Class-Based Language Modeling for Translating into Morphologically Rich Languages

Arianna Bisazza and **Christof Monz**
Informatics Institute, University of Amsterdam
Science Park 904, 1098 XH Amsterdam, The Netherlands
{a.bisazza,c.monz}@uva.nl

Abstract

Class-based language modeling (LM) is a long-studied and effective approach to overcome data sparsity in the context of n-gram model training. In statistical machine translation (SMT), different forms of class-based LMs have been shown to improve baseline translation quality when used in combination with standard word-level LMs but no published work has systematically compared different kinds of classes, model forms and LM combination methods in a unified SMT setting. This paper aims to fill these gaps by focusing on the challenging problem of translating into Russian, a language with rich inflectional morphology and complex agreement phenomena. We conduct our evaluation in a large-data scenario and report statistically significant BLEU improvements of up to 0.6 points when using a refined variant of the class-based model originally proposed by Brown et al. (1992).

1 Introduction

Class-based n-gram modeling is an effective approach to overcome data sparsity in language model (LM) training. By grouping words with similar distributional behavior into equivalence classes, class-based LMs have less parameters to train and can make predictions based on longer histories. This makes them particularly attractive in situations where n-gram coverage is low due to shortage of training data or to specific properties of the language at hand.

While translation into English has drawn most of the research effort in statistical machine translation (SMT) so far, there is now a growing interest in translating into languages that are more challenging for standard n-gram modeling techniques. Notably, morphologically rich languages are characterized by high type/token ratios (T/T) that reflect in high out-of-vocabulary word rates and frequent backing-off to low order n-gram estimates, even when large amounts of training data are used. These problems have been long studied in the field of speech recognition but much less in SMT, although the target LM is a core component of all state-of-the-art SMT frameworks.

Partly inspired by successful research in the field of speech recognition, various forms of class-based LMs have been shown to improve the quality of SMT when used in combination with standard word-level LMs. These approaches, however, have mostly focused on English (Uszkoreit and Brants, 2008; Dyer et al., 2011; Monz, 2011; Hassan et al., 2007; Birch et al., 2007) with only recent exceptions (Green and DeNero, 2012; Ammar et al., 2013; Wuebker et al., 2013; Durrani et al., 2014). Moreover, there is no published work that systematically evaluates different kinds of classes, model forms and LM combination methods in a unified SMT setting. On the contrary, most of the existing literature on LM combination uses mixtures of multiple *word*-level LMs for domain adaptation purposes.

This paper aims to fill these gaps by applying various class-based LM techniques to the challenging problem of translating *into* a morphologically rich language. In particular we focus on English-Russian, a language pair for which a fair amount of both parallel data and monolingual data has been provided by the Workshop of Machine Translation (Bojar et al., 2013). Russian is characterized by a rich inflectional morphology, with a particularly complex nominal declension (six core cases, three genders and two

Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers, pages 1918–1927, Dublin, Ireland, August 23-29 2014.

number categories). This results in complex agreement phenomena and an extremely rich vocabulary. Indeed, by examining our training data (see Section 4), we find the Russian T/T ratio to be almost two times higher than the English one.

Given this task, we make a number of contributions leading to a better understanding of ways to utilize class-based language models for translating into morphologically rich languages. We conduct a comparative evaluation of different target LMs along the following axes: (1) Classes: data-driven versus shallow morphology-based; (2) Model forms: simple class sequence (stream-based) versus original class-based (Brown et al., 1992); and (3) Combination frameworks: model-level log-linear combination versus word-level linear interpolation. When comparing the different model forms we pay particular attention to the role word emission probabilities play in class-based models, which turns out to be a significant factor for translating into morphologically rich languages. In this context we also evaluate for the first time a specific form of class-based LM called fullibm (Goodman, 2001) within statistical MT.

