Inverted indexing for cross-lingual NLP

Anders Søgaard* **Željko Agić*** **Héctor Martínez Alonso***
Barbara Plank* **Bernd Bohnet†** **Anders Johannsen***
*Center for Language Technology, University of Copenhagen, Denmark
†Google, London, United Kingdom
soegaard@hum.ku.dk

Abstract

We present a novel, count-based approach to obtaining inter-lingual word representations based on inverted indexing of Wikipedia. We present experiments applying these representations to 17 datasets in document classification, POS tagging, dependency parsing, and word alignment. Our approach has the advantage that it is simple, computationally efficient and almost parameter-free, and, more importantly, it enables multi-source cross-lingual learning. In 14/17 cases, we improve over using state-of-the-art bilingual embeddings.

1 Introduction

Linguistic resources are hard to come by and unevenly distributed across the world's languages. Consequently, transferring linguistic resources or knowledge from one language to another has been identified as an important research problem. Most work on cross-lingual transfer has used English as the source language. There are two reasons for this; namely, the availability of English resources and the availability of parallel data for (and translations between) English and most other languages.

In cross-lingual syntactic parsing, for example, two approaches to cross-lingual learning have been explored, namely annotation projection and delexicalized transfer. Annotation projection (Hwa et al., 2005) uses word-alignments in human translations to project predicted source-side analyses to the target language, producing a noisy syntactically annotated resource for the target language. On the other hand, delexicalized transfer (Zeman and Resnik, 2008; McDonald et al., 2011; Søgaard, 2011) simply removes lexical features from mono-lingual parsing models, but assumes reliable POS tagging for the target language. Delexicalized transfer works particularly well when resources from several source languages are used for training; learning from multiple other languages prevents over-fitting to the peculiarities of the source language. Some authors have also combined annotation projection and delexicalized transfer, e.g., McDonald et al. (2011). Others have tried to augment delexicalized transfer models with bilingual word representations (Täckström et al., 2013; Xiao and Guo, 2014).

In cross-lingual POS tagging, mostly annotation projection has been explored (Fossum and Abney, 2005; Das and Petrov, 2011), since all features in POS tagging models are typically lexical. However, using bilingual word representations was recently explored as an alternative to projection-based approaches (Gouws and Søgaard, 2015).

The major drawback of using bi-lexical representations is that it limits us to using a single source language. Täckström et al. (2013) obtained significant improvements using bilingual word clusters over a single source delexicalized transfer model, for example, but even better results were obtained with delexicalized transfer in McDonald et al. (2011) by simply using several source languages.

This paper introduces a simple method for obtaining *truly* inter-lingual word representations in order to train models with lexical features on several source languages at the same time. Briefly put, we represent words by their occurrence in clusters of Wikipedia articles linking to the same concept. Our representations are competitive with

Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics
and the 7th International Joint Conference on Natural Language Processing, pages 1713–1722,
Beijing, China, July 26-31, 2015. ©2015 Association for Computational Linguistics

state-of-the-art neural net word embeddings when using only a single source language, but also enable us to exploit the availability of resources in multiple languages. This also makes it possible to explore multi-source transfer for POS tagging. We evaluate the method across POS tagging and dependency parsing datasets in four languages in the Google Universal Treebanks v. 1.0 (see §3.2.1), as well as two document classification datasets and four word alignment problems using a hand-aligned text. Finally, we also directly compare our results to Xiao and Guo (2014) on parsing data for four languages from CoNLL 2006 and 2007.

Contribution

- We present a novel approach to cross-lingual word representations with several advantages over existing methods: (a) It does not require training neural networks, (b) it does not rely on the availability of parallel data between source and target language, and (c) it enables multi-source transfer with lexical representations.
- We present an evaluation of our inter-lingual word representations, based on inverted indexing, across four tasks: document classification, POS tagging, dependency parsing, and word alignment, comparing our representations to two state-of-the-art neural net word embeddings. For the 17 datasets, for which we can make this comparison, our system is better than these embedding models on 14 datasets. The word representations are made publicly available at `https://bitbucket.org/lowlands/`

2 Distributional word representations

Most NLP models rely on lexical features. Encoding the presence of words leads to high-dimensional and sparse models. Also, simple bag-of-words models fail to capture the relatedness of words. In many tasks, synonymous words should be treated alike, but their bag-of-words representations are as different as those of *dog* and *therefore*.

