A Hybrid Morpheme-Word Representation for Machine Translation of Morphologically Rich Languages*

Minh-Thang Luong **Preslav Nakov** **Min-Yen Kan**
Department of Computer Science
National University of Singapore
13 Computing Drive
Singapore 117417
{luongmin,nakov,kanmy}@comp.nus.edu.sg

Abstract

We propose a language-independent approach for improving statistical machine translation for morphologically rich languages using a hybrid morpheme-word representation where the basic unit of translation is the morpheme, but word boundaries are respected at all stages of the translation process. Our model extends the classic phrase-based model by means of (1) word boundary-aware morpheme-level phrase extraction, (2) minimum error-rate training for a morpheme-level translation model using word-level BLEU, and (3) joint scoring with morpheme- and word-level language models. Further improvements are achieved by combining our model with the classic one. The evaluation on English to Finnish using *Europarl* (714K sentence pairs; 15.5M English words) shows statistically significant improvements over the classic model based on BLEU and human judgments.

1 Introduction

The fast progress of statistical machine translation (SMT) has boosted translation quality significantly. While research keeps diversifying, *the word* remains the atomic token-unit of translation. This is fine for languages with limited morphology like English and French, or no morphology at all like Chinese, but it is inadequate for morphologically rich languages like Arabic, Czech or Finnish (Lee, 2004; Goldwater and McClosky, 2005; Yang and Kirchhoff, 2006).

There has been a line of recent SMT research that incorporates morphological analysis as part of the translation process, thus providing access to the information within the individual words. Unfortunately, most of this work either relies on language-specific tools, or only works for very small datasets.

Below we propose a language-independent approach to SMT of morphologically rich languages using a hybrid morpheme-word representation where the basic unit of translation is the morpheme, but word boundaries are respected at all stages of the translation process. We use unsupervised morphological analysis and we incorporate its output into the process of translation, as opposed to relying on pre-processing and post-processing only as has been done in previous work.

The remainder of the paper is organized as follows. Section 2 reviews related work. Sections 3 and 4 present our morphological and phrase merging enhancements. Section 5 describes our experiments, and Section 6 analyzes the results. Finally, Section 7 concludes and suggests directions for future work.

2 Related Work

Most previous work on morphology-aware approaches relies heavily on language-specific tools, e.g., the *TreeTagger* (Schmid, 1994) or the *Buckwalter* Arabic Morphological Analyzer (Buckwalter, 2004), which hampers their portability to other languages. Moreover, the prevalent method for incorporating morphological information is by heuristically-driven pre- or post-processing. For example, Sadat and Habash (2006) use different combinations of Arabic pre-processing schemes

*This research was sponsored in part by CSIDM (grant # 200805) and by a National Research Foundation grant entitled “Interactive Media Search” (grant # R-252-000-325-279).

for Arabic-English SMT, whereas Oflazer and El-Kahlout (2007) post-processes Turkish morpheme-level translations by re-scoring n-best lists with a word-based language model. These systems, however, do not attempt to incorporate their analysis as part of the decoding process, but rather rely on models designed for word-token translation.

We should also note the importance of the translation direction: it is much harder to translate from a morphologically poor to a morphologically rich language, where morphological distinctions not present in the source need to be generated in the target language. Research in translating into morphologically rich languages, has attracted interest for languages like *Arabic* (Badr et al., 2008), *Greek* (Avramidis and Koehn, 2008), *Hungarian* (Novák, 2009; Koehn and Haddow, 2009), *Russian* (Toutanova et al., 2008), and *Turkish* (Oflazer and El-Kahlout, 2007). These approaches, however, either only succeed in enhancing the performance for small bi-texts (Badr et al., 2008; Oflazer and El-Kahlout, 2007), or improve only modestly for large bi-texts[1].

3 Morphological Enhancements

We present a morphologically-enhanced version of the classic phrase-based SMT model (Koehn et al., 2003). We use a hybrid morpheme-word representation where the basic unit of translation is the morpheme, but word boundaries are respected at all stages of the translation process. This is in contrast with previous work, where morphological enhancements are typically performed as pre-/post-processing steps only.

In addition to changing the basic translation token unit from a word to a morpheme, our model extends the phrase-based SMT model with the following:

1. word boundary-aware morpheme-level phrase extraction;
2. minimum error-rate training for a morpheme-level model using word-level BLEU;
3. joint scoring with morpheme- and word-level language models.

We first introduce our morpheme-level representation, and then describe our enhancements.

3.1 Morphological Representation

Our morphological representation is based on the output of an unsupervised morphological analyzer. Following Virpioja et al. (2007), we use *Morfessor*, which is trained on raw tokenized text (Creutz and Lagus, 2007). The tool segments words into morphemes annotated with the following labels: PRE (prefix), STM (stem), SUF (suffix). Multiple prefixes and suffixes can be proposed for each word; word compounding is allowed as well. The output can be described by the following regular expression:

$$\texttt{WORD} = (\ \texttt{PRE* STM SUF*}\ )^{+}$$

For example, uncarefully is analyzed as

un/PRE+ care/STM+ ful/SUF+ ly/SUF

The above token sequence forms the input to our system. We keep the PRE/STM/SUF tags as part of the tokens, and distinguish between care/STM+ and care/STM. Note also that the "+" sign is appended to each nonfinal tag so that we can distinguish word-internal from word-final morphemes.