2 Class-based language models

As introduced by (Brown et al., 1992), the idea of class-based n-gram language modeling is to group words with similar distributional behavior into equivalence classes. The word transition probability is then decomposed into a class transition probability and a word emission probability:

$$P_{\text{class}}(w_i|w_{i-n+1}^{i-1}) = p_0(C(w_i)|C(w_{i-n+1}^{i-1})) \cdot p_1(w_i|C(w_i)) \quad (1)$$

This results in models that are more compact and more robust to data sparsity. Often, in the context of SMT, the word emission probability is dropped and only the class sequence is modeled. In this work, we refer to this model form as *stream-based* n-gram LM:[1]

$$P_{\text{stream}}(w_i|w_{i-n+1}^{i-1}) = p_0(C(w_i)|C(w_{i-n+1}^{i-1})) \quad (2)$$

Stream-based LMs are used, for instance, in factored SMT (Koehn et al., 2007), and in general many of the 'class-based LMs' mentioned in the SMT literature are actually of the latter form (2) (Dyer et al., 2011; Green and DeNero, 2012; Ammar et al., 2013; Chahuneau et al., 2013; Wuebker et al., 2013; Durrani et al., 2014). One exception is the work of Uszkoreit and Brants (2008), who incorporate word emission probabilities in their class-based LM used as an additional feature function in the log-linear combination (cf. Section 3.1). Interestingly, we are not aware of work that compares actual class-based LMs and stream-based LMs with respect to SMT quality.

While class-based LMs are known to be effective at counteracting data sparsity issues due to rich vocabularies, it is worth noting that they adhere to the fundamental constraints of n-gram modeling. Thus, grammatical agreement may be improved by a class-based LM approach only within a limited context window. Previous work that attempted to overcome this limitation includes (i) syntactic LMs for n-best reranking (Hasan et al., 2006; Carter and Monz, 2011) or integrated into decoding with significant engineering challenges (Galley and Manning, 2009; Schwartz et al., 2011) and (ii) unification-based constraints applied to a syntax-based SMT framework (Williams and Koehn, 2011).

We will now describe different kinds of word-to-class mapping functions used by class-based LMs. These can be completely data-driven or based on different sorts of linguistic or orthographic features.

2.1 Data-driven classes

The most popular form of class-based LMs was introduced by (Brown et al., 1992). In this approach, the corpus vocabulary is partitioned into a preset number of clusters by directly maximizing the likelihood of a training corpus. No linguistic or orthographic features are taken into account while training the classes.[2] Later work has focused on decreasing the large computational cost of the exchange algorithm proposed by Brown et al. (1992), either with a distributed algorithm (Uszkoreit and Brants, 2008) or by using a whole-context distributional vector space model (Schütze and Walsh, 2011). In this paper we use the standard SRILM implementation of Brown clustering.

[1] Not to be confused with the incrementally trainable stream-based LMs of Levenberg and Osborne (2009).

[2] Och (1999) extends a similar approach to bilingual clustering with the aim of generalizing the applicability of translation rules in an alignment template SMT framework.

2.2 Linguistic classes

Linguistic knowledge is another way to establish word equivalence classes. Common examples include lemma, part of speech and morphology-based classes, each of which can capture different aspects of the word sequence, such as the relative order of syntactic constituents or grammatical agreement. Hassan et al. (2007) and Birch et al. (2007) went as far as scoring n-grams of Combinatorial Categorial Grammar supertags. When using linguistic classes, one has to deal with the fact that the same word can belong to different classes when used in different contexts. Solutions to this problem include tagging the target word sequence as it is generated (Koehn et al., 2007; Birch et al., 2007; Green and DeNero, 2012), choosing the most probable class sequence for each phrase pair (Monz, 2011) or—even more lightweight—choosing the most probable class for each word (Bisazza and Federico, 2012).

Alternatively, simpler deterministic class mappings can be derived by using shallow linguistic knowledge, such as suffixes or orthographic features. The former can be obtained with a rule-based stemmer (as in this work), or, even more simply, by selecting the ϕ most common word suffixes in a training corpus and then mapping each word to its longest matching suffix (Müller et al., 2012). Orthographic features may include capitalization information or the presence of digits, punctuation or other special characters (Müller et al., 2012).

2.3 Hybrid surface/class models

Müller et al. (2012) obtain the best perplexity reduction when excluding frequent words from the class mapping. That is, each word with more than θ occurrences in the training corpus is assigned to a singleton class with word emission probability equal to 1. The frequency threshold θ is determined with a grid search on a monolingual held-out set. Optimal values for perplexities are shown to vary considerably among languages. In this work we follow this setup closely.

It is worth noting that Bisazza and Federico (2012) have applied a similar idea to the problem of style adaptation: they train a hybrid POS/word n-gram LM on an in-domain corpus and use it as an additional SMT feature function with the goal of counterbalancing the bias towards the style of the large out-of-domain data. The idea of modeling sequences of mixed granularity (word/subword) was earlier introduced to speech recognition by Yazgan and Saraçlar (2004).