Distributional word representations are supposed to capture distributional similarities between words. Intuitively, we want similar words to have similar representations. Known approaches focus on different kinds of similarity, some more syntactic, some more semantic. The representations are typically either clusters of distributionally similar words, e.g., Brown et al. (1992), or vector representations. In this paper, we focus on vector representations. In vector-based approaches, similar representations are vectors close in some multi-dimensional space.

2.1 Count-based and prediction-based representations

There are, briefly put, two approaches to inducing vector-based distributional word representations from large corpora: count-based and prediction-based approaches (Baroni et al., 2014). Count-based approaches represent words by their co-occurrences. Dimensionality reduction is typically performed on a raw or weighted co-occurrence matrix using methods such as singular value decomposition (SVD), a method for maximizing the variance in a dataset in few dimensions. In our inverted indexing, we use raw co-occurrence data.

Prediction-based methods use discriminative learning techniques to learn how to predict words from their context, or vice versa. They rely on a neural network architecture, and once the network converges, they use word representations from a middle layer as their distributional representations. Since the network learns to predict contexts from this representation, words occurring in the same contexts will get similar representations. In §2.1.2, we briefly introduce the skip-gram and CBOW models (Mikolov et al., 2013; Collobert and Weston, 2008).

Baroni et al. (2014) argue in favor of prediction-based representations, but provide little explanation why prediction-based representations should be better. One key finding, however, is that prediction-based methods tend to be more robust than count-based methods, and one reason for this seems to be better regularization.

2.1.1 Monolingual representations

Count-based representations rely on co-occurrence information in the form of binary matrices, raw counts, or point-wise mutual information (PMI). The PMI between two words is

$$P(w_i; w_j) = \log \frac{P(w_i \mid w_j)}{P(w_i)}$$

and PMI representations associate a word w_i with a vector of its PMIs with all other words w_j. Dimensionality reduction is typically performed using SVD. We will refer to two prediction-based approaches to learning word vectors, below: the

	KLEMENTIEV	CHANDAR	INVERTED
es			
coche ('car', NOUN)	approximately beyond upgrading	car bicycle cars	driving car cars
expressed ('expressed', VERB)	1.61 55.8 month-to-month	reiterates reiterating confirming	exists defining example
teléfono ('phone', NOUN)	alexandra davison creditor	phone telephone e-mail	phones phone telecommunication
árbol ('tree', NOUN)	tree market-oriented assassinate	tree bread wooden	tree trees grows
escribió ('wrote', VERB)	wrote alleges testified	wrote paul palace	wrote inspired inspiration
amarillo ('yellow', ADJ)	yellow louisiana 1911	crane grabs outfit	colors yellow oohs
de			
auto ('car', NOUN)			car cars camaro
ausgedrückt ('expressed', VERB)			adjective decimal imperative
fr			
voiture ('car', NOUN)			mercedes-benz cars quickest
exprimé ('expressed', VERB)			simultaneously instead possible
téléphone ('phone', NOUN)			phone create allowing
arbre ('tree', NOUN)			tree trees grows
écrit ('wrote', VERB)			published writers books
jaune ('yellow', ADJ)			classification yellow stages
sv			
bil ('car', NOUN)			cars car automobiles
uttryckte ('expressed', VERB)			rejected threatening unacceptable
telefon ('phone', NOUN)			telephones telephone share
träd ('tree', NOUN)			trees tree trunks
skrev ('wrote', VERB)			death wrote biography
gul ('yellow', ADJ)			greenish bluish colored

Table 1: Three nearest neighbors in the English training data of six words occurring in the Spanish test data, in the embeddings used in our experiments. Only 2/6 words were in the German data.

skip-gram model and CBOW. The two models both rely on three level architectures with input, output and a middle layer for intermediate target word representations. The major difference is that skip-gram uses the target word as input and the context as output, whereas the CBOW model does it the other way around. Learning goes by back-propagation, and random target words are used as negative examples. Levy and Goldberg (2014) show that prediction-based representations obtained with the skip-gram model can be related to count-based ones obtained with PMI. They argue that which is best, varies across tasks.

