However, word alignment models are not fit for character-level SMT, where the vocabulary shrinks.



*Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, pages 301–305,
Jeju, Republic of Korea, 8-14 July 2012. 

*original:*
MK: навистина ?
BG: наистина ?

*characters:*
MK: н а в и с т и н а _ ?
BG: н а и с т и н а _ ?

*character bigrams:*
MK: на ав ви ис ст ти ин на а_ _? ?_
BG: на аи ис ст ти ин на а_ _? ?_

Figure 1: Preparing the training corpus for alignment.

Statistical word alignment models heavily rely on context-independent lexical translation parameters and, therefore, are unable to properly distinguish character mapping differences in various contexts. The alignment models used in the transliteration literature have the same problem as they are usually based on edit distance operations and finite-state automata without contextual history (Jiampojamarn et al., 2007; Damper et al., 2005; Ristad and Yianilos, 1998). We, thus, transformed the input to sequences of character $n$-grams as suggested by Tiedemann (2012); examples are shown in Figure 1. This artificially increases the vocabulary as shown in Table 2, making standard alignment models and their lexical translation parameters more expressive.

| | **Macedonian** | **Bulgarian** |
|---|---|---|
| single characters | 99 | 101 |
| character bigrams | 1,851 | 1,893 |
| character trigrams | 13,794 | 14,305 |
| words | 41,816 | 30,927 |

Table 2: Vocabulary size of character-level alignment models and the corresponding word-level model.

It turns out that bigrams constitute a good compromise between generality and contextual specificity, which yields useful character alignments with good performance in terms of phrase-based translation. In our experiments, we used GIZA++ (Och and Ney, 2003) with standard settings and the *grow-diagonal-final-and* heuristics to symmetrize the final IBM-model-4-based Viterbi alignments (Brown et al., 1993). The phrases were extracted and scored using the Moses training tools (Koehn et al., 2007).[1]

We tuned the parameters of the log-linear SMT model using minimum error rate training (Och, 2003), optimizing BLEU (Papineni et al., 2002). Since BLEU over matching character sequences does not make much sense, especially if the $k$-gram size is limited to small values of $k$ (usually, 4 or less), we post-processed $n$-best lists in each tuning step to calculate the usual word-based BLEU score.

[1]Note that the extracted phrase table does not include sequences of character $n$-grams. We map character $n$-gram alignments to links between single characters before extraction.

## 3 Transliteration

We also built a character-level SMT system for word-level transliteration, which we trained on a list of automatically extracted pairs of likely cognates.

### 3.1 Cognate Extraction

Classic NLP approaches to cognate extraction look for words with similar spelling that co-occur in parallel sentences (Kondrak et al., 2003). Since our Macedonian-Bulgarian bitext (MK–BG) was small, we further used a MK–EN and an EN–BG bitext.

First, we induced IBM-model-4 word alignments for MK–EN and EN–BG, from which we extracted four conditional lexical translation probabilities: $\Pr(m|e)$ and $\Pr(e|m)$ for MK–EN, and $\Pr(b|e)$ and $\Pr(e|b)$ for EN–BG, where $m$, $e$, and $b$ stand for a Macedonian, an English, and a Bulgarian word.

Then, following (Callison-Burch et al., 2006; Wu and Wang, 2007; Utiyama and Isahara, 2007), we induced conditional lexical translation probabilities as $\Pr(m|b) = \sum_e \Pr(m|e)\Pr(e|b)$, where $\Pr(m|e)$ and $\Pr(e|b)$ are estimated using maximum likelihood from MK–EN and EN–BG word alignments.

Then, we induced translation probability estimations for the reverse direction $\Pr(b|m)$ and we calculated the quantity $\mathrm{Piv}(m,b) = \Pr(m|b)\Pr(b|m)$. We calculated a similar quantity $\mathrm{Dir}(m,b)$, where the probabilities $\Pr(m|b)$ and $\Pr(b|m)$ are estimated using maximum likelihood from the MK–BG bitext directly. Finally, we calculated the similarity score $S(m,b) = \mathrm{Piv}(m,b) + \mathrm{Dir}(m,b) + 2\times\mathrm{LCSR}(m,b)$, where LCSR is the longest common subsequence of two strings, divided by the length of the longer one.

The score $S(m,b)$ is high for words that are likely to be cognates, i.e., that (i) have high probability of being mutual translations, which is expressed by the first two terms in the summation, and (ii) have similar spelling, as expressed by the last term. Here we give equal weight to $\mathrm{Dir}(m,b)$ and $\mathrm{Piv}(m,b)$; we also give equal weights to the translational similarity (the sum of the first two terms) and to the spelling similarity (twice LCSR).

