BIRA: Improved Predictive Exchange Word Clustering

Jon Dehdari[1,2] and **Liling Tan**[2] and **Josef van Genabith**[1,2]
[1]DFKI, Saarbrücken, Germany
{jon.dehdari,josef.van_genabith}@dfki.de
[2]University of Saarland, Saarbrücken, Germany
liling.tan@uni-saarland.de

Abstract

Word clusters are useful for many NLP tasks including training neural network language models, but current increases in datasets are outpacing the ability of word clusterers to handle them. Little attention has been paid thus far on inducing high-quality word clusters at a large scale. The predictive exchange algorithm is quite scalable, but sometimes does not provide as good perplexity as other slower clustering algorithms.

We introduce the bidirectional, interpolated, refining, and alternating (BIRA) predictive exchange algorithm. It improves upon the predictive exchange algorithm's perplexity by up to 18%, giving it perplexities comparable to the slower two-sided exchange algorithm, and better perplexities than the slower Brown clustering algorithm. Our BIRA implementation is fast, clustering a 2.5 billion token English News Crawl corpus in 3 hours. It also reduces machine translation training time while preserving translation quality. Our implementation is portable and freely available.

1 Introduction

Words can be grouped together into equivalence classes to help reduce data sparsity and better generalize data. Word clusters are useful in many NLP applications. Within machine translation word classes are used in word alignment (Brown et al., 1993; Och and Ney, 2000), translation models (Koehn and Hoang, 2007; Wuebker et al., 2013), reordering (Cherry, 2013), preordering (Stymne, 2012), target-side inflection (Chahuneau et al., 2013), SAMT (Zollmann and Vogel, 2011), and OSM (Durrani et al., 2014), among many others.

Word clusterings have also found utility in parsing (Koo et al., 2008; Candito and Seddah, 2010; Kong et al., 2014), chunking (Turian et al., 2010), NER (Miller et al., 2004; Liang, 2005; Ratinov and Roth, 2009; Ritter et al., 2011), structure transfer (Täckström et al., 2012), and discourse relation discovery (Rutherford and Xue, 2014).

Word clusters also speed up normalization in training neural network and MaxEnt language models, via class-based decomposition (Goodman, 2001a). This reduces the normalization time from $\mathcal{O}(|V|)$ (the vocabulary size) to $\approx \mathcal{O}(\sqrt{|V|})$. More improvements to $\mathcal{O}(\log(|V|))$ are found using hierarchical softmax (Morin and Bengio, 2005; Mnih and Hinton, 2009).

2 Word Clustering

Word clustering partitions a vocabulary V, grouping together words that function similarly. This helps generalize language and alleviate data sparsity. We discuss flat clustering in this paper. Flat, or strict partitioning clustering surjectively maps word types onto a smaller set of clusters.

The **exchange algorithm** (Kneser and Ney, 1993) is an efficient technique that exhibits a general time complexity of $\mathcal{O}(|V| \times |C| \times I)$, where $|V|$ is the number of word types, $|C|$ is the number of classes, and I is the number of training iterations, typically < 20. This omits the specific method of exchanging words, which adds further complexity. Words are exchanged from one class to another until convergence or I.

Proceedings of NAACL-HLT 2016, pages 1169–1174,
San Diego, California, June 12-17, 2016.

One of the oldest and still most popular exchange algorithm implementations is `mkcls` (Och, 1995)[1], which adds various metaheuristics to escape local optima. Botros et al. (2015) introduce their implementation of three exchange-based algorithms. Martin et al. (1998) and Müller and Schütze (2015)[2] use trigrams within the exchange algorithm. Clark (2003) adds an orthotactic bias.[3]

The previous algorithms use an unlexicalized (two-sided) language model: $P(w_i|w_{i-1}) = P(w_i|c_i)\,P(c_i|c_{i-1})$, where the class c_i of the predicted word w_i is conditioned on the class c_{i-1} of the previous word w_{i-1}. Goodman (2001b) altered this model so that c_i is conditioned directly upon w_{i-1}, hence: $P(w_i|w_{i-1}) = P(w_i|c_i)\,P(c_i|w_{i-1})$. This new model fractionates the history more, but it allows for a large speedup in hypothesizing an exchange since the history doesn't change. The resulting partially lexicalized (one-sided) class model gives the accompanying **predictive exchange algorithm** (Goodman, 2001b; Uszkoreit and Brants, 2008) a time complexity of $\mathcal{O}((B+|V|)\times|C|\times I)$ where B is the number of unique bigrams in the training set.[4] We introduce a set of improvements to this algorithm to enable high-quality large-scale word clusters.