2.1.2 Bilingual representations

Klementiev et al. (2012) learn distinct embedding models for the source and target languages, but while learning to minimize the sum of the two models' losses, they jointly learn a regularizing interaction matrix, enforcing word pairs aligned in parallel text to have similar representations. Note that Klementiev et al. (2012) rely on word-aligned parallel text, and thereby on a large-coverage soft mapping of source words to target words. Other approaches rely on small coverage dictionaries with hard 1:1 mappings between words. Klementiev et al. (2012) do not use skip-gram or CBOW, but the language model presented in Bengio et al. (2003).

Chandar et al. (2014) also rely on sentence-aligned parallel text, but do not make use of word alignments. They begin with bag-of-words representations of source and target sentences. They then use an auto-encoder architecture. Auto-encoders for document classification typically try to reconstruct bag-of-words input vectors at the output layer, using back-propagation, passing the representation through a smaller middle layer. This layer then provides a dimensionality reduction. Chandar et al. (2014) instead replace the output layer with the target language bag-of-words reconstruction. In their final set-up, they simultaneously minimize the loss of a source-source, a target-target, a source-target, and a target-source auto-encoder, which corresponds to training a single auto-encoder with randomly chosen instances from source-target pairs. The bilingual word vectors can now be read off the auto-encoder's middle layer.

Xiao and Guo (2014) use a CBOW model and random target words as negative examples. The trick they introduce to learn bilingual embeddings, relies on a bilingual dictionary, in their case obtained from Wiktionary. They only use the unambiguous translation pairs for the source and target languages in question and simply force translation equivalents to have the same representation. This corresponds to replacing words from unambiguous translation pairs with a unique dummy symbol.

Gouws and Søgaard (2015) present a much simpler approach to learning prediction-based bilingual representations. They assume a list of source-target pivot word pairs that should obtain similar representations, i.e., translations or words with similar representations in some knowledge base. They then present a generative model for constructing a mixed language corpus by randomly selecting sentences from source and target corpora, and randomly replacing pivot words with their equivalent in the other language. They show that running the CBOW model on such a mixed corpus suffices to learn competitive bilingual embeddings. Like Xiao and Guo (2014), Gouws and Søgaard (2015) only use unambiguous translation pairs.

There has, to the best of our knowledge, been no previous work on count-based approaches to bilingual representations.

2.2 Inverted indexing

In this paper, we introduce a new count-based approach, INVERTED, to obtaining cross-lingual word representations using inverted indexing, comparing it with bilingual word representations learned using discriminative techniques. The main advantage of this approach, apart for its simplicity, is that it provides *truly* inter-lingual representations.

Our idea is simple. Wikipedia is a cross-lingual, crowd-sourced encyclopedia with more than 35 million articles written in different languages. At the time of writing, Wikipedia contains more than 10,000 articles in 129 languages. 52 languages had more than 100,000 articles. Several articles are written on the same topic, but in different languages, and these articles all link to the same node in the Wikipedia ontology, the same Wikipedia concept. If for a set of languages, we identify the common subset of Wikipedia concepts, we can thus describe each concept by the set of terms used in the corresponding articles. Each term set will include terms from each of the different languages.

We can now present a word by the corresponding row in the inverted indexing of this concept-to-term set matrix. Instead of representing a Wikipedia concept by the terms used across languages to describe it, we describe a word by the Wikipedia concepts it is used to describe. Note that because of the cross-lingual concepts, this vector representation is by definition cross-lingual. So, for example, if the word *glasses* is used in the English Wikipedia article on Harry Potter, and the English Wikipedia article on Google, and the word *Brille* occurs in the corresponding German ones, the two words are likely to get similar representations.