The most extensive comparison of distributional, morphological and hybrid classes that we are aware of is the work by Müller et al. (2012), but that does not include any SMT evaluation. Looking at perplexity results over a large number of European language pairs (not including Russian), Müller et al. (2012) conclude that a hybrid suffix/word class-based LM simply built on frequency-based suffixes performs as well as a model trained on much more expensive distributional classes. Motivated by this finding, we evaluate these two kinds of classes in the context of SMT into a morphologically rich language.

2.4 Fullibm language model

As outlined above, the class-based LMs generally used in SMT are in fact stream-based models in the sense that they only estimate the probability of the class sequence (see Equation 2). However, the classic form of class-based LM (Brown et al., 1992) also includes a class-to-word emission probability $p_1(w_i|C(w_i))$ whose utility has not been properly assessed in the context of SMT.

Besides, we observe that a variety of class-based LM variants have been studied in speech recognition but not in SMT. In particular, Goodman (2001) presents a generalization of the standard class-based form where the word emission is also conditioned on the class history rather than on the current class alone. The resulting model is called *fullibm*:

$$P_{\text{fullibm}}(w_i|w_{i-n+1}^{i-1}) = p_0(C(w_i)|C(w_{i-n+1}^{i-1})) \cdot p_1(w_i|C(w_{i-n+1}^{i})) \quad (3)$$

We expect this model to yield more refined, context-sensitive word emission distributions which may result in better target LM probabilities for our SMT system.

3 SMT combining framework

Class-based LMs are rarely used in isolation, but are rather combined with standard word-level models. There exist at least two ways to combine multiple LMs into a log-linear SMT decoder: (i) as separate feature functions in the global log-linear combination or (ii) as components of a linear mixture counting as a single feature function in the global combination.

3.1 Log-linear combination

The standard log-linear approach to SMT allows for the combination of m arbitrary model components (or feature functions), each weighted by a corresponding weight α_m:

$$p(x|h) = \prod_m p_m(x|h)^{\alpha_m} \tag{4}$$

In typical SMT settings, $p_m(x|h)$ are phrase- or word-level translation probabilities, reordering probabilities, and so on. Treating the new LM as an additional feature function has the advantage that its weight can be directly optimized for SMT quality together with all other feature weights, using standard parameter tuning techniques (Och, 2003; Hopkins and May, 2011).

3.2 Linear interpolation

The other widely used combining framework is linear interpolation or mixture model:

$$p(x|h) = \sum_q \lambda_q p_q(x|h) \tag{5}$$

More specifically, word LMs are usually interpolated as a word-level weighted average of the n-gram probabilities:

$$p_{\text{mixLM}}(\mathbf{e}) = \prod_{i=1}^{n} \left(\sum_q \lambda_q p_q(e_i|h_i) \right) \tag{6}$$

The drawback of this approach is that the linear interpolation weights, or *lambdas*, cannot be set with standard SMT tuning techniques. Instead, interpolation weights are typically determined by maximizing the likelihood of a held-out monolingual data set, but this does not always outperform simple uniform weighting in terms of translation quality.[3]

Despite the lambda optimization issue, linear interpolation with uniform or maximum-likelihood weights has been shown to work better for SMT than log-linear combination when combining regular word n-gram LMs (Foster and Kuhn, 2007). However, to the best of our knowledge, the linear interpolation of word- and class-based LMs has never been tested in SMT.

In their intrinsic evaluation, Müller et al. (2012) show that linear mixing with hybrid class/surface models of various kinds consistently decrease the perplexity of a Kneser-Ney smoothed word-level LM, with relative improvements ranging between 3% (English) and 11% (Finnish). All their models are interpolated with class-specific lambda weights, according to the following formula:

$$P_{\text{mix}}(w_i|w_{i-n+1}^{i-1}) = \lambda_{C(w_{i-1})} \cdot P_{\text{class}}(w_i|w_{i-n+1}^{i-1}) + (1 - \lambda_{C(w_{i-1})}) \cdot P_{\text{word}}(w_i|w_{i-n+1}^{i-1}) \tag{7}$$

where P_{word} corresponds to the standard n-gram model using the lexical forms. Equation 7 can be seen as a generalization of the simple interpolation $\lambda P_{\text{class}} + (1 - \lambda) P_{\text{word}}$ used by Brown et al. (1992). The class-specific lambdas are estimated by a deleted interpolation algorithm (Bahl et al., 1991). In our experiments, we test both generic and class-specific lambda interpolation for SMT.