3 BIRA Predictive Exchange

We developed a *bidirectional, interpolated, refining, and alternating* (BIRA) predictive exchange algorithm. The goal of BIRA is to produce better clusters by using multiple, changing models to escape local optima. This uses both forward and reversed bigram class models to improve cluster quality by evaluating log-likelihood on two different models. Unlike using trigrams, bidirectional bigram models only linearly increase time and memory requirements, and in fact some data structures can be shared. The two directions are interpolated to allow softer integration of these two models:

$$P(w_i|w_{i-1},w_{i+1}) \triangleq P(w_i|c_i) \cdot (\lambda P(c_i|w_{i-1}) + (1-\lambda)P(c_i|w_{i+1})) \quad (1)$$

The interpolation weight λ for the forward direction alternates to $1-\lambda$ every a iterations (i):

$$\lambda_i := \begin{cases} 1-\lambda_0 & \text{if } \; i \bmod a = 0 \\ \lambda_0 & \text{otherwise} \end{cases} \quad (2)$$

Figure 1 illustrates the benefit of this λ-inversion to help escape local minima, with lower training set perplexity by inverting λ every four iterations:

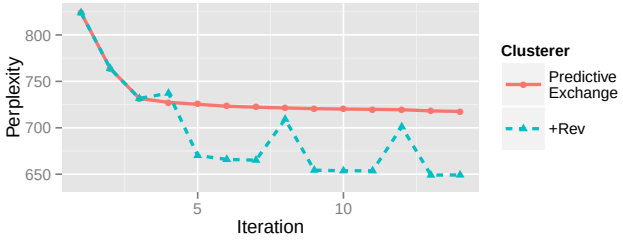

Figure 1: Training set perplexity using lambda inversion (`+Rev`), using 100M tokens of the Russian News Crawl (cf. §4.1). Here $a=4$, $\lambda_0=1$, and $|C|=800$.

The time complexity is $\mathcal{O}(2\times(B+|V|)\times|C|\times I)$. The original predictive exchange algorithm can be obtained by setting $\lambda=1$ and $a=0$.[5]

Another innovation, both in terms of cluster quality *and* speed, is *cluster refinement*. The vocabulary is initially clustered into $|G|$ sets, where $|G| \ll |C|$, typically 2–10. After a few iterations (i) of this, the full partitioning C_f is explored. Clustering G converges very quickly, typically requiring no more than 3 iterations.[6]

$$|C|_i := \begin{cases} |G| & \text{if } \; i \leq 3 \\ |C|_f & \text{otherwise} \end{cases} \quad (3)$$

The intuition behind this is to group words first into broad classes, like nouns, verbs, adjectives, etc. In contrast to divisive hierarchical clustering and coarse-to-fine methods (Petrov, 2009), after the initial iterations, the algorithm is still able to exchange

[1] https://github.com/moses-smt/mgiza

[2] http://cistern.cis.lmu.de/marlin

[3] http://bit.ly/1VJwZ7n

[4] Green et al. (2014) provide a Free implementation of the original predictive exchange algorithm within the Phrasal MT system, at http://nlp.stanford.edu/phrasal. Another implementation is in the Cicada semiring MT system.

[5] The time complexity is $\mathcal{O}((B+|V|)\times|C|\times I)$ if $\lambda=1$.

[6] The piecewise definition could alternatively be conditioned upon a percentage threshold of moved words.

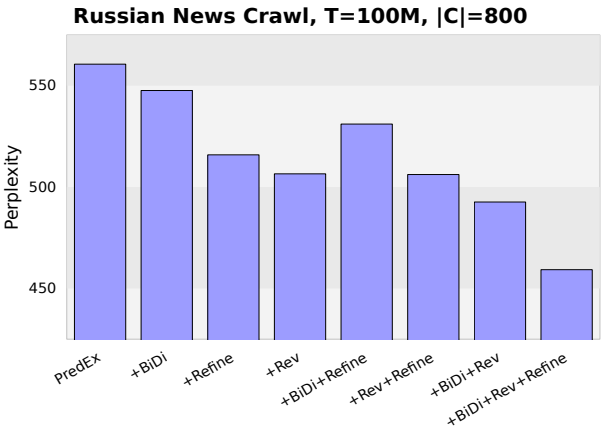

Figure 2: Development set PP of combinations of improvements to predictive exchange (cf. §3), using 100M tokens of the Russian News Crawl, with 800 word classes.

any word to any cluster—there is no hard constraint that the more refined partitions be subsets of the initial coarser partitions. This gives more flexibility in optimizing on log-likelihood, especially given the noise that naturally arises from coarser clusterings. We explored cluster refinement over more stages than just two, successively increasing the number of clusters. We observed no improvement over the two-stage method described above.