In our experiments, we use the common subset of available German, English, French, Spanish, and Swedish Wikipedia dumps.[1] We leave out words occurring in more than 5000 documents and perform dimensionality reduction using stochastic, two-pass, rank-reduced SVD - specifically, the latent semantic indexing implementation in Gensim using default parameters.[2]

2.3 Baseline embeddings

We use the word embedding models of Klementiev et al. (2012)[3] (KLEMENTIEV), and Chandar et al. (2014) (CHANDAR) as baselines in the experiments below. We also ran some of our experiments with the embeddings provided by Gouws and Søgaard (2015), but results were very similar to Chandar et al. (2014). We compare the nearest cross-language neighbors in the various representations in Table 1. Specifically, we selected five words from the Spanish test data and searched for its three nearest neighbors in KLEMENTIEV, CHANDAR and INVERTED. The nearest neighbors are presented left to right. We note that CHANDAR and INVERTED seem to contain less noise. KLEMENTIEV is the only model that relies on word-alignments. Whether the noise originates from alignments, or just model differences, is unclear to us.

2.4 Parameters of the word representation models

For KLEMENTIEV and CHANDAR, we rely on embeddings provided by the authors. The only parameter in inverted indexing is the fixed dimensionality in SVD. Our baseline models use 40 dimensions. In document classification, we also use 40 dimensions, but for POS tagging and dependency parsing, we tune the dimensionality parameter $\delta \in \{40, 80, 160\}$ on Spanish development data when possible. For document clas-

[1] `https://sites.google.com/site/rmyeid/projects/polyglot`

[2] `http://radimrehurek.com/gensim/`

[3] `http://klementiev.org/data/distrib/`

	TRAIN		TEST		TOKEN COVERAGE		
lang	data points	tokens	data points	tokens	KLEMENTIEV	CHANDAR	INVERTED
RCV – DOCUMENT CLASSIFICATION							
en	10000	–	–	–	0.314	0.314	0.779
de	–	–	4998	–	0.132	0.132	0.347
AMAZON – DOCUMENT CLASSIFICATION							
en	6000	–	–	–	0.314	0.314	0.779
de	–	–	6000	–	0.132	0.132	0.347
GOOGLE UNIVERSAL TREEBANKS – POS TAGGING & DEPENDENCY PARSING							
en	39.8k	950k	2.4k	56.7k	–	–	–
de	2.2k	30.4k	1.0k	16.3k	0.886	0.884	0.587
es	3.3k	94k	0.3k	8.3k	0.916	0.916	0.528
fr	3.3k	74.9k	0.3k	6.9k	0.888	0.888	0.540
sv	4.4k	66.6k	1.2k	20.3k	n/a	n/a	0.679
CONLL 07 – DEPENDENCY PARSING							
en	18.6	447k	–	–	–	–	–
es	–	–	206	5.7k	0.841	0.841	0.455
de	–	–	357	5.7k	0.616	0.612	0.294
sv	–	–	389	5.7k	n/a	n/a	0.561
EUROPARL – WORD ALIGNMENT							
en	–	–	100	–	0.370	0.370	0.370
es	–	–	100	–	0.533	0.533	0.533

Table 2: Characteristics of the data sets. Embeddings coverage (token-level) for KLEMENTIEV, CHANDAR and INVERTED on the test sets. We use the common vocabulary on WORD ALIGNMENT.

sification and word alignment, we fix the number of dimensions to 40. For both our baselines and systems, we also tune a scaling factor $\sigma \in \{1.0, 0.1, 0.01, 0.001\}$ for POS tagging and dependency parsing, using the scaling method from Turian et al. (2010), also used in Gouws and Søgaard (2015). We do not scale our embeddings for document classification or word alignment.

3 Experiments

The data set characteristics are found in Table 2.3.