[3] Foster and Kuhn (2007) also tried more sophisticated techniques to set interpolation weights but did not obtain significant improvements.

Corpus	Lang.	#Sent.	#Tok.	T/T
paral.train	EN	1.9M	48.9M	.0107
	RU		45.9M	.0204
Wiki dict.	EN/RU	508K	–	–
mono.train	RU	21.0M	390M	.0068
newstest12	EN	3K	64K	–
newstest13		3K	56K	–

Table 1: Training and test data statistics: number of sentences, number of tokens and type/token ratio (T/T). All numbers refer to tokenized, lowercased data.

4 Evaluation

We perform a series of experiments to compare the effectiveness for SMT of various class mapping functions, different model forms, and different LM combining frameworks.

The task, organized by the Workshop of Machine Translation (WMT, Bojar et al. (2013)), consists of translating a set of news stories from English to Russian. As shown in Table 1, the available data includes a fairly large parallel training corpus (1.9M sentences) from various sources, a set of Wikipedia parallel headlines shared by CMU,[4] and a larger monolingual corpus for model training (21M sentences). By measuring the type/token ratios of the two sides of a parallel corpus, we can estimate the difference in morphological complexity between two languages: as shown in Table 1, the Russian T/T is almost two times higher than the English one (.0204 vs .0107) in the WMT13 parallel training data. As is usually the case, much more data is available for LM training. Nevertheless we report a rather high out-of-vocabulary word rate on the devsets' reference translations (2.28%).

4.1 Baseline

Our baseline is an in-house phrase-based (Koehn et al., 2003) statistical machine translation system very similar to Moses (Koehn et al., 2007). All system runs use hierarchical lexicalized reordering (Galley and Manning, 2008; Cherry et al., 2012), distinguishing between monotone, swap, and discontinuous reordering, all with respect to left-to-right and right-to-left decoding. Other features include linear distortion, bidirectional lexical weighting (Koehn et al., 2003), word and phrase penalties, and finally a word-level 5-gram target language model trained on all available monolingual data with modified Kneser-Ney smoothing (Chen and Goodman, 1999). The distortion limit is set to 6 and for each source phrase the top 30 translation candidates are considered.

The feature weights for all approaches were tuned by using pairwise ranking optimization (Hopkins and May, 2011) on newstest12. During tuning, 14 PRO parameter estimation runs are performed in parallel on different samples of the n-best list after each decoder iteration. The weights of the individual PRO runs are then averaged and passed on to the next decoding iteration. Performing weight estimation independently for a number of samples corrects for some of the instability that can be caused by individual samples.

4.2 Language models

The additional LMs are trained with Witten-Bell smoothing (Witten and Bell, 1991), which is a common choice for class-based LM training as Kneser-Ney smoothing cannot be used for computing discount factors when the count-of-counts are zero. The main series of experiments employ 5-gram models, but we also evaluate the usefulness of increasing the order to 7-gram (see Table 3).[5]

Data-driven clusters are learned with the standard Brown clustering algorithm, which greedily maximizes the log likelihood of a class bigram model on the training data. Following Ammar et al. (2013), we set the number of data-driven clusters to 600. In preliminary experiments we also tested a 256-cluster setting, but 600 yielded better BLEU scores. For time reasons, we train the clusters on a subset of the

[4] http://www.statmt.org/wmt13/wiki-titles.ru-en.tar.gz

[5] For this second series of experiments we use the feature weights tuned for the corresponding 5-gram LMs.

LM type	smoothing	vocab.	PP
words	Kneser-Ney	2.7M	270
Brown clusters	Witten-Bell	600	588
suffixes	Witten-Bell	968	2455
suffix/word hybrid (θ=5000)	Witten-Bell	8530	460

Linear interp.	PP	
	generic λ	class-spec.λ's
words + clusters	225	224
words + suffixes	266	265
words + hybrid	243	247

Table 2: Intrinsic evaluation of various types of LMs and their linear interpolations. Perplexity (PP) is computed on a separate held-out set of 5K Russian sentences. All models are 5-grams.

monolingual data including all the parallel data (news commentary) and the large commoncrawl corpus for a total of 1M sentences (22M tokens). We then map all monolingual data to the learned clusters and use that to train all our cluster-based LMs.