Each BIRA component can be applied to any exchange-based clusterer. The contributions of each of these are shown in Figure 2, which reports the development set perplexities (PP) of all combinations of BIRA components over the original predictive exchange algorithm. The data and configurations are discussed in more detail in Section 4. The greatest PP reduction is due to using lambda inversion (`+Rev`), followed by cluster refinement (`+Refine`), then interpolating the bidirectional models (`+BiDi`), with robust improvements by using all three of these—an 18% reduction in perplexity over the predictive exchange algorithm. We have found that both lambda inversion and cluster refinement prevent early convergence at local optima, while bidirectional models give immediate and consistent training set PP improvements, but this is attenuated in a unidirectional evaluation.

We observed that most of the computation for the predictive exchange algorithm is spent on the logarithm function, calculating $\delta \leftarrow \delta - N(w,c) \cdot \log N(w,c)$.[7] Since the codomain of $N(w,c)$ is $\mathbb{N}_0$, and due to the power law distribution of the algorithm's access to these entropy terms, we can precompute $N \cdot \log N$ up to, say 10e+7, with minimal memory requirements.[8] This results in a considerable speedup of around 40%.

4 Experiments

Our experiments consist of both intrinsic and extrinsic evaluations. The intrinsic evaluation measures the perplexity (PP) of two-sided class-based models for English and Russian, and the extrinsic evaluation measures BLEU scores of phrase-based MT of Russian↔English and Japanese↔English texts.

4.1 Class-based Language Model Evaluation

In this task we used 400, 800, and 1200 classes for English, and 800 classes for Russian. The data comes from the 2011–2013 News Crawl monolingual data of the WMT task.[9] For these experiments the data was deduplicated, shuffled, tokenized, digit-conflated, and lowercased. In order to have a large test set, one line per 100 of the resulting (shuffled) corpus was separated into the test set.[10] The minimum count threshold was set to 3 occurrences in the training set. Table 1 shows information on the resulting corpus.

Corpus	Tokens	Types	Lines
English Train	1B	2M	42M
English Test	12M	197K	489K
Russian Train	550M	2.7M	31M
Russian Test	6M	284K	313K

Table 1: Monolingual training & test set sizes.

The clusterings are evaluated on the PP of an external 5-gram *unidirectional two-sided* class-based language model (LM). The *n*-gram-order interpolation weights are tuned using a distinct development set of comparable size and quality as the test set.

Table 2 and Figure 3 show perplexity results using a varying number of classes. Two-sided exchange gives the lowest perplexity across the board, although this is within a two-sided LM evaluation.

[7] δ is the change in log-likelihood, and $N(w,c)$ is the count of a given word followed by a given class.

[8] This was independently discovered in Botros et al. (2015).

[9] `http://www.statmt.org/wmt15/translation-task.html`

[10] The data setup script is at `http://www.dfki.de/~jode03/naacl2016.sh`.

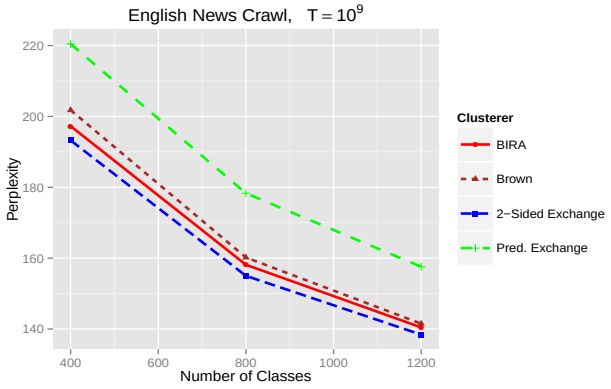

Figure 3: 5-gram two-sided class-based LM perplexities for various clusterers on English News Crawl varying the number of classes.

We also evaluated clusters derived from word2vec (Mikolov et al., 2013) using various configurations[11], and all gave poor perplexities. BIRA gives better perplexities than both the original predictive exchange algorithm and Brown clusters.[12] The Russian experiments yielded higher perplexities for all clusterings, but otherwise the same comparative results.