3.1 Document classification

Data Our first document classification task is topic classification on the cross-lingual multi-domain sentiment analysis dataset AMAZON in Prettenhofer and Stein (2010).[4] We represent each document by the average of the representations of those words that we find both in the documents and in our embeddings. Rather than classifying reviews by sentiment, we classify by topic, trying to discriminate between book reviews, music reviews and DVD reviews, as a three-way classification problem, training on English and testing on German. Unlike in the other tasks below, we always use unscaled word representations, since these are our only features. All word representations have 40 dimensions.

The other document classification task is a four-way classification problem distinguishing between four topics in RCV corpus.[5] See Klementiev et al. (2012) for details. We use exactly the same set-up as for AMAZON.

Baselines We use the default parameters of the implementation of logistic regression in Sklearn as our baseline.[6] The feature representation is the average embedding of non-stopwords in KLEMENTIEV, resp., CHANDAR. Out-of-vocabulary words do not affect the feature representation of the documents.

System For our system, we replace the above neural net word embeddings with INVERTED representations. Again, out-of-vocabulary words do not affect the feature representation of the documents.

3.2 POS tagging

Data We use the coarse-grained part-of-speech annotations in the Google Universal Treebanks v. 1.0

[4] http://www.webis.de/research/corpora/

[5] http://www.ml4nlp.de/code-and-data

[6] http://scikit-learn.org/stable/

(McDonald et al., 2013).[7] Out of the languages in this set of treebanks, we focus on five languages (de, en, es, fr, sv), with English only used as training data. Those are all treebanks of significant size, but more importantly, we have baseline embeddings for four of these languages, as well as tag dictionaries (Li et al., 2012) needed for the POS tagging experiments.

Baselines One baseline method is a type-constrained structured perceptron with only ortographic features, which are expected to transfer across languages. The type constraints come from Wiktionary, a crowd-sourced tag dictionary.[8] Type constraints from Wiktionary were first used by Li et al. (2012), but note that their set-up is unsupervised learning. Täckström et al. (2013) also used type constraints in a supervised set-up. Our learning algorithm is the structured perceptron algorithm originally proposed by Collins (2002). In our POS tagging experiments, we always do 10 passes over the data. We also present two other baselines, where we augment the feature representation with different embeddings for the target word, KLEMENTIEV and CHANDAR. With all the embeddings in POS tagging, we assign a mean vector to out-of-vocabulary words.

System For our system, we simply augment the delexicalized POS tagger with the INVERTED distributional representation of the current word. The best parameter setting on Spanish development data was $\sigma = 0.01, \delta = 160$.

3.3 Dependency parsing

Data We use the same treebanks from the Google Universal Treebanks v. 1.0 as used in our POS tagging experiments. We again use the Spanish development data for parameter tuning. For compatibility with Xiao and Guo (2014), we also present results on CoNLL 2006 and 2007 treebanks for languages for which we had baseline and system word representations (de, es, sv). Our parameter settings for these experiments were the same as those tuned on the Spanish development data from the Google Universal Treebanks v. 1.0.

Baselines The most obvious baseline in our experiments is delexicalized transfer (DELEX) (McDonald et al., 2011; Søgaard, 2011). This baseline system simply learns models without lexical features. We use a modified version of the first-order Mate parser (Bohnet, 2010) that also takes continuous-valued embeddings as input an disregards features that include lexical items.

For our embeddings baselines, we augment the feature space by adding embedding vectors for head h and dependent d. We experimented with different versions of combining embedding vectors, from firing separate h and d per-dimension features (Bansal et al., 2014) to combining their information. We found that combining the embeddings of h and d is effective and consistently use the absolute difference between the embedding vectors, since that worked better than addition and multiplication on development data.

Delexicalized transfer (DELEX) uses three (3) iterations over the data in both the single-source and the multi-source set-up, a parameter set on the Spanish development data. The remaining parameters were obtained by averaging over performance with different embeddings on the Spanish development data, obtaining: $\sigma = 0.005, \delta = 20, i = 3$, and absolute difference for vector combination. With all the embeddings in dependency parsing, we assign a POS-specific mean vector to out-of-vocabulary words, i.e., the mean of vectors for words with the input word's POS.