3.2 Word Boundary-aware Phrase Extraction

The core translation structure of a phrase-based SMT model is the *phrase table*, which is learned from a bilingual parallel sentence-aligned corpus, typically using the alignment template approach (Och and Ney, 2004). It contains a set of bilingual phrase pairs, each associated with five scores: forward and backward phrase translation probabilities, forward and backward lexicalized translation probabilities, and a constant phrase penalty.

The maximum phrase length n is normally limited to seven words; higher values of n increase the table size exponentially without actually yielding performance benefit (Koehn et al., 2003). However, things are different when translating with morphemes, for two reasons: (1) morpheme-token phrases of length n can span less than n words; and (2) morpheme-token phrases may only partially span words.

The first point means that morpheme-token phrase pairs span fewer word tokens, and thus cover a smaller context, which may result in fewer total extracted pairs compared to a word-level approach. Figure 1 shows a case where three Finnish words consist of nine morphemes. Previously, this issue was addressed by simply increasing the value of n when using morphemes, which is of limited help.

[1] Avramidis and Koehn (2008) improved by 0.15 BLEU over a 18.05 English-Greek baseline; Toutanova et al. (2008) improved by 0.72 BLEU over a 36.00 English-Russian baseline.

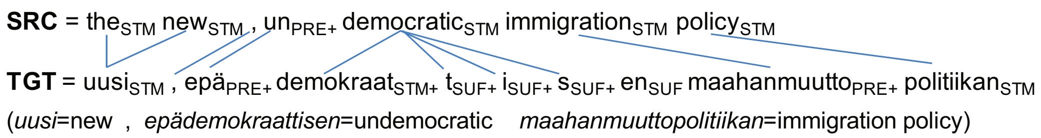
SRC = the$_{STM}$ new$_{STM}$, un$_{PRE+}$ democratic$_{STM}$ immigration$_{STM}$ policy$_{STM}$
TGT = uusi$_{STM}$, epä$_{PRE+}$ demokraat$_{STM+}$ t$_{SUF+}$ i$_{SUF+}$ s$_{SUF+}$ en$_{SUF}$ maahanmuutto$_{PRE+}$ politiikan$_{STM}$
(*uusi*=new , *epädemokraattisen*=undemocratic *maahanmuuttopolitiikan*=immigration policy)

Figure 1: **Example of English-Finnish bilingual fragments morphologically segmented by *Morfessor*.** Solid links represent IBM Model 4 alignments at the morpheme-token level. Translation glosses for Finnish are given below.

The second point is more interesting: morpheme-level phrases may span words partially, making them potentially usable in translating unknown inflected forms of known source language words, but also creates the danger of generating sequences of morphemes that are not legal target language words.

For example, let us consider the phrase in Figure 1: $\mathtt{un}_{\mathtt{PRE+}}$ $\mathtt{democratic}_{\mathtt{STM}}$. The original algorithm will extract the spurious phrase $\mathtt{epä}_{\mathtt{PRE+}}$ $\mathtt{demokraat}_{\mathtt{STM+}}$ $\mathtt{t}_{\mathtt{SUF+}}$ $\mathtt{i}_{\mathtt{SUF+}}$ $\mathtt{s}_{\mathtt{SUF+}}$, beside the correct one that has $\mathtt{en}_{\mathtt{SUF}}$ appended at the end. Such a spurious phrase does not generally help in translating unknown inflected forms, especially for morphologically-rich languages that feature multiple affixes, but negatively affects the translation model in terms of complexity and quality.

We solve both problems by modifying the phrase-pair extraction algorithm so that morpheme-token phrases can extend longer than n, as long as they span n words or less. We further require that word boundaries be respected[2], i.e., morpheme-token phrases span a sequence of whole words. This is a fair extension of the morpheme-token system with respect to a word-token one since both are restricted to span up to n word-tokens.

3.3 Morpheme-Token MERT Optimizing Word-Token BLEU

Modern phrase-based SMT systems use a log-linear model with the following typical feature functions: language model probabilities, word penalty, distortion cost, and the five parameters from the phrase table. Their weights are set by optimizing BLEU score (Papineni et al., 2001) directly using minimum error rate training (MERT), as suggested by Och (2003).

In previous work, phrase-based SMT systems using morpheme-token input/output naturally performed MERT at the morpheme-token level as well. This is not optimal since the final expected system output is a sequence of words, not morphemes. The main danger is that optimizing a morpheme-token BLEU score could lead to a suboptimal weight for the word penalty feature function: this is because the brevity penalty of BLEU is calculated with respect to the number of morphemes, which may vary for sentences with an identical number of words.

This motivates us to perform MERT at the word-token level, although our input consists of morphemes. In particular, for each iteration of MERT, as soon as the decoder generates a morpheme-token translation for a sentence, we convert it into a word-token sequence, which is used to calculate BLEU. We thus achieve MERT optimization at the word-token level while translating a morpheme-token input and generating a morpheme-token output.

3.4 Scoring with Twin Language Models

An SMT system that takes morpheme-token input and generates morpheme-token output should naturally use a morpheme-token language model (LM). This has the advantage of alleviating the problem of data sparseness, especially when translating into a morphologically rich language, since the LM would be able to handle some new unseen inflected forms of known words. On the negative side, a morpheme-token LM spans fewer word-tokens and thus has a more limited word "horizon" compared to one operating at the word level. As with the maximum phrase length, mechanically increasing the order of the morpheme-token LM has a limited impact.