Training Set	2-Side Ex.	BIRA	Brown	Pred. Ex.
EN, $\|C\| = 400$	193.3	197.3	201.8	220.5
EN, $\|C\| = 800$	155.0	158.1	160.2	178.3
EN, $\|C\| = 1200$	138.4	140.4	141.5	157.6
RU, $\|C\| = 800$	322.4	340.7	350.4	389.3

Table 2: 5-gram two-sided class-based LM perplexities.

In general Brown clusters give slightly worse results relative to exchange-based clusters, since Brown clustering requires an early, permanent placement of frequent words, with further restrictions imposed on the $|C|$-most frequent words (Liang, 2005).[13] Liang-style Brown clustering is only efficient on a small number of clusters, since there is a $|C|^2$ term in its time complexity.

[11]Negative sampling & hierarchical softmax; CBOW & skip-gram; various window sizes; various dimensionalities.

[12]For the two-sided exchange we used `mkcls`; for the original pred. exchange we used Phrasal's clusterer; for Brown clustering we used Percy Liang's brown-cluster (329dc). All had min-count=3, and all but `mkcls` (which is not multithreaded) had threads=12, iterations=15.

[13]Recent work by Derczynski and Chester (2016) loosens some restrictions on Brown clustering.

Training Set	mkcls	BIRA	Brown	Phrasal
EN, $\|C\| = 400$	39.0	1.0	2.3	3.1
EN, $\|C\| = 800$	48.8	1.4	12.5	5.1
EN, $\|C\| = 1200$	68.8	1.7	25.5	6.2
RU, $\|C\| = 800$	75.0	1.5	14.6	5.5

Table 3: Clustering times (hours) of full training sets. Mkcls implements two-sided exchange, and Phrasal implements one-sided predictive exchange.

The original predictive exchange algorithm has a more fractionated history than the two-sided exchange algorithm. Interestingly, increasing the number of clusters causes a convergence in the word clusterings themselves, while also causing a divergence in the time complexities of these two varieties of the exchange algorithm. The metaheuristic techniques employed by the two-sided clusterer `mkcls` can be applied to other exchange-based clusterers—including ours—for further improvements.

Table 3 presents wall clock times using the full training set, varying the number of word classes $|C|$ (for English).[14] The predictive exchange-based clusterers (BIRA and Phrasal) exhibit slow increases in time as the number of classes increases, while the others (Brown and `mkcls`) are much more sensitive to $|C|$. Our BIRA-based clusterer is three times faster than Phrasal for all these sets.

We performed an additional experiment, adding more English News Crawl training data.[15] *Our implementation took 3.0 hours to cluster 2.5 billion training tokens*, with $|C| = 800$ using modest hardware.[14]

4.2 Machine Translation Evaluation

We also evaluated the BIRA predictive exchange algorithm extrinsically in machine translation. As discussed in Section 1, word clusters are employed in a variety of ways within machine translation systems, the most common of which is in word alignment where `mkcls` is widely used. As training sets get larger every year, `mkcls` struggles to keep pace, and

[14]All time experiments used a 2.4 GHz Opteron 8378 featuring 16 threads.

[15]Adding years 2008–2010 and 2014 to the existing training data. This training set was too large for the external class-based LM to fit into memory, so no perplexity evaluation of this clustering was possible.

is a substantial time bottleneck in MT pipelines with large datasets.

We used data from the Workshop on Machine Translation 2015 (WMT15) Russian↔English dataset and the Workshop on Asian Translation 2014 (WAT14) Japanese↔English dataset (Nakazawa et al., 2014). Both pairs used standard configurations, like truecasing, MeCab segmentation for Japanese, MGIZA alignment, grow-diag-final-and phrase extraction, phrase-based Moses, quantized KenLM 5-gram modified Kneser-Ney LMs, and MERT tuning.

$\|C\|$	EN-RU	RU-EN	EN-JA	JA-EN
10	20.8→20.9*	26.2→26.0	23.5→23.4	16.9→16.8
50	21.0→21.2*	25.9→25.7	24.0→23.7*	16.9→16.9
100	20.4→21.1	25.9→25.8	23.8→23.5	16.9→17.0
200	21.0→20.8	25.8→25.9	23.8→23.4	17.0→16.8
500	20.9→20.9	25.8→25.9*	24.0→23.8	16.8→17.1*
1000	20.9→21.1	25.9→26.0**	23.6→23.5	16.9→17.1

Table 4: BLEU scores (`mkcls`→BIRA) and significance across cluster sizes ($|C|$).