System We use the same parameters as those used for our baseline systems. In the single-source set-up, we use absolute difference for combining vectors, while addition in the multi-source set-up.

3.4 Word alignment

Data We use the manually word-aligned English-Spanish Europarl data from Graca et al. (2008). The dataset contains 100 sentences. The annotators annotated whether word alignments were certain or possible, and we present results with *all* word alignments and with only the certain ones. See Graca et al. (2008) for details.

Baselines For word alignment, we simply align every aligned word in the gold data, for which we have a word embedding, to its (Euclidean) nearest neighbor in the target sentence. We evaluate this strategy by its precision (P@1).

System We compare INVERTED with KLEMENTIEV and CHANDAR. To ensure a fair comparison, we use the subset of words covered by all three embeddings.

[7] http://code.google.com/p/uni-dep-tb/

[8] https://code.google.com/p/wikily-supervised-pos-tagger/

		de	es	fr	sv	av-sv
		EN→TARGET				
EMBEDS	K12	80.20	73.16	47.69	-	67.02
	C14	74.85	83.03	48.24	-	68.71
INVERTED	SVD	**81.18**	82.12	49.68	78.72	70.99
		MULTI-SOURCE→TARGET				
INVERTED	SVD	80.10	**84.69**	49.68	78.72	70.66

Table 4: POS tagging (accuracies), K12: KLEMENTIEV, C14: CHANDAR. Parameters tuned on development data: $\sigma = 0.01, \delta = 160$. Iterations not tuned ($i = 10$). Averages do not include Swedish, for comparability.

Dataset	KLEMENTIEV	CHANDAR	INVERTED
AMAZON	0.32	0.36	**0.49**
RCV	0.75	**0.90**	0.55

Table 3: Document classification results (F_1-scores)

		UAS		
		de	es	sv
		EN→TARGET		
DELEX	-	44.78	47.07	56.75
DELEX-XIAO	-	46.24	52.05	57.79
EMBEDS	K12	44.77	47.31	-
	C14	44.32	47.56	
INVERTED	-	45.01	47.45	56.15
XIAO	-	49.54	55.72	61.88

Table 6: Dependency parsing for CoNLL 2006/2007 datasets. Parameters same as on the Google Universal Treebanks.

4 Results

4.1 Document classification

Our document classification results in Table 3 are mixed, but we note that both Klementiev et al. (2012) and Chandar et al. (2014) developed their methods using development data from the RCV corpus. It is therefore not surprising that they obtain good results on this data. On AMAZON, INVERTED is superior to both KLEMENTIEV and CHANDAR.

4.2 POS tagging

In POS tagging, INVERTED leads to significant improvements over using KLEMENTIEV and CHANDAR. See Table 4 for results. Somewhat surprisingly, we see no general gain from using multiple source languages. This is very different from what has been observed in dependency parsing (McDonald et al., 2011), but may be explained by treebank sizes, language similarity, or the noise introduced by the word representations.

4.3 Dependency parsing

In dependency parsing, distributional word representations do not lead to significant improvements, but while KLEMENTIEV and CHANDAR hurt performance, the INVERTED representations lead to small improvements on some languages. The fact that improvements are primarily seen on Spanish suggest that our approach is parameter-sensitive. This is in line with previous observations that count-based methods are more parameter-sensitive than prediction-based ones (Baroni et al., 2014).

For comparability with Xiao and Guo (2014), we also did experiments with the CoNLL 2006 and CoNLL 2007 datasets for which we had embeddings (Table 6). Again, we see little effects from using the word representations, and we also see that our baseline model is weaker than the one in Xiao and Guo (2014) (DELEX-XIAO). See §5 for further discussion.

4.4 Word alignment

The word alignment results are presented in Table 7. On the certain alignments, we see an accuracy of more than 50% with INVERTED in one case. KLEMENTIEV and CHANDAR have the advantage of having been trained on the English-Spanish Europarl data, but nevertheless we see consistent improvements with INVERTED over their off-the-shelf embeddings.