For the suffix-based class LMs we closely follow the setup of Müller et al. (2012) with the only difference that we use the Russian Snowball stemmer[6] to segment the vocabulary instead of frequency-based suffixes. The suffix threshold θ (see Section 2.3) is determined by minimizing perplexity on a separate held-out set (5K sentences): θ=5000 is the optimal setting among $\{2000, 5000, 10000, 20000\}$.[7] The same held-out set is used to estimate both the generic and the class-specific lambdas for the linear interpolation experiments.

Table 2 presents an overview of the LMs used in our experiments. We can see on the left side that all class-based LMs have notably higher perplexities compared to the word-level, with the fully suffix-based LM performing worst by far. Nevertheless, all class-based models yield a decrease in perplexity when they are interpolated with the word-level model (right side). The best improvement is achieved by the data-driven classes (225 versus 270, that is -17%), but the result of the hybrid LM is also quite successful (-10%) and much in line with the improvements reported by Müller et al. (2012) on other Slavic languages. Because the fully suffix-based LM yields only a modest reduction, we do not to include it in the SMT evaluation. The right side of Table 2 also shows that using class-specific interpolation weights is not significantly better, and sometimes is even worse than using only one generic λ, at least from the point of view of perplexity. Since weight estimation for linear interpolation is still an open problem for SMT, we decide nevertheless to compare these two interpolation methods in our translation experiments (see Table 4).

4.3 SMT results

Table 3 shows the results for English to Russian translation using log-linear combination with Brown clusters and the hybrid suffix/word classes. Translation quality is measured by case-insensitive BLEU (Papineni et al., 2002) on newstest13 using one reference translation. The relative improvements of the different class-based LM runs are with respect to the baseline which uses a word-based LM only and achieves comparable results to the state-of-the-art. We use approximate randomization (Noreen, 1989) to test for statistically significant differences between runs (Riezler and Maxwell, 2005).

We can see from Table 2(a) that using a stream-based LM as an additional feature, which is log-linearly interpolated with the other decoder features during parameter estimation, leads to small but statistically significant improvements. The results also indicate that using a higher n-gram class model (7-gram) does not yield additional improvements over a 5-gram class model, which is in contrast with the results reported by Wuebker et al. (2013) on a French-German task.

Since the stream-based models ignore word emission probabilities, one would expect further improvements from the theoretically more correct class-based model which include word emission probabilities (see Equation 1). Somewhat surprisingly, this is not the case. On the contrary, both 5- and 7-gram class-based models perform slightly worse than the stream-based models. We suspect that this is due to the limited context used to estimate the emission probabilities in the original Brown class-based models. To verify this we compared this to the fullibm model (Equation 3) which conditions word emission

[6] http://snowball.tartarus.org/algorithms/russian/stemmer.html

[7] Our training corpus is considerably larger than those used by Müller et al. (2012), therefore we search among higher values.

(a) Brown clusters (600)

Additional LM	surface BLEU	surface Δ	stem BLEU	stem Δ
⋆ none [baseline]	18.8	—	24.7	—
⋆ 5g stream-based	19.1	$+0.3^{\bullet}$	24.8	$+0.1$
7g stream-based	19.1	$+0.3^{\bullet}$	24.9	$+0.2$
⋆ 5g class-based	18.9	$+0.1$	24.6	-0.1
7g class-based	18.8	± 0.0	24.7	± 0.0
5g fullibm	19.4	$+0.6^{\bullet}$	25.0	$+0.3^{\bullet}$
7g fullibm	19.3	$+0.5^{\bullet}$	25.0	$+0.3^{\bullet}$

(b) Suffixes/words, $\theta = 5000$

Additional LM	surface BLEU	surface Δ	stem BLEU	stem Δ
⋆ none [baseline]	18.8	—	24.7	—
⋆ 5g stream-based	18.9	$+0.1$	24.6	-0.1
7g stream-based	18.9	$+0.1$	24.6	-0.1
⋆ 5g class-based	19.0	$+0.2^{\circ}$	24.8	$+0.1$
7g class-based	19.1	$+0.3^{\circ}$	24.7	± 0.0
5g fullibm	19.1	$+0.3^{\bullet}$	24.8	$+0.1$
7g fullibm	19.2	$+0.4^{\bullet}$	24.9	$+0.2^{\circ}$