The BLEU score differences between using `mkcls` and our BIRA implementation are small but there are a few statistically significant changes, using bootstrap resampling (Koehn, 2004). Table 4 presents the BLEU score changes across varying cluster sizes (*: p-value < 0.05, **: p-value < 0.01). MERT tuning is quite erratic, and some of the BLEU differences could be affected by noise in the tuning process in obtaining quality weight values. Using our BIRA implementation reduces the translation model training time with 500 clusters from 20 hours using `mkcls` (of which 60% of the time is spent on clustering) to just 8 hours (of which 5% is spent on clustering).

5 Conclusion

We have presented improvements to the predictive exchange algorithm that address longstanding drawbacks of the original algorithm compared to other clustering algorithms, enabling new directions in using large scale, high cluster-size word classes in NLP.

Botros et al. (2015) found that the one-sided model of the predictive exchange algorithm produces better results for training LSTM-based language models compared to two-sided models, while two-sided models generally give better perplexity in class-based LM experiments. Our paper shows that BIRA-based predictive exchange clusters are competitive with two-sided clusters even in a two-sided evaluation. They also give better perplexity than the original predictive exchange algorithm and Brown clustering.

The software is freely available at `https://github.com/jonsafari/clustercat`.

Acknowledgements

We would like to thank Hermann Ney and Kazuki Irie, as well as the reviewers for their useful comments. This work was supported by the QT21 project (Horizon 2020 No. 645452).

References

Rami Botros, Kazuki Irie, Martin Sundermeyer, and Hermann Ney. 2015. On Efficient Training of Word Classes and their Application to Recurrent Neural Network Language Models. In *Proceedings of INTERSPEECH-2015*, pages 1443–1447, Dresden, Germany.

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

Marie Candito and Djamé Seddah. 2010. Parsing Word Clusters. In *Proceedings of the NAACL HLT 2010 First Workshop on Statistical Parsing of Morphologically-Rich Languages*, pages 76–84, Los Angeles, CA, USA.

Victor Chahuneau, Eva Schlinger, Noah A. Smith, and Chris Dyer. 2013. Translating into Morphologically Rich Languages with Synthetic Phrases. In *Proceedings of EMNLP*, pages 1677–1687, Seattle, WA, USA.

Colin Cherry. 2013. Improved Reordering for Phrase-Based Translation using Sparse Features. In *Proceedings of NAACL-HLT*, pages 22–31, Atlanta, GA, USA.

Alexander Clark. 2003. Combining Distributional and Morphological Information for Part of Speech Induction. In *Proceedings of EACL*, pages 59–66.

Leon Derczynski and Sean Chester. 2016. Generalised Brown Clustering and Roll-up Feature Generation. In *Proceedings of AAAI*, Phoenix, AZ, USA.

Nadir Durrani, Philipp Koehn, Helmut Schmid, and Alexander Fraser. 2014. Investigating the Usefulness of Generalized Word Representations in SMT. In *Proceedings of Coling*, pages 421–432, Dublin, Ireland.

Joshua Goodman. 2001a. Classes for Fast Maximum Entropy Training. In *Proceedings of ICASSP*, pages 561–564.

Joshua T. Goodman. 2001b. A Bit of Progress in Language Modeling, Extended Version. Technical Report MSR-TR-2001-72, Microsoft Research.

Spence Green, Daniel Cer, and Christopher Manning. 2014. An Empirical Comparison of Features and Tuning for Phrase-based Machine Translation. In *Proc. of WMT*, pages 466–476, Baltimore, MD, USA.

Reinhard Kneser and Hermann Ney. 1993. Improved clustering techniques for class-based statistical language modelling. In *Proceedings of EUROSPEECH'93*, pages 973–976, Berlin, Germany.

Philipp Koehn and Hieu Hoang. 2007. Factored Translation Models. In *Proceedings of EMNLP-CoNLL*, pages 868–876, Prague, Czech Republic.

Philipp Koehn. 2004. Statistical significance tests for machine translation evaluation. In *Proceedings of EMNLP*, pages 388–395.

Lingpeng Kong, Nathan Schneider, Swabha Swayamdipta, Archna Bhatia, Chris Dyer, and Noah A. Smith. 2014. A Dependency Parser for Tweets. In *Proceedings of EMNLP*, pages 1001–1012, Doha, Qatar.