We excluded all words of length less than three, as well as all Macedonian-Bulgarian word pairs $(m, b)$ for which $\mathrm{Piv}(m, b) + \mathrm{Dir}(m, b) < 0.01$, and those for which $\mathrm{LCSR}(m, b)$ was below 0.58, a value found by Kondrak et al. (2003) to work well for a number of European language pairs.

Finally, using $S(m, b)$, we induced a weighted bipartite graph, and we performed a greedy approximation to the maximum weighted bipartite matching in that graph using *competitive linking* (Melamed, 2000), to produce the final list of cognate pairs.

Note that the above-described cognate extraction algorithm has three important components: (1) orthographic, based on LCSR, (2) semantic, based on word alignments and pivoting over English, and (3) competitive linking. The orthographic component is essential when looking for cognates since they must have similar spelling by definition, while the semantic component prevents the extraction of false friends like вреден, which means 'valuable' in Macedonian but 'harmful' in Bulgarian. Finally, competitive linking helps prevent issues related to word inflection that cannot be handled using the semantic component alone.

### 3.2 Transliteration Training

For each pair in the list of cognate pairs, we added spaces between any two adjacent letters for both words, and we further appended special start and end characters. We split the resulting list into training, development and testing parts and we trained and tuned a character-level Macedonian-Bulgarian phrase-based monotone SMT system similar to that in (Finch and Sumita, 2008; Tiedemann and Nabende, 2009; Nakov and Ng, 2009; Nakov and Ng, 2012). The system used a character-level Bulgarian language model trained on words. We set the maximum phrase length and the language model order to 10, and we tuned the system using MERT.

### 3.3 Transliteration Lattice Generation

Given a Macedonian sentence, we generated a lattice where each input Macedonian word of length three or longer was augmented with Bulgarian alternatives: $n$-best transliterations generated by the above character-level Macedonian-Bulgarian SMT system (after the characters were concatenated to form a word and the special symbols were removed).

In the lattice, we assigned the original Macedonian word the weight of 1; for the alternatives, we assigned scores between 0 and 1 that were the sum of the translation model probabilities of generating each alternative (the sum was needed since some options appeared multiple times in the $n$-best list).

## 4 Experiments and Evaluation

For our experiments, we used translated movie subtitles from the OPUS corpus (Tiedemann, 2009b). For Macedonian-Bulgarian there were only about 102,000 aligned sentences containing approximately 1.3 million tokens altogether. There was substantially more monolingual data available for Bulgarian: about 16 million sentences containing ca. 136 million tokens.

However, this data was noisy. Thus, we realigned the corpus using `hunalign` and we removed some Bulgarian files that were misclassified as Macedonian and vice versa, using a BLEU-filter. Furthermore, we also removed sentence pairs containing language-specific characters on the wrong side. From the remaining data we selected 10,000 sentence pairs (roughly 128,000 words) for development and another 10,000 (ca. 125,000 words) for testing; we used the rest for training.

The evaluation results are summarized in Table 3.

| MK→BG | | BLEU % | NIST | TER | METEOR |
|---|---|---|---|---|---|
| **Transliteration** | | | | | |
| | no translit. | 10.74 | 3.33 | 67.92 | 60.30 |
| t1 | letter-based | 12.07 | 3.61 | 66.42 | 61.87 |
| t2 | cogn.+lattice | 22.74 | 5.51 | 55.99 | 66.42 |
| **Word-level SMT** | | | | | |
| w0 | Apertium | 21.28 | 5.27 | 56.92 | 66.35 |
| w1 | SMT baseline | **31.10** | 6.56 | 50.72 | 70.53 |
| w2 | w1 + t1-lattice | **32.19**$^{(+1.19)}$ | 6.76 | 49.68 | 71.18 |
| **Character-level SMT** | | | | | |
| c1 | char-aligned | **32.28**$^{(+1.18)}$ | 6.70 | 49.70 | 71.35 |
| c2 | bigram-aligned | **32.71**$^{(+1.61)}$ | 6.77 | 49.23 | 71.65 |
| | trigram-aligned | 32.07$^{(+0.97)}$ | 6.68 | 49.82 | 71.21 |
| **System combination** | | | | | |
| | w2 + c2 | 32.92$^{(+1.82)}$ | 6.90 | 48.73 | 71.71 |
| | w1 + c2 | 33.31$^{(+2.21)}$ | 6.91 | 48.60 | 71.81 |
| **Merged phrase tables** | | | | | |
| m1 | w1 + c2 | **33.33**$^{(+2.13)}$ | 6.86 | 48.86 | 71.73 |
| m2 | w2 + c2 | **33.94**$^{(+2.84)}$ | 6.89 | 48.99 | 71.76 |

Table 3: **Macedonian-Bulgarian translation and transliteration.** Superscripts show the absolute improvement in BLEU compared to the word-level baseline (w1).