In order to address the issue in a more principled manner, we enhance our model with a second LM that works at the word-token level. This LM is used together with the morpheme-token LM, which is achieved by using two separate feature functions in the log-linear SMT model: one for each LM. We further had to modify the Moses decoder so that

[2] This means that we miss the opportunity to generate new wordforms for known baseforms, but removes the problem of proposing nonwords in the target language.

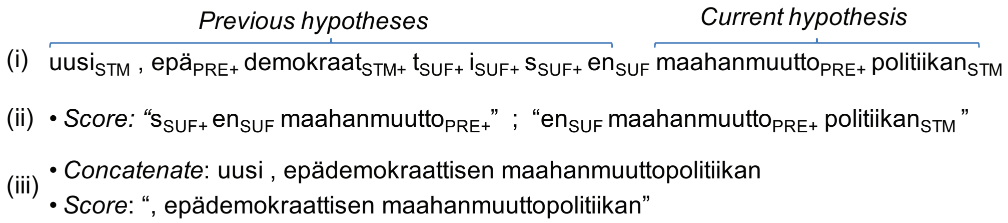
Previous hypotheses
Current hypothesis
(i) uusiSTM , epäPRE+ demokraatSTM+ tSUF+ iSUF+ sSUF+ enSUF maahanmuuttoPRE+ politiikanSTM
(ii) • Score: "sSUF+ enSUF maahanmuuttoPRE+" ; "enSUF maahanmuuttoPRE+ politiikanSTM"
(iii) • Concatenate: uusi , epädemokraattisen maahanmuuttopolitiikan
• Score: ", epädemokraattisen maahanmuuttopolitiikan"

Figure 2: **Scoring with twin LMs.** Shown are: (i) The current state of the decoding process with the target phrases covered by the current partial hypotheses. (ii, iii) Scoring with 3-gram morpheme-token and 3-gram word-token LMs, respectively. For the word-token LM, the morpheme-token sequence is concatenated into word-tokens before scoring.

it can be enhanced with an appropriate word-token "view" on the partial morpheme-level hypotheses[3].

The interaction of the twin LMs is illustrated in Figure 2. The word-token LM can capture much longer phrases and more complete contexts such as "*, epädemokraattisen maahanmuuttopolitiikan*" compared to the morpheme-token LM.

Note that scoring with two LMs that see the output sequence as different numbers of tokens is not readily offered by the existing SMT decoders. For example, the phrase-based model in Moses (Koehn et al., 2007) allows scoring with multiple LMs, but assumes they use the same token granularity, which is useful for LMs trained on different monolingual corpora, but cannot handle our case. While the factored translation model (Koehn and Hoang, 2007) in Moses does allow scoring with models of different granularity, e.g., lemma-token and word-token LMs, it requires a 1:1 correspondence between the tokens in the different factors, which clearly is not our case.

Note that scoring with twin LMs is conceptually superior to n-best re-scoring with a word-token LM, e.g., (Oflazer and El-Kahlout, 2007), since it is tightly integrated into decoding: it scores partial hypotheses and influenced the search process directly.

4 Enriching the Translation Model

Another general strategy for combining evidence from the word-token and the morpheme-token representations is to build two separate SMT systems and then combine them. This can be done as a post-processing system combination step; see (Chen et al., 2009a) for an overview of such approaches. However, for phrase-based SMT systems, it is theoretically more appealing to combine their phrase tables since this allows the translation models of both systems to influence the hypothesis search directly.

We now describe our phrase table combination approach. Note that it is orthogonal to the work presented in the previous section, which suggests combining the two (which we will do in Section 5).

4.1 Building a Twin Translation Model

Figure 3 shows a general scheme of our twin translation model. First, we tokenize the input at different granularities: (1) morpheme-token and (2) word-token. We then build separate phrase tables (PT) for the two inputs: a word-token PT_w and a morpheme-token PT_m. Second, we re-tokenize PT_w at the morpheme level, thus obtaining a new phrase table $PT_{w\rightarrow m}$, which is of the same granularity as PT_m. Finally, we merge $PT_{w\rightarrow m}$ and PT_m, and we input the resulting phrase table to the decoder.

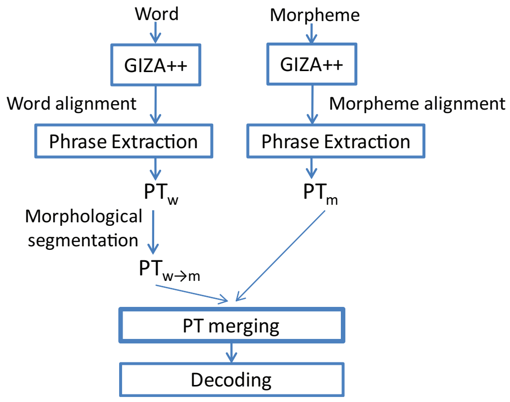

Figure 3: **Building a twin phrase table (PT).** First, separate PTs are generated for different input granularities: word-token and morpheme-token. Second, the word-token PT is retokenized at the morpheme-token level. Finally, the two PTs are merged and used by the decoder.

[3] We use the term "hypothesis" to collectively refer to the following (Koehn, 2003): the *source phrase* covered, the corresponding *target phrase*, and most importantly, a *reference to the previous hypothesis* that it extends.