Table 3: SMT translation quality on newstest13 when using different kinds of class-based language models as additional features in the log-linear combination. The settings used for weight tuning are marked with ⋆. Statistically significant differences wrt the baseline are marked with $^{\bullet}$ at the $p \leq .01$ level and $^{\circ}$ at the $p \leq .05$ level.

probabilities on the entire n-gram class history of length $n - 1$. The fullibm class-based models yield the biggest statistically significant improvements over the baseline and also compare favorably to the stream-based and original class-based models. Similarly to stream- and class-based models we do not observe a difference in performance between 5- and 7-gram models for fullibm.

Table 2(b) shows the results obtained by the shallow morphology-based classes inspired by Müller et al. (2012). This form of classes is easy to implement in many languages and computationally much cheaper than the Brown clusters. Although less than the data-driven class models, the hybrid suffix/word models also appear to improve translation quality. We can see that fullibm again yields the highest improvements, but we can also observe more consistent trends where longer n-grams help and class-based models are preferable to stream-based models without emission probabilities.

When translating into a morphologically rich language, such as Russian, the role of the target language model is two-fold. On the one hand, it helps choose the correct meaning from the available phrase translation candidates, on the other hand, it helps choose the correct surface realization of the translation candidate that agrees grammatically with the previous target context. For morphologically rich languages the second aspect plays a considerably larger role than for morphologically poor languages. To disentangle these two roles of the language model we also evaluated the different language models with respect to stem-based information only, stripping off any inflectional information using the Snowball stemmer. These results are also reported in Table 3 and in general exhibit the same trend as the surface-based BLEU scores. Again, fullibm performs best, and the original class-based LMs do not lead to any improvements over the baseline. As a general observation, we find that the surface-level gains are most of the time larger than the stem-level ones, which suggests that the additional LMs are mainly improving the choice of word inflections.

All systems compared in Table 3 use a class language model as an additional feature, which is log-linearly interpolated with the other decoder features. Alternatively, the word- and the class-based lan-

(a) Brown clusters (600)

Additional LM	surface BLEU	surface Δ	stem BLEU	stem Δ
⋆ none [baseline]	18.8	—	24.7	—
⋆ 5g class, log-linear comb.	18.9	$+0.1$	24.6	-0.1
⋆ 5g class, linear (global λ)	18.5	–0.3	24.4	-0.3
5g class, linear (class λ's)	18.6	–0.2	24.5	-0.2

(b) Suffixes/words, $\theta = 5000$

Additional LM	surface BLEU	surface Δ	stem BLEU	stem Δ
⋆ none [baseline]	18.8	—	24.7	—
⋆ 5g class, log-linear comb.	19.0	$+0.2^{\circ}$	24.8	$+0.1$
⋆ 5g class, linear (global λ)	18.9	$+0.1$	24.8	$+0.1$
5g class, linear (class λ's)	18.6	-0.1	24.6	-0.1

Table 4: SMT translation quality on newstest13 when using different LM combining frameworks: additional feature in the log-linear combination or linear interpolation with perplexity-tuned weights (one global lambda or class-specific lambdas).

guage models may be linearly interpolated with weights determined by maximizing the likelihood of a held-out monolingual data set (see Section 3.2). While linear interpolation often outperforms log-linear interpolation for combining language models for domain adaptation (Foster and Kuhn, 2007), this does not seem to be the case for language models for morphologically rich target languages. The results presented in Table 4 consistently show that linear interpolation under-performs log-linear combination under all conditions. Even using class-specific interpolation weights as suggested by Müller et al. (2012) did not lead to any further improvements.

5 Conclusion

We have presented the first systematic comparison of different forms of class-based LMs and different class LM combination methods in the context of SMT into a morphologically rich language.

First of all, our results have shown that careful modeling of class-to-word emission probabilities—often omitted from the models used in SMT—is actually important for improving translation quality. In particular, we have achieved best results when using a refined variant of the original class-based LM, called fullibm, which had never been tested for SMT but only for speech recognition (Goodman, 2001). Secondly, we have found that a rather simple LM based on shallow morphology-based classes can get close, in terms of BLEU, to the performance of more computationally expensive data-driven classes. Although the reported improvements are modest, they are statistically significant and obtained in a competitive large-data scenario against a state-of-the-art baseline.