		UAS				LAS			
		de	es	fr	sv	de	es	fr	sv
EN→TARGET									
Delex	-	56.26	62.11	64.30	66.61	48.24	53.01	54.98	56.93
Embeds	K12	56.47	61.92	61.51	-	48.26	52.88	51.76	-
	C14	56.19	61.97	62.95	-	48.11	52.97	53.90	-
Inverted	-	56.18	61.71	63.81	66.54	48.82	53.04	54.81	57.18
Multi-source→target									
Delex	-	**56.80**	63.21	66.00	**67.49**	**49.32**	54.77	56.53	**57.86**
Inverted	-	56.56	**64.03**	**66.22**	67.32	48.82	**55.03**	**56.79**	57.70

Table 5: Dependency parsing results on the Universal Treebanks (unlabeled and labeled attachment scores). Parameters tuned on development data: $\sigma = 0.005, \delta = 20, i = 3$.

	Klementiev	Chandar	Inverted
En-Es (S+P)	0.20	0.24	**0.25**
Es-En (S+P)	0.35	0.32	**0.41**
En-Es (S)	0.20	**0.25**	**0.25**
Es-En (S)	0.38	0.39	**0.53**

Table 7: Word alignment results (P@1). S=sure (certain) alignments. P=possible alignments.

5 Related Work

As noted in §1, there has been some work on learning word representations for cross-lingual parsing lately. Täckström et al. (2013) presented a bilingual clustering algorithm and used the word clusters to augment a delexicalized transfer baseline. Bansal et al. (2014), in the context of monolingual dependency parsing, investigate continuous word representation for dependency parsing in a monolingual cross-domain setup and compare them to word clusters. However, to make the embeddings work, they had to i) bucket real values and perform hierarchical clustering on them, ending up with word clusters very similar to those of Täckström et al. (2013); ii) use syntactic context to estimate embeddings. In the cross-lingual setting, syntactic context is not available for the target language, but doing clustering on top of inverted indexing is an interesting option we did not explore in this paper.

Xiao and Guo (2014) is, to the best of our knowledge, the only parser using bilingual embeddings for unsupervised cross-lingual parsing. They evaluate their models on CoNLL 2006 and CoNLL 2007, and we compare our results to theirs in §4. They obtain much better relative improvements on dependency parsing that we do - comparable to those we observe in document classification and POS tagging. It is not clear to us what is the explanation for this improvement. The approach relies on a bilingual dictionary as in Klementiev et al. (2012) and Gouws and Søgaard (2015), but none of these embeddings led to improvements. Unfortunately, we did not have the code or embeddings of Xiao and Guo (2014). One possible explanation is that they use the embeddings in a very different way in the parser. They use the MSTParser. Unfortunately, they do not say exactly how they combine the embeddings with their baseline feature model.

The idea of using inverted indexing in Wikipedia for modelling language is not entirely new either. In cross-lingual information retrieval, this technique, sometimes referred to as *explicit semantic analysis*, has been used to measure source and target language document relatedness (Potthast et al., 2008; Sorg and Cimiano, 2008). Gabrilovich and Markovitch (2009) also use this technique to model documents, and they evaluate their method on text categorization and on computing the degree of semantic relatedness between text fragments. See also Müller and Gurevych (2009) for an application of explicit semantic analysis to modelling documents. This line of work is very different from ours, and to the best of our knowledge, we are the first to propose to use inverted indexing of Wikipedia for cross-lingual word representations.

6 Conclusions

We presented a simple, scalable approach to obtaining cross-lingual word representations that enables multi-source learning. We compared these representations to two state-of-the-art approaches to neural net word embeddings across four tasks and 17 datasets, obtaining better results than *both* approaches in 14/17 of these cases.

References

Mohit Bansal, Kevin Gimpel, and Karen Livescu. 2014. Tailoring continuous word representations for dependency parsing. In *ACL*.

Marco Baroni, Georgiana Dinu, and Germán Kruszewski. 2014. Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors. In *ACL*.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Janvin. 2003. A neural probabilistic language model. *The Journal of Machine Learning Research*, 3:1137–1155.