4.2 Merging and Normalizing Phrase Tables

Below we first describe the two general phrase table combination strategies used in previous work: (1) direct merging using additional feature functions, and (2) phrase table interpolation. We then introduce our approach.

Add-feature methods. The first line of research on phrase table merging is exemplified by (Niehues et al., 2009; Chen et al., 2009b; Do et al., 2009; Nakov and Ng, 2009). The idea is to select one of the phrase tables as primary and to add to it all non-duplicating phrase pairs from the second table together with their associated scores. For each entry, features can be added to indicate its origin (whether from the primary or from the secondary table). Later in our experiments, we will refer to these baseline methods as *add-1* and *add-2*, depending on how many additional features have been added. The values we used for these features in the baseline are given in Section 5.4; their weights in the log-linear model were set in the standard way using MERT.

Interpolation-based methods. A problem with the above method is that the scores in the merged phrase table that correspond to forward and backward phrase translation probabilities, and forward and backward lexicalized translation probabilities can no longer be interpreted as probabilities since they are not normalized any more. Theoretically, this is not necessarily a problem since the log-linear model used by the decoder does not assume that the scores for the feature functions come from a normalized probability distribution. While it is possible to re-normalize the scores to convert them into probabilities, this is rarely done; it also does not solve the problem with the dropped scores for the duplicated phrases. Instead, the conditional probabilities in the two phrase tables are often interpolated directly, e.g., using linear interpolation. Representative work adopting this approach is (Wu and Wang, 2007). We refer to this method as *interpolation*.

Our method. The above phrase merging approaches have been proposed for phrase tables derived from different sources. This is in contrast with our twin translation scenario, where the morpheme-token phrase tables are built from the same training dataset; the main difference being that word alignments and phrase extraction were performed at the word-token level for $PT_{w\rightarrow m}$ and at the morpheme-token level for PT_m. Thus, we propose different merging approaches for the phrase translation probabilities and for the lexicalized probabilities.

In phrase-based SMT, phrase translation probabilities are computed using maximum likelihood (ML) estimation $\phi(\bar{f}|\bar{e}) = \frac{\#(\bar{f},\bar{e})}{\sum_{\bar{f}} \#(\bar{f},\bar{e})}$, where $\#(\bar{f},\bar{e})$ is the number of times the pair $(\bar{f},\bar{e})$ is extracted from the training dataset (Koehn et al., 2003). In order to preserve the normalized ML estimations as much as possible, we refrain from interpolation. Instead, we use the raw counts for the two models $\#_m(\bar{f},\bar{e})$ and $\#_{w\rightarrow m}(\bar{f},\bar{e})$ directly as follows:

$$\phi(\bar{f},\bar{e}) = \frac{\#_m(\bar{f},\bar{e}) + \#_{w\rightarrow m}(\bar{f},\bar{e})}{\sum_{\bar{f}} \#_m(\bar{f},\bar{e}) + \sum_{\bar{f}} \#_{w\rightarrow m}(\bar{f},\bar{e})}$$

For lexicalized translation probabilities, we would like to use simple interpolation. However, we notice that when a phrase pair belongs to only one of the phrase tables, the corresponding lexicalized score for the other table would be zero. This might cause some good phrases to be penalized just because they were not extracted in both tables, which we want to prevent. We thus perform interpolation from PT_m and PT_w according to the following formula:

$$\begin{aligned} \text{lex}(\bar{f}|\bar{e}) &= \alpha \times \text{lex}_m(\bar{f}_m|\bar{e}_m) \\ &+ (1-\alpha) \times \text{lex}_w(\bar{f}_w|\bar{e}_w) \end{aligned}$$

where the concatenation of $\bar{f}_m$ and $\bar{e}_m$ into word-token sequences yields $\bar{f}_w$ and $\bar{e}_w$, respectively.

If both $(\bar{f}_m,\bar{e}_m)$ and $(\bar{f}_w,\bar{e}_w)$ are present in PT_m and PT_w, respectively, we have a simple interpolation of their corresponding lexicalized scores lex_m and lex_w. However, if one of them is missing, we do not use a zero for its corresponding lexicalized score, but use an estimate as follows.

For example, if only the entry $(\bar{f}_m,\bar{e}_m)$ is present in PT_m, we first convert ($\bar{f}_m$,$\bar{e}_m$) into a word-token pair ($\bar{f}_{m\rightarrow w}$,$\bar{e}_{m\rightarrow w}$), and then induce a corresponding word alignment from the morpheme-token alignment of ($\bar{f}_m$,$\bar{e}_m$). We then estimate a lexicalized phrase score using the original formula given in (Koehn et al., 2003), where we plug this induced word alignment and word-token lexical translation probabilities estimated from the word-token dataset The case when $(\bar{f}_w,\bar{e}_w)$ is present in PT_w, but $(\bar{f}_m,\bar{e}_m)$ is not, is solved similarly.

5 Experiments and Evaluation

5.1 Datasets

In our experiments, we use the English-Finnish data from the 2005 shared task (Koehn and Monz, 2005), which is split into training, development, and test portions; see Table 1 for details. We further split the training dataset into four subsets T_1, T_2, T_3, and T_4 of sizes 40K, 80K, 160K, and 320K parallel sentence pairs, which we use for studying the impact of training data size on translation performance.