On the downside, and somewhat in contrast with previous findings in domain adaptation, we have observed that linear interpolation of word- and class-based LMs with perplexity-tuned weights performs worse than the log-linear combination of models with model-level weights globally tuned for translation quality. This result was confirmed also when using class-specific lambdas as suggested by Müller et al. (2012).

Indeed, modeling morphologically rich languages remains a challenging problem for SMT but, with our evaluation, we have contributed to assess how far existing language modeling techniques may go in this direction. Natural extensions of this work include combining multiple LMs based on different, and possibly complementary, kinds of classes such as data-driven and suffix-based, or using supervised morphological analyzers instead of a simple stemmer. In a broader perspective, we believe that future research should question the fundamental constraints of n-gram modeling and develop innovative modeling techniques that conform to the specific requirements of translating into morphologically rich languages.

Acknowledgments

This research was funded in part by the Netherlands Organisation for Scientific Research (NWO) under project numbers 639.022.213 and 612.001.218. We kindly thank Thomas Müller for providing code and support for the weight optimization of linearly interpolated models.

References

Waleed Ammar, Victor Chahuneau, Michael Denkowski, Greg Hanneman, Wang Ling, Austin Matthews, Kenton Murray, Nicola Segall, Alon Lavie, and Chris Dyer. 2013. The CMU machine translation systems at WMT 2013: Syntax, synthetic translation options, and pseudo-references. In *Proceedings of the Eighth Workshop on Statistical Machine Translation*, pages 70–77, Sofia, Bulgaria, August. Association for Computational Linguistics.

Lalit R. Bahl, Peter F. Brown, Peter V. de Souza, Robert L. Mercer, and David Nahamoo. 1991. A fast algorithm for deleted interpolation. In *Eurospeech*. ISCA.

Alexandra Birch, Miles Osborne, and Philipp Koehn. 2007. CCG supertags in factored statistical machine translation. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 9–16, Prague, Czech Republic, June. Association for Computational Linguistics.

Arianna Bisazza and Marcello Federico. 2012. Cutting the long tail: Hybrid language models for translation style adaptation. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 439–448, Avignon, France, April. Association for Computational Linguistics.

Ondřej Bojar, Christian Buck, Chris Callison-Burch, Christian Federmann, Barry Haddow, Philipp Koehn, Christof Monz, Matt Post, Radu Soricut, and Lucia Specia. 2013. Findings of the 2013 Workshop on Statistical Machine Translation. In *Proceedings of the Eighth Workshop on Statistical Machine Translation*, pages 1–44, Sofia, Bulgaria, August. Association for Computational Linguistics.

Peter F. Brown, Peter V. deSouza, Robert L. Mercer, Vincent J. Della Pietra, and Jenifer C. Lai. 1992. Class-based n-gram models of natural language. *Computational Linguistics*, 18(4):467–479.

Simon Carter and Christof Monz. 2011. Syntactic discriminative language model rerankers for statistical machine translation. *Machine Translation*, 25(4):317–339.

Victor Chahuneau, Eva Schlinger, Noah A. Smith, and Chris Dyer. 2013. Translating into morphologically rich languages with synthetic phrases. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1677–1687, Seattle, Washington, USA, October. Association for Computational Linguistics.

Stanley F. Chen and Joshua Goodman. 1999. An empirical study of smoothing techniques for language modeling. *Computer Speech and Language*, 4(13):359–393.

Colin Cherry, Robert C. Moore, and Chris Quirk. 2012. On hierarchical re-ordering and permutation parsing for phrase-based decoding. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, pages 200–209, Montréal, Canada, June. Association for Computational Linguistics.

Nadir Durrani, Philipp Koehn, Helmut Schmid, and Alexander Fraser. 2014. Investigating the usefulness of generalized word representations in SMT. In *Proceedings of the 25th International Conference on Computational Linguistics (COLING)*, Dublin, Ireland, August.

Chris Dyer, Kevin Gimpel, Jonathan H. Clark, and Noah A. Smith. 2011. The CMU-ARK German-English Translation System. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 337–343, Edinburgh, Scotland, July. Association for Computational Linguistics.

