Trans-gram, Fast Cross-lingual Word-embeddings

$\text{rey}_{\text{es}} - \text{Mann}_{\text{de}} = \text{regina}_{\text{it}} - \text{femme}_{\text{fr}}$

Jocelyn Coulmance
105 rue La Fayette
75010 Paris
joc@proxem.com

Jean-Marc Marty *
105 rue La Fayette
75010 Paris
jmm@proxem.com

Guillaume Wenzek *
105 rue La Fayette
75010 Paris
guw@proxem.com

Amine Benhalloum
105 rue La Fayette
75010 Paris
aba@proxem.com

Abstract

We introduce *Trans-gram*, a simple and computationally-efficient method to simultaneously learn and align word-embeddings for a variety of languages, using only monolingual data and a smaller set of sentence-aligned data. We use our new method to compute aligned word-embeddings for twenty-one languages using English as a pivot language. We show that some linguistic features are aligned across languages for which we do not have aligned data, even though those properties do not exist in the pivot language. We also achieve state of the art results on standard cross-lingual text classification and word translation tasks.

1 Introduction

Word-embeddings are a representation of words with fixed-sized vectors. It is a distributed representation (Hinton, 1984) in the sense that there is not necessarily a one-to-one correspondence between vector dimensions and linguistic properties. The linguistic properties are distributed along the dimensions of the space.

A popular method to compute word-embeddings is the Skip-gram model (Mikolov et al., 2013a). This algorithm learns high-quality word vectors with a computation cost much lower than previous methods. This allows the processing of very important amounts of data. For instance, a 1.6 billion words dataset can be processed in less than one day.

Several authors came up with different methods to align word-embeddings across two languages (Klementiev et al., 2012; Mikolov et al., 2013b; Lauly et al., 2014; Gouws et al., 2015).

In this article, we introduce a new method called Trans-gram, which learns word embeddings aligned across many languages, in a simple and efficient fashion, using only sentence alignments rather than word alignments. We compare our method with previous approaches on a cross-lingual document classification task and on a word translation task and obtain state of the art results on these tasks. Additionally, word-embeddings for twenty-one languages are learned simultaneously - to our knowledge - for the first time, in less than two and a half hours. Furthermore, we illustrate some interesting properties that are captured such as cross-lingual analogies, e.g $\vec{\text{rey}}_{\text{es}} - \vec{\text{Mann}}_{\text{de}} + \vec{\text{femme}}_{\text{fr}} \approx \vec{\text{regina}}_{\text{it}}$ which can be used for disambiguation.

2 Review of Previous Work

A number of methods have been explored to train and align bilingual word-embeddings. These methods pursue two objectives: first, similar representations (i.e. spatially close) must be assigned to similar words (i.e. "semantically close") within each language - this is the **mono-lingual objective**; second, similar representations must be assigned to similar words across languages - this is the **cross-lingual objective**.

The simplest approach consists in separating the mono-lingual optimization task from the cross-lingual optimization task. This is for example the case in (Mikolov et al., 2013b). The idea is to separately train two sets of word-embeddings for each language and then to do a parametric estimation of the mapping between word-embeddings across languages. This method was further extended by (Faruqui and Dyer, 2014). Even though those algorithms proved to be viable and fast, it is not clear whether or not a simple mapping between whole languages exists. Moreover, they require word alignments which are a rare and expensive resource.

*These authors contributed equally.

Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pages 1109–1113,
Lisbon, Portugal, 17-21 September 2015.

Another approach consists in focusing entirely on the cross-lingual objective. This was explored in (Hermann and Blunsom, 2013; Lauly et al., 2014) where every couple of aligned sentences is transformed into two fixed-size vectors. Then, the model minimizes the Euclidean distance between both vectors. This idea allows processing corpus aligned at sentence-level rather than word-level. However, it does not leverage the abundance of existing mono-lingual corpora .

A popular approach is to jointly optimize the mono-lingual and cross-lingual objectives simultaneously. This is mostly done by minimizing the sum of mono-lingual loss functions for each language and the cross-lingual loss function. (Klementiev et al., 2012) proved this approach to be useful by obtaining state-of-the-art results on several tasks. (Gouws et al., 2015) extends their work with a more computationally-efficient implementation.

3 From Skip-Gram to Trans-Gram

3.1 Skip-gram

We briefly introduce the Skip-gram algorithm, as we will need it for further explanations. Skip-gram allows to train word embeddings for a language using mono-lingual data. This method uses a dual representation for words. Each word w has two embeddings: a target vector, $\vec{w}$ ($\in \mathbb{R}^D$), and a context vector, $\overleftarrow{w}$ ($\in \mathbb{R}^D$). The algorithm tries to estimate the probability of a word w to appear in the context of a word c. More precisely we are learning the embeddings $\vec{w}, \overleftarrow{c}$ so that: $\sigma(\vec{w} \cdot \overleftarrow{c}) = P(w|c)$ where σ is the sigmoid function.

A simplified version of the loss function minimized by Skip-gram is the following:

$$J = \sum_{s \in C} \sum_{w \in s} \sum_{c \in s[w-l:w+l]} -\log \sigma(\vec{w} \cdot \overleftarrow{c}) \quad (1)$$

where C is the set of sentences constituting the training corpus, and $s[w-l:w+l]$ is a word window on the sentence s centered around w. For the sake of simplicity this equation does not include the "negative-sampling" term, see (Mikolov et al., 2013a) for more details.

Skip-gram can be seen as a materialization of the distributional hypothesis (Harris, 1968): "Words used in similar contexts have similar meanings". We will now see how to extend this idea to cross-lingual contexts.