	Sent.	**Avg. words**		**Avg. morph.**	
		en	fi	en	fi
Train	714K	21.62	15.80	24.68	26.15
Dev	2K	29.33	20.99	33.40	34.94
Test	2K	28.98	20.72	33.10	34.47

Table 1: **Dataset statistics.** Shown are the number of parallel sentences, and the average number of words and *Morfessor* morphemes on the English and Finnish sides of the training, development and test datasets.

5.2 Baseline Systems

We build two phrase-based baseline SMT systems, both using Moses (Koehn et al., 2007):

w-system: works at the word-token level, extracts phrases of up to seven words, and uses a 4-gram word-token LM (as typical for phrase-based SMT);

m-system: works at the morpheme level, tokenized using *Morfessor*[4] and augmented with “+” as described in Section 3.1.

Following Oflazer and El-Kahlout (2007) and Virpioja et al. (2007), we use phrases of up to 10 morpheme-tokens and a 5-gram morpheme-token LM. None of the enhancements described previously is applied yet. After decoding, morphemes are concatenated back to words using the “+” markers.

To evaluate the translation quality, we compute BLEU (Papineni et al., 2001) at the word-token level. We further introduce a morpheme-token version of BLEU, which we call m-BLEU: it first segments the system output and the reference translation into morpheme-tokens and then calculates a BLEU score as usual. Table 2 shows the baseline results. We can see that the *m-system* achieves much higher m-BLEU scores, indicating that it may have better morpheme coverage[5]. However, the *m-system* is outperformed by the *w-system* on the classic word-token BLEU, which means that it either does not perform as well as the *w-system* or that word-token BLEU is not capable of measuring the morpheme-level improvements. We return to this question later.

	w-system		**m-system**	
	BLEU	m-BLEU	BLEU	m-BLEU
T_1	11.56	45.57	11.07	49.15
T_2	12.95	48.63	12.68	53.78
T_3	13.64	50.30	13.32	54.40
T_4	14.20	50.85	13.57	54.70
Full	14.58	53.05	14.08	55.26

Table 2: **Baseline system performance** (on the test dataset). Shown are word BLEU and morpheme m-BLEU scores for the *w-system* and *m-system*.

5.3 Adding Morphological Enhancements

We now add our three morphological enhancements from Section 3 to the baseline *m-system*:

phr (training) allow morpheme-token phrases to get potentially longer than seven morpheme-tokens as long as they cover no more than seven words;

tune (tuning) MERT for morpheme-token translations while optimizing word-token BLEU;

lm (decoding) scoring morpheme-token translation hypotheses with a 5-gram morpheme-token and a 4-gram word-token LM.

The results are shown in Table 3 (ii). As we can see, each of the three enhancements yields improvements in BLEU score over the *m-system*, both for small and for large training corpora. In terms of performance ranking, *tune* achieves the best absolute improvement of 0.66 BLEU points on T_1 and of 0.47 points on the full dataset, followed by *lm* and *phr*.

Table 3 (iii) further shows that using *phr* and *lm* together yields absolute improvements of 0.70 BLEU points on T_1 and 0.50 points on the full training dataset. Further incorporating *tune*, however, only helps when training on T_1.

Overall, the morphological enhancements are on par with the *w-system* baseline, and yield sizable im-

[4] We retrained *Morfessor* for Finnish/English on the Finnish/English side of the training dataset.

[5] Note that these morphemes were generated automatically and thus many of them are erroneous.

	System	T$_1$ (40K)	Full (714K)
(i)	*w-system* (*w*)	11.56	14.58
	m-system (*m*)	11.07	14.08
(ii)	*m+phr*	$11.44^{+0.37}$	$14.43^{+0.35}$
	m+tune	$11.73^{+0.66}$	$14.55^{+0.47}$
	m+lm	$11.58^{+0.51}$	$14.53^{+0.45}$
(iii)	*m+phr+lm*	$11.77^{+0.70}$	$\mathbf{14.58}^{+0.50}$
	m+phr+lm+tune	$\mathbf{11.90}^{+0.83}$	$14.39^{+0.31}$

Table 3: **Impact of the morphological enhancements** (on test dataset). Shown are BLEU scores (in %) for training on T_1 and on the full dataset for (i) baselines, (ii) enhancements individually, and (iii) combined. Superscripts indicate absolute improvements w.r.t *m-system*.

provements over the *m-system* baseline: 0.83 BLEU points on T_1 and 0.50 on the full training dataset.

5.4 Combining Translation Tables

Finally, we investigate the effect of combining phrase tables derived from a word-token and a morpheme-token input, as described in Section 4. We experiment with the following merging methods:

add-1: phrase table merging using one table as primary and adding *one* extra feature[6];

add-2: phrase table merging using one table as primary and adding *two* extra features[7];

interpolation: simple linear interpolation with one parameter α;

ourMethod: our interpolation-like merging method described in Section 4.2.

Parameter tuning. We tune the parameters of the above methods on the development dataset.

	T$_1$ (40K)	Full (714K)
PT_m is primary	11.99	13.45
$PT_{w\rightarrow m}$ is primary	12.26	14.19

Table 4: **Effect of selection of primary phrase table for *add-1*** (on dev dataset): $PT_{w\rightarrow m}$, derived from a word-token input, vs. PT_m, from a morpheme-token input. Shown is BLEU (in %) on T_1 and the full training dataset.

For *add-1* and *add-2*, we need to decide which ($PT_{w\rightarrow m}$ or PT_m) phrase table should be considered the primary table. Table 4 shows the results when trying both strategies on *add-1*. As we can see, using $PT_{w\rightarrow m}$ as primary performs better on T_1 and on the full training dataset; thus, we will use it as primary on the test dataset for *add-1* and *add-2*.