**Transliteration.** The top rows of Table 3 show the results for Macedonian-Bulgarian transliteration. First, we can see that the BLEU score for the original Macedonian testset evaluated against the Bulgarian reference is 10.74, which is quite high and reflects the similarity between the two languages. The next line (t1) shows that many differences between Macedonian and Bulgarian stem from mere differences in orthography: we mapped the six letters in the Macedonian alphabet that do not exist in the Bulgarian alphabet to corresponding Bulgarian letters and letter sequences, gaining over 1.3 BLEU points. The following line (t2) shows the results using the sophisticated transliteration described in Section 3, which takes two kinds of context into account: (1) word-internal letter context, and (2) sentence-level word context. We generated a lattice for each Macedonian test sentence, which included the original Macedonian words and the 1-best[2] Bulgarian transliteration option from the character-level transliteration model. We then decoded the lattice using a Bulgarian language model; this increased BLEU to 22.74.

**Word-level translation.** Naturally, lattice-based transliteration cannot really compete against standard word-level translation (w1), which is better by 8 BLEU points. Still, as line (w2) shows, using the 1-best transliteration lattice as an input to (w1) yields[3] consistent improvement over (w1) for four evaluation metrics: BLEU (Papineni et al., 2002), NIST v. 13, TER (Snover et al., 2006) v. 0.7.25, and METEOR (Lavie and Denkowski, 2009) v. 1.3. The baseline system is also significantly better than the on-line version of Apertium (http://www.apertium.org/), a shallow transfer-rule-based MT system that is optimized for closely-related languages (accessed on 2012/05/02). Here, Apertium suffers badly from a large number of unknown words in our testset (ca. 15%).

**Character-level translation.** Moving down to the next group of experiments in Table 3, we can see that standard character-level SMT (c1), i.e., simply treating characters as separate words, performs significantly better than word-level SMT. Using bigram-based character alignments yields further improvement of +0.43 BLEU.

**System combination.** Since word-level and character-level models have different strengths and weaknesses, we further tried to combine them. We used MEMT, a state-of-the-art Multi-Engine Machine Translation system (Heafield and Lavie, 2010), to combine the outputs of (c3) with the output of (w1) and of (w2). Both combinations improved over the individual systems, but (w1)+(c2) performed better, by +0.6 BLEU points over (c2).

**Combining word-level and phrase-level SMT.** Finally, we also combined (w1) with (c3) in a more direct way: by merging their phrase tables. First, we split the phrases in the word-level phrase tables of (w1) to characters as in character-level models. Then, we generated four versions of each phrase pair: with/without "_" at the beginning/end of the phrase. Finally, we merged these phrase pairs with those in the phrase table of (c3), adding two extra features indicating each phrase pair's origin: the first/second feature is 1 if the pair came from the first/second table, and 0.5 otherwise. This combination outperformed MEMT, probably because it expands the search space of the SMT system more directly. We further tried scoring with two language models in the process of translation, character-based and word-based, but we did not get consistent improvements. Finally, we experimented with a 1-best character-level lattice input that encodes the same options and weights as for (w2). This yielded our best overall BLEU score of 33.94, which is +2.84 BLEU points of absolute improvement over the (w1) baseline, and +1.23 BLEU points over (c2).[4]

## 5 Conclusion and Future Work

We have explored several combinations of character- and word-level translation models for translating between closely-related languages with scarce resources. In future work, we want to use such a model for pivot-based translations from the resource-poor language (Macedonian) to other languages (such as English) via the related language (Bulgarian).

## Acknowledgments

The research is partially supported by the EU ICT PSP project LetsMT!, grant number 250456.

[2]Using 3/5/10/100-best made very little difference.

[3]The decoder can choose between (a) translating a Macedonian word and (b) using its 1-best Bulgarian transliteration.