George Foster and Roland Kuhn. 2007. Mixture-model adaptation for SMT. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 128–135, Prague, Czech Republic, June. Association for Computational Linguistics.

Michel Galley and Christopher D. Manning. 2008. A simple and effective hierarchical phrase reordering model. In *EMNLP '08: Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 848–856, Morristown, NJ, USA. Association for Computational Linguistics.

Michel Galley and Christopher D. Manning. 2009. Quadratic-time dependency parsing for machine translation. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 773–781, Suntec, Singapore, August. Association for Computational Linguistics.

Joshua T. Goodman. 2001. A bit of progress in language modeling. *Computer Speech & Language*, 15(4):403–434.

Spence Green and John DeNero. 2012. A class-based agreement model for generating accurately inflected translations. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, ACL '12, pages 146–155, Stroudsburg, PA, USA. Association for Computational Linguistics.

Saša Hasan, Oliver Bender, and Hermann Ney. 2006. Reranking translation hypotheses using structural properties. In *Proceedings of the EACL'06 Workshop on Learning Structured Information in Natural Language Applications*, pages 41–48, Trento, Italy, April.

Hany Hassan, Khalil Sima'an, and Andy Way. 2007. Supertagged phrase-based statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 288–295, Prague, Czech Republic, June. Association for Computational Linguistics.

Mark Hopkins and Jonathan May. 2011. Tuning as ranking. In *Proceedings of 2011 Conference on Empirical Methods in Natural Language Processing (EMNLP'11)*.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proceedings of HLT-NAACL 2003*, pages 127–133, 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 the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic. Association for Computational Linguistics.

Abby Levenberg and Miles Osborne. 2009. Stream-based randomised language models for smt. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2 - Volume 2*, EMNLP '09, pages 756–764, Stroudsburg, PA, USA. Association for Computational Linguistics.

Christof Monz. 2011. Statistical Machine Translation with Local Language Models. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 869–879, Edinburgh, Scotland, UK., July. Association for Computational Linguistics.

Thomas Müller, Hinrich Schütze, and Helmut Schmid. 2012. A comparative investigation of morphological language modeling for the languages of the European Union. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 386–395, Montréal, Canada, June. Association for Computational Linguistics.

Eric W. Noreen. 1989. *Computer Intensive Methods for Testing Hypotheses. An Introduction*. Wiley-Interscience.

Franz Josef Och. 1999. An efficient method for determining bilingual word classes. In *Proceedings of the 9th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 71–76.

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In Erhard Hinrichs and Dan Roth, editors, *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 160–167.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association of Computational Linguistics (ACL)*, pages 311–318, Philadelphia, PA.

Stefan Riezler and John T. Maxwell. 2005. On some pitfalls in automatic evaluation and significance testing for MT. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 57–64, Ann Arbor, Michigan, June. Association for Computational Linguistics.

Hinrich Schütze and Michael Walsh. 2011. Half-context language models. *Comput. Linguist.*, 37(4):843–865, December.

Lane Schwartz, Chris Callison-Burch, William Schuler, and Stephen Wu. 2011. Incremental syntactic language models for phrase-based translation. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 620–631, Portland, Oregon, USA, June. Association for Computational Linguistics.

Jakob Uszkoreit and Thorsten Brants. 2008. Distributed word clustering for large scale class-based language modeling in machine translation. In *Proceedings of ACL-08: HLT*, pages 755–762, Columbus, Ohio, June. Association for Computational Linguistics.

Philip Williams and Philipp Koehn. 2011. Agreement constraints for statistical machine translation into german. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 217–226, Edinburgh, Scotland, July. Association for Computational Linguistics.

Ian H. Witten and Timothy C. Bell. 1991. The zero-frequency problem: Estimating the probabilities of novel events in adaptive text compression. *IEEE Trans. Inform. Theory*, IT-37(4):1085–1094.

Joern Wuebker, Stephan Peitz, Felix Rietig, and Hermann Ney. 2013. Improving statistical machine translation with word class models. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1377–1381, Seattle, Washington, USA, October. Association for Computational Linguistics.

Ali Yazgan and Murat Saraçlar. 2004. Hybrid language models for out of vocabulary word detection in large vocabulary conversational speech recognition. In *Proceedings of ICASSP*, volume 1, pages I – 745–8 vol.1, may.