For interpolation-based methods, we need to choose a value for the interpolation parameters. Due to time constraints, we use the same value for the phrase translation probabilities and for the lexicalized probabilities, and we perform grid search for $\alpha \in \{0.3, 0.4, 0.5, 0.6, 0.7\}$ using *interpolate* on the full training dataset. As Table 5 shows, $\alpha = 0.6$ turns out to work best on the development dataset; we will use this value in our experiments on the test dataset both for *interpolate* and for *ourMethod*[8].

α	0.3	0.4	0.5	0.6	0.7
BLEU	14.17	14.49	14.6	14.73	14.52

Table 5: **Trying different values for *interpolate*** (on dev dataset). BLEU (in %) is for the full training dataset.

Evaluation on the test dataset. We integrate the morphologically enhanced system *m+phr+lm* and the word-token based *w-system* using the four merging methods above. The results for the full training dataset are shown in Table 6. As we can see, *add-1* and *add-2* make little difference compared to the *m-system* baseline. In contrast, *interpolation* and *ourMethod* yield sizable absolute improvements of 0.55 and 0.74 BLEU points, respectively, over the *m-system*; moreover, they outperform the *w-system*.

	Merging methods	Full (714K)
(i)	*m-system*	14.08
	w-system	14.58
(ii)	*add-1*	$14.25^{+0.17}$
	add-2	$13.89^{-0.19}$
(iii)	*interpolation*	$14.63^{+0.55}$
	ourMethod	$\mathbf{14.82}^{+0.74}$

Table 6: **Merging *m+phr+lm* and *w-system*** (on test dataset). BLEU (in %) is for the full training dataset. Superscripts indicate performance gain/loss w.r.t *m-system*.

6 Discussion

Below we assess the significance of our results based on micro-analysis and human judgments.

[6] The feature values are e^1, $e^{2/3}$ or $e^{1/3}$ (e=2.71828...); when the phrase pair comes from both tables, from the primary table only, and from the secondary table only, respectively.

[7] The feature values are (e^1, e^1), (e^1, e^0) or (e^0, e^1) when the phrase pair comes from both tables, from the primary table only, and from the secondary table only, respectively.

[8] Note that this might put *ourMethod* at disadvantage.

6.1 Translation Model Comparison

We first compare the following three phrase tables: PT_m of *m-system*, maximum phrase length of 10 morpheme-tokens; $PT_{w\to m}$ of *w-system*, maximum phrase length of 7 word-tokens, re-segmented into morpheme-tokens; and PT_{m+phr} – morpheme-token input using word boundary-aware phrase extraction, maximum phrase length of 7 word-tokens.

		Full (714K)
(i)	PT_m	43.5M
	$PT_{w\to m}$	28.9M
	PT_{m+phr}	22.5M
(ii)	$PT_{m+phr} \bigcap PT_m$	21.4M
	$PT_{m+phr} \bigcap PT_{w\to m}$	10.7M

Table 7: **Phrase table statistics.** The number of phrase pairs in (i) individual PTs and (ii) PT overlap, is shown.

$\boldsymbol{PT_{m+phr}}$ **versus** $\boldsymbol{PT_m}$**.** Table 7 shows that PT_{m+phr} is about half the size of PT_m. Still, as Table 3 shows, *m+phr* outperforms the *m-system*. Moreover, 95.07% (21.4M/22.5M) of the phrase pairs in PT_{m+phr} are also in PT_m, which confirms that boundary-aware phrase extraction selects good phrase pairs from PT_m to be retained in PT_{m+phr}.

$\boldsymbol{PT_{m+phr}}$ **versus** $\boldsymbol{PT_{w\to m}}$**.** These two tables are comparable in size: 22.5M and 28.9M pairs, but their overlap is only 47.67% (10.7M/22.5M) of PT_{m+phr}. Thus, enriching the translation model with $PT_{w\to m}$ helps improve coverage.

6.2 Significance of the Results

Table 8 shows the performance of our system compared to the two baselines: *m-system* and *w-system*. We achieve an absolute improvement of 0.74 BLEU points over the *m-system*, from which our system evolved. This might look modest, but note that the baseline BLEU is only 14.08, and thus the relative improvement is 5.6%, which is not trivial. Furthermore, we outperform the *w-system* by 0.24 points (1.56% relative). Both improvements are statistically significant with $p < 0.01$, according to Collins' sign test (Collins et al., 2005).

In terms of m-BLEU, we achieve an improvement of 2.59 points over the *w-system*, which suggest our system might be performing better than what standard BLEU suggests. Below we test this hypothesis by means of micro-analysis and human evaluation.

	BLEU	**m-BLEU**
ourSystem	14.82	55.64
m-system	14.08	55.26
w-system	14.58	53.05

Table 8: **Our system vs. the two baselines** (on the test dataset): BLEU and m-BLEU scores (in %).

Translation Proximity Match. We performed automatic comparison based on corresponding phrases between the translation output (*out*) and the reference (*ref*), using the source (*src*) test dataset as a pivot. The decoding log gave us the phrases used to translate *src* to *out*, and we only needed to find correspondences between *src* and *ref*, which we accomplished by appending the test dataset to training and performing IBM Model 4 word alignments.