Bernd Bohnet. 2010. Top accuracy and fast dependency parsing is not a contradiction. In *COLING*.

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.

Sarath Chandar, Stanislas Lauly, Hugo Larochelle, Mitesh Khapra, Balaraman Ravindran, Vikas C Raykar, and Amrita Saha. 2014. An autoencoder approach to learning bilingual word representations. In *NIPS*.

Michael Collins. 2002. Discriminative Training Methods for Hidden Markov Models: Theory and Experiments with Perceptron Algorithms. In *EMNLP*.

Ronan Collobert and Jason Weston. 2008. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *ICML*.

Dipanjan Das and Slav Petrov. 2011. Unsupervised part-of-speech tagging with bilingual graph-based projections. In *ACL*.

Victoria Fossum and Steven Abney. 2005. Automatically inducing a part-of-speech tagger by projecting from multiple source languages across aligned corpora. In *IJCNLP*.

Evgeniy Gabrilovich and Shaul Markovitch. 2009. Wikipedia-based semantic interpretation for natural language processing. *Journal of Artificial Intelligence Research*, pages 443–498.

Stephan Gouws and Anders Søgaard. 2015. Simple task-specific bilingual word embeddings. In *NAACL*.

Joao Graca, Joana Pardal, Luísa Coheur, and Diamantino Caseiro. 2008. Building a golden collection of parallel multi-language word alignments. In *LREC*.

Rebecca Hwa, Philip Resnik, Amy Weinberg, Clara Cabezas, and Okan Kolak. 2005. Bootstrapping parsers via syntactic projection across parallel texts. *Natural Language Engineering*, 11(3):311–325.

Alexandre Klementiev, Ivan Titov, and Binod Bhattarai. 2012. Inducing crosslingual distributed representations of words. In *COLING*.

Omer Levy and Yoav Goldberg. 2014. Dependency-based word embeddings. In *ACL*.

Shen Li, João Graça, and Ben Taskar. 2012. Wiki-ly supervised part-of-speech tagging. In *EMNLP*.

Ryan McDonald, Slav Petrov, and Keith Hall. 2011. Multi-source transfer of delexicalized dependency parsers. In *EMNLP*.

Ryan McDonald, Joakim Nivre, Yvonne Quirmbach-Brundage, Yoav Goldberg, Dipanjan Das, Kuzman Ganchev, Keith Hall, Slav Petrov, Hao Zhang, Oscar Täckström, Claudia Bedini, Núria Bertomeu Castelló, and Jungmee Lee. 2013. Universal dependency annotation for multilingual parsing. In *ACL*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In *NIPS*.

Christof Müller and Iryna Gurevych. 2009. A study on the semantic relatedness of query and document terms in information retrieval. In *EMNLP*.

Martin Potthast, Benno Stein, and Maik Anderka. 2008. A wikipedia-based multilingual retrieval model. In *Advances in Information Retrieval*.

Peter Prettenhofer and Benno Stein. 2010. Cross-language text classification using structural correspondence learning. In *ACL*.

Anders Søgaard. 2011. Data point selection for cross-language adaptation of dependency parsers. In *Proceedings of ACL*.

Philipp Sorg and Philipp Cimiano. 2008. Cross-lingual information retrieval with explicit semantic analysis. In *Working Notes for the CLEF 2008 Workshop*.

Oscar Täckström, Dipanjan Das, Slav Petrov, Ryan McDonald, and Joakim Nivre. 2013. Token and type constraints for cross-lingual part-of-speech tagging. *TACL*, 1:1–12.

Joseph Turian, Lev Ratinov, and Yoshua Bengio. 2010. Word representations: a simple and general method for semi-supervised learning. In *ACL*.

Min Xiao and Yuhong Guo. 2014. Distributed word representation learning for cross-lingual dependency parsing. In *CoNLL*.

Daniel Zeman and Philip Resnik. 2008. Cross-language parser adaptation between related languages. In *IJCNLP*.