[4]All improvements over (w1) in Table 3 that are greater or equal to 0.97 BLEU points are statistically significant according to Collins' sign test (Collins et al., 2005).

## References


Peter Brown, Vincent Della Pietra, Stephen Della Pietra, and Robert Mercer. 1993. The mathematics of statistical machine translation: parameter estimation. *Computational Linguistics*, 19(2):263–311.

Chris Callison-Burch, Philipp Koehn, and Miles Osborne. 2006. Improved statistical machine translation using paraphrases. In *Proceedings of HLT-NAACL '06*, pages 17–24, New York, NY.

Michael Collins, Philipp Koehn, and Ivona Kučerová. 2005. Clause restructuring for statistical machine translation. In *Proceedings of ACL '05*, pages 531–540, Ann Arbor, MI.

Robert Damper, Yannick Marchand, John-David Marsters, and Alex Bazin. 2005. Aligning text and phonemes for speech technology applications using an EM-like algorithm. *International Journal of Speech Technology*, 8(2):149–162.

Andrew Finch and Eiichiro Sumita. 2008. Phrase-based machine transliteration. In *Proceedings of the Workshop on Technologies and Corpora for Asia-Pacific Speech Translation*, pages 13–18, Hyderabad, India.

Kenneth Heafield and Alon Lavie. 2010. Combining machine translation output with open source: The Carnegie Mellon multi-engine machine translation scheme. *The Prague Bulletin of Mathematical Linguistics*, 93(1):27–36.

Sittichai Jiampojamarn, Grzegorz Kondrak, and Tarek Sherif. 2007. Applying many-to-many alignments and hidden Markov models to letter-to-phoneme conversion. In *Proceedings of NAACL-HLT '07*, pages 372–379, Rochester, New York.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proceedings of NAACL '03*, pages 48–54, Edmonton, Canada.

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, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of ACL '07*, pages 177–180, Prague, Czech Republic.

Grzegorz Kondrak, Daniel Marcu, and Kevin Knight. 2003. Cognates can improve statistical translation models. In *Proceedings of NAACL '03*, pages 46–48, Edmonton, Canada.

Alon Lavie and Michael Denkowski. 2009. The Meteor metric for automatic evaluation of machine translation. *Machine Translation*, 23:105–115.

David Matthews. 2007. Machine transliteration of proper names. Master's thesis, School of Informatics, University of Edinburgh, Edinburgh, UK.

Dan Melamed. 2000. Models of translational equivalence among words. *Computational Linguistics*, 26(2):221–249.

Preslav Nakov and Hwee Tou Ng. 2009. Improved statistical machine translation for resource-poor languages using related resource-rich languages. In *Proceedings of EMNLP '09*, pages 1358–1367, Singapore.

Preslav Nakov and Hwee Tou Ng. 2012. Improving statistical machine translation for a resource-poor language using related resource-rich languages. *Journal of cial Intelligence Research*, 44.

Franz Josef Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In *Proceedings of ACL '03*, pages 160–167, Sapporo, Japan.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of ACL '02*, pages 311–318, Philadelphia, PA.

Eric Ristad and Peter Yianilos. 1998. Learning string edit distance. *IEEE Transactions on Pattern Recognition and Machine Intelligence*, 20(5):522–532.

Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proceedings of AMTA '06*, pages 223–231.

Jörg Tiedemann and Peter Nabende. 2009. Translating transliterations. *International Journal of Computing and ICT Research*, 3(1):33–41.

Jörg Tiedemann. 2009a. Character-based PSMT for closely related languages. In *Proceedings of EAMT '09*, pages 12–19, Barcelona, Spain.

Jörg Tiedemann. 2009b. News from OPUS - A collection of multilingual parallel corpora with tools and interfaces. In *Recent Advances in Natural Language Processing*, volume V, pages 237–248. John Benjamins.

Jörg Tiedemann. 2012. Character-based pivot translation for under-resourced languages and domains. In *Proceedings of EACL '12*, pages 141–151, Avignon, France.

Masao Utiyama and Hitoshi Isahara. 2007. A comparison of pivot methods for phrase-based statistical machine translation. In *Proceedings of NAACL-HLT '07*, pages 484–491, Rochester, NY.

David Vilar, Jan-Thorsten Peter, and Hermann Ney. 2007. Can we translate letters? In *Proceedings of WMT '07*, pages 33–39, Prague, Czech Republic.

Hua Wu and Haifeng Wang. 2007. Pivot language approach for phrase-based statistical machine translation. *Machine Translation*, 21(3):165–181.