We then looked for phrase triples (*src*, *out*, *ref*), where there was a high character-level similarity between *out* and *ref*, measured using *longest common subsequence ratio* with a threshold of 0.7, set experimentally. We extracted 16,262 triples: for 6,758 of them, the translations matched the references exactly, while in the remaining triples, they were close wordforms[9]. These numbers support the hypothesis that our approach yields translations close to the reference wordforms but unjustly penalized by BLEU, which only gives credit for exact word matches[10].

Human Evaluation. We asked four native Finnish speakers to evaluate 50 random test sentences. Following (Callison-Burch et al., 2009), we provided them with the source sentence, its reference translation, and the outputs of three SMT systems (*m-system*, *w-system*, and *ourSystem*), which were shown in different order for each example and were named *sys1*, *sys2* and *sys3* (by order of appearance). We asked for three pairwise judgments: (i) *sys1* vs. *sys2*, (ii) *sys1* vs. *sys3*, and (iii) *sys2* vs. *sys3*. For each pair, a winner had to be designated; ties were allowed. The results are shown in Table 10. We can see that the judges consistently preferred

[9] Examples of such triples are (`constitutional structure`, perustuslaillinen rakenne, perustuslaillisempi rakenne) and (`economic and social`, taloudellisia ja sosiaalisia, taloudellisten ja sosiaalisten)

[10] As a reference, the *w-system* yielded 15,673 triples, and 6,392 of them were exact matches. Compared to our system, this means 589 triples and 366 exact matches less.

`src`: as a conservative , i am incredibly thrifty with taxpayers ' money .
`ref`: maltillisen kokoomuspuolueen edustajana suhtaudun **erittain saastavaisesti veronmaksajien** rahoihin .
`our`: konservatiivinen , olen **erittain saastavaisesti veronmaksajien** rahoja .
`w` : konservatiivinen , olen aarettoman tarkeaa kanssa *veronmaksajien* rahoja .
`m` : *kuten* konservatiivinen , olen **erittain saastavaisesti veronmaksajien** rahoja .
Comment: **`our`** $\succ$ **`m`** $\succ$ **`w`**. **`our`** uses better paraphrases, from which the correct meaning could be inferred. The part "aarettoman tarkeaa kanssa" in **`w`** does not mention the "thriftiness" and replaces it with "important" (tarkeaa), which is wrong. **`m`** introduces "kuten", which slightly alters the meaning towards "like a conservative, ...".

`src`: we were very constructive and we negotiated until the last minute of these talks in the hague .
`ref`: olimme erittain **rakentavia** ja neuvottelimme haagissa **viime hetkeen saakka** .
`our`: olemme olleet hyvin **rakentavia** ja olemme neuvotelleet **viime hetkeen saakka** naiden neuvottelujen haagissa .
`w` : olemme olleet hyvin **rakentavia** ja olemme neuvotelleet *viime tippaan niin* naiden neuvottelujen haagissa .
`m` : olimme erittain *rakentavan* ja neuvottelimme **viime hetkeen saakka** naiden neuvotteluiden haagissa .
Comment: **`our`** $\succ$ **`m`** $\succeq$ **`w`**. In **`our`**, the meaning is very close to **`ref`** with only a minor difference in tense at the beginning. **`m`** only gets the case wrong in "rakentavan", and the correct case is easily guessable. For **`w`**, the "viime tippaan" is in principle correct but somewhat colloquial, and the "niin" is extra and somewhat confusing.

`src`: it would be a very dangerous situation if the europeans were to become logistically reliant on russia .
`ref`: olisi **erittäin** vaarallinen tilanne , jos **eurooppalaiset** tulisivat **logistisesti** riippuvaisiksi venäjästä .
`our`: olisi **erittäin** vaarallinen tilanne , jos **eurooppalaiset** tulee **logistisesti** riippuvaisia venäjän .
`w` : *se* olisi **erittäin** vaarallinen tilanne , jos *eurooppalaisten* tulisi *logistically* riippuvaisia venäjän .
`m` : *se* olisi *hyvin* vaarallinen tilanne , jos **eurooppalaiset** *haluavat* tulla **logistisesti** riippuvaisia venäjän .
Comment: **`our`** $\succ$ **`w`** $\succeq$ **`m`**. **`our`** is almost correct except for the wrong inflections at the end. **`w`** is inferior since it failed to translate "logistically". "haluavat tulla" in **`m`** suggests that the Europeans would "want to become logistically dependent", which is not the case. The "se" (it), and "hyvin" (a synonym of "erittäin") are minor mistakes/differences.

Table 9: **English-Finnish translation examples**. Shown are the source (`src`), the reference (`ref`), and the translations of three systems (`our`, `w`, `m`). Text in bold indicates matches with respect to the `ref`, while italics show where a system was judged inferior to the rest, as judged by native Finnish speakers.

(1) *ourSystem* to the *m-system*, (2) *ourSystem* to the *w-system*, (3) *w-system* to the *m-system*. These preferences are statistically significant, as found by the sign test. Comparing to Table 8, we can see that BLEU correlates with human judgments better than m-BLEU; we plan to investigate this in future work.

	our vs. m		**our vs. w**		**w vs. m**	
Judge 1	25	18	19	12	21	19
Judge 2	24	16	19	15	25	14
Judge 3	**27**†	**12**	17	11	**27**†	**15**
Judge 4	25	20	**26**†	**12**	22	22
Total	**101**‡	**66**	**81**‡	**50**	**95**†	**70**

Table 10: **Human judgments:** *ourSystem* (our) vs. *m-system* (m) vs. *w-system* (w). For each pair, we show the number of times each system was judged better than the other one, ignoring ties. Statistically significant differences are marked with † ($p < 0.05$) and ‡ ($p < 0.01$).