Terry Koo, Xavier Carreras, and Michael Collins. 2008. Simple Semi-supervised Dependency Parsing. In *Proceedings of ACL: HLT*, pages 595–603, Columbus, OH, USA.

Percy Liang. 2005. Semi-Supervised Learning for Natural Language. Master's thesis, MIT.

Sven Martin, Jörg Liermann, and Hermann Ney. 1998. Algorithms for Bigram and Trigram Word Clustering. *Speech Communication*, 24(1):19–37.

Tomáš Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. In *Workshop Proceedings of the International Conference on Learning Representations (ICLR)*, Scottsdale, AZ, USA.

Scott Miller, Jethran Guinness, and Alex Zamanian. 2004. Name Tagging with Word Clusters and Discriminative Training. In Susan Dumais, Daniel Marcu, and Salim Roukos, editors, *Proceedings of HLT-NAACL*, pages 337–342, Boston, MA, USA.

Andriy Mnih and Geoffrey Hinton. 2009. A Scalable Hierarchical Distributed Language Model. In D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, editors, *Advances in NIPS-21*, volume 21, pages 1081–1088.

Frederic Morin and Yoshua Bengio. 2005. Hierarchical Probabilistic Neural Network Language Model. In *Proceedings of AISTATS*, volume 5, pages 246–252.

Thomas Müller and Hinrich Schütze. 2015. Robust Morphological Tagging with Word Representations. In *Proceedings of NAACL*, pages 526–536, Denver, CO, USA.

Toshiaki Nakazawa, Hideya Mino, Isao Goto, Sadao Kurohashi, and Eiichiro Sumita. 2014. Overview of the first Workshop on Asian Translation. In *Proceedings of the Workshop on Asian Translation (WAT)*.

Franz Josef Och and Hermann Ney. 2000. A Comparison of Alignment Models for Statistical Machine Translation. In *Proceedings of Coling*, pages 1086–1090, Saarbrücken, Germany.

Franz Josef Och. 1995. Maximum-Likelihood-Schätzung von Wortkategorien mit Verfahren der kombinatorischen Optimierung. Bachelor's thesis (Studienarbeit), Friedrich-Alexander-Universität Erlangen-Nürnburg, Germany.

Slav Petrov. 2009. *Coarse-to-Fine Natural Language Processing*. Ph.D. thesis, University of California at Berkeley, Berkeley, CA, USA.

Lev Ratinov and Dan Roth. 2009. Design Challenges and Misconceptions in Named Entity Recognition. In *Proc. of CoNLL*, pages 147–155, Boulder, CO, USA.

Alan Ritter, Sam Clark, Mausam, and Oren Etzioni. 2011. Named Entity Recognition in Tweets: An Experimental Study. In *Proceedings of EMNLP 2011*, pages 1524–1534, Edinburgh, Scotland.

Attapol Rutherford and Nianwen Xue. 2014. Discovering Implicit Discourse Relations Through Brown Cluster Pair Representation and Coreference Patterns. In *Proc. of EACL*, pages 645–654, Gothenburg, Sweden.

Sara Stymne. 2012. Clustered Word Classes for Preordering in Statistical Machine Translation. In *Proceedings of the Joint Workshop on Unsupervised and Semi-Supervised Learning in NLP*, pages 28–34, Avignon, France.

Oscar Täckström, Ryan McDonald, and Jakob Uszkoreit. 2012. Cross-lingual Word Clusters for Direct Transfer of Linguistic Structure. In *Proceedings of NAACL:HLT*, pages 477–487, Montréal, Canada.

Joseph Turian, Lev-Arie Ratinov, and Yoshua Bengio. 2010. Word Representations: A Simple and General Method for Semi-Supervised Learning. In *Proceedings of ACL*, pages 384–394, Uppsala, Sweden.

Jakob Uszkoreit and Thorsten Brants. 2008. Distributed Word Clustering for Large Scale Class-Based Language Modeling in Machine Translation. In *Proc. of ACL: HLT*, pages 755–762, Columbus, OH, USA.

Joern Wuebker, Stephan Peitz, Felix Rietig, and Hermann Ney. 2013. Improving Statistical Machine Translation with Word Class Models. In *Proceedings of EMNLP*, pages 1377–1381, Seattle, WA, USA.

Andreas Zollmann and Stephan Vogel. 2011. A Word-Class Approach to Labeling PSCFG Rules for Machine Translation. In *Proceedings of ACL-HLT*, pages 1–11, Portland, OR, USA.