Finally, Table 9 shows some examples demonstrating how our system improves over the *w-system* and the *m-system*.

7 Conclusion and Future Work

In the quest towards a morphology-aware SMT that only uses unannotated data, there are two key challenges: (1) to bring the performance of morpheme-token systems to a level rivaling the standard word-token ones, and (2) to incorporate morphological analysis directly into the translation process.

This work satisfies the first challenge: we have achieved statistically significant improvements in BLEU for a large training dataset of 714K sentence pairs and this was confirmed by human evaluation.

We think we have built a solid framework for the second challenge, and we plan to extend it further.

Acknowledgements

We thank Joanna Bergström-Lehtovirta (Helsinki Institute for Information Technology), Katri Haverinen (University of Turku and Turku Centre for Computer Science), Veronika Laippala (University of Turku), and Sampo Pyysalo (University of Tokyo) for judging the Finnish translations.

References

Eleftherios Avramidis and Philipp Koehn. 2008. Enriching morphologically poor languages for statistical machine translation. In *ACL-HLT*.

Ibrahim Badr, Rabih Zbib, and James Glass. 2008. Segmentation for English-to-Arabic statistical machine translation. In *ACL-HLT*.

Tim Buckwalter. 2004. Buckwalter Arabic Morphological Analyzer Version 2.0. Linguistic Data Consortium, Philadelphia".

Chris Callison-Burch, Philipp Koehn, Christof Monz, and Josh Schroeder. 2009. Findings of the 2009 Workshop on Statistical Machine Translation. In *EACL*.

Boxing Chen, Min Zhang, Haizhou Li, and Aiti Aw. 2009a. A comparative study of hypothesis alignment and its improvement for machine translation system combination. In *ACL-IJCNLP*.

Yu Chen, Michael Jellinghaus, Andreas Eisele, Yi Zhang, Sabine Hunsicker, Silke Theison, Christian Federmann, and Hans Uszkoreit. 2009b. Combining multiengine translations with Moses. In *EACL*.

Michael Collins, Philipp Koehn, and Ivona Kučerová. 2005. Clause restructuring for statistical machine translation. In *ACL*.

Mathias Creutz and Krista Lagus. 2007. Unsupervised models for morpheme segmentation and morphology learning. *ACM Trans. Speech Lang. Process.*, 4(1):3.

Thi Ngoc Diep Do, Viet Bac Le, Brigitte Bigi, Laurent Besacier, and Eric Castelli. 2009. Mining a comparable text corpus for a Vietnamese-French statistical machine translation system. In *EACL*.

Sharon Goldwater and David McClosky. 2005. Improving statistical MT through morphological analysis. In *HLT*.

Philipp Koehn and Barry Haddow. 2009. Edinburgh's submission to all tracks of the WMT2009 shared task with reordering and speed improvements to Moses. In *EACL*.

Philipp Koehn and Hieu Hoang. 2007. Factored translation models. In *EMNLP-CoNLL*.

Philipp Koehn and Christof Monz. 2005. Shared task: Statistical machine translation between European languages. In *WPT*.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *NAACL*.

Philipp Koehn, Hieu Hoang, Alexandra Birch Mayne, Christopher 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 *ACL, Demonstration Session*.

Philipp Koehn. 2003. *Noun phrase translation*. Ph.D. thesis, University of Southern California, Los Angeles, CA, USA.

Young-Suk Lee. 2004. Morphological analysis for statistical machine translation. In *HLT-NAACL*.

Preslav Nakov and Hwee Tou Ng. 2009. Improved statistical machine translation for resource-poor languages using related resource-rich languages. In *EMNLP*.

Jan Niehues, Teresa Herrmann, Muntsin Kolss, and Alex Waibel. 2009. The Universität Karlsruhe translation system for the EACL-WMT 2009. In *EACL*.

Attila Novák. 2009. MorphoLogic's submission for the WMT 2009 shared task. In *EACL*.

Franz Josef Och and Hermann Ney. 2004. The alignment template approach to statistical machine translation. *Computational Linguistics*, 30(4):417–449.

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In *ACL*.

Kemal Oflazer and Ilknur El-Kahlout. 2007. Exploring different representational units in English-to-Turkish statistical machine translation. In *StatMT*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2001. Bleu: a method for automatic evaluation of machine translation. In *ACL*.

Fatiha Sadat and Nizar Habash. 2006. Combination of Arabic preprocessing schemes for statistical machine translation. In *ACL*.

Helmut Schmid. 1994. Probabilistic part-of-speech tagging using decision trees. In *International Conference on New Methods in Language Processing*.

Kristina Toutanova, Hisami Suzuki, and Achim Ruopp. 2008. Applying morphology generation models to machine translation. In *ACL-HLT*.

Sami Virpioja, Jaakko J. Vyrynen, Mathias Creutz, and Markus Sadeniemi. 2007. Morphology-aware statistical machine translation based on morphs induced in an unsupervised manner. In *Machine Translation Summit XI*.

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

Mei Yang and Katrin Kirchhoff. 2006. Phrase-based backoff models for machine translation of highly inflected languages. In *EACL*.