3.2 Trans-gram

In this section we introduce Trans-gram, a new method to compute aligned word-embeddings for a variety of languages.

Our method will minimize the summation of mono-lingual losses and cross-lingual losses. Like in BilBOWA (Gouws et al., 2015), we use Skip-gram as a mono-lingual loss. Assuming we are trying to learn aligned word vectors for languages e (e.g. English) and f (e.g. French), we note J_e and J_f the two mono-lingual losses.

In BilBOWA, the cross-lingual loss function is a distance between bag-of-words representations of two aligned sentences. But as (Levy and Goldberg, 2014) showed that the Skip-gram loss function extracts interesting linguistic features, we wanted to use a loss function for the cross-lingual objective that will be closer to Skip-gram than BilBOWA.

Therefore, we introduce a new task, Trans-gram, similar to Skip-gram. Each English sentence s_e in our aligned corpus $A_{e,f}$ is aligned with a French sentence s_f. In Skip-gram, the context picked for a target word w_e in a sentence s_e is the set of words c_e appearing in the window centered around w_e: $s_e[w_e - l : w_e + l]$. In Trans-gram, the context picked for a target word w_e in a sentence s_e will be all the words c_f appearing in s_f. The loss can thus be written as:

$$\Omega_{e,f} = \sum_{(s_e,s_f) \in A_{e,f}} \sum_{w_e \in s_e} \sum_{c_f \in s_f} -\log \sigma(\vec{w_e} \cdot \overleftarrow{c_f}) \quad (2)$$

This loss isn't symmetric with respect to the languages. We, therefore, use two cross-lingual objectives: $\Omega_{e,f}$ aligning e's target vectors and f's context vectors and $\Omega_{f,e}$ aligning f's target vectors and e's context vectors. By comparison BilBOWA only aligns e's target vectors and f's target vectors. The figure 1 illustrates the four objectives.

Notice that we make the assumption that the meaning of a word is uniformly distributed in the whole sentence. This assumption, although a naive one, gave us in practice excellent results. Also our method uses only sentence-aligned corpus and not word-aligned corpus which are rarer.

To add a third language i (e.g. Italian), we just have to add 3 new objectives (J_i, $\Omega_{e,i}$ and $\Omega_{i,e}$) to the global loss. If available we could also add $\Omega_{f,i}$ or $\Omega_{i,f}$ but in our case we only used corpora aligned with English.

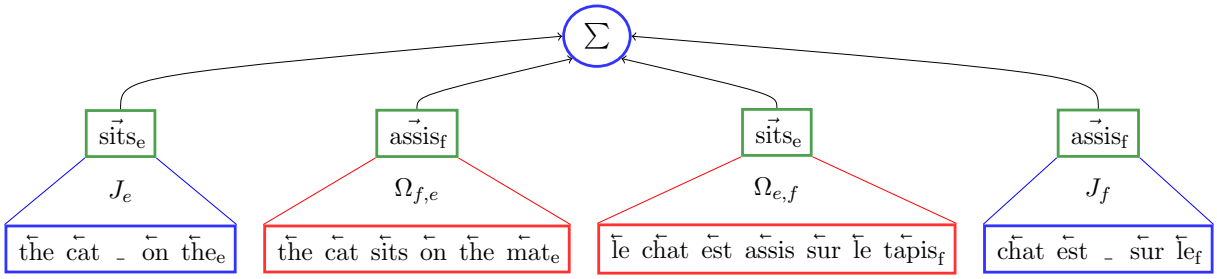

Figure 1: The four partial objectives contributing to the alignment of English and French: a Skip-gram objective per language (J_e and J_f) over a window surrounding a target word (blue) and two Trans-gram objectives ($\Omega_{e,f}$ and $\Omega_{f,e}$) over the whole sentence aligned with the sentence from which the target word is extracted (red).

4 Implementation

In our experiments, we used the Europarl (Koehn, 2005) aligned corpora. Europarl-v7 has two peculiarities: firstly, the corpora are aligned at sentence-level; secondly each pair of languages contains English as one of its members: for instance, there is no French/Italian pair. In other words, English is used as a pivot language. No bi-lingual lexicons nor other bi-lingual datasets aligned at the word level were used.

Using only the Europarl-v7 texts as both mono-lingual and bilingual data, it took 10 minutes to align 2 languages, and two and a half hours to align the 21 languages of the corpus, in a 40 dimensional space on a 6 core computer. We also computed 300 dimensions vectors using the Wikipedia extracts provided by (Al-Rfou et al., 2013) as monolingual data for each language. The training time was 21 hours.

5 Experiments

5.1 Reuters Cross-lingual Document Classification

We used a subset of the English and German sections of the Reuters RCV1/RCV2 corpora (Lewis and Li, 2004) (10000 documents each), as in (Klementiev et al., 2012), and we replicated the experimental setting. In the English dataset, there are four topics: CCAT (Corporate/Industrial), ECAT (Economics), GCAT (Government/Social), and MCAT (Markets). We used these topics as our labels and we only selected documents labeled with a single topic. We trained our classifier on the articles of one language, where each document was represented using an IDF weighted sum of the vectors of its words, we then tested it on the articles of the other language. The classifier used was an averaged perceptron, and we used the implementation from (Klementiev et al., 2012)[1]. The word vectors were computed on the Europarl-v7 parallel corpus with size 40 like other methods. For this task only the target vectors where used.

We report the percentage precision obtained with our method, in comparison with other methods, in Table 1. The table also include results obtained with 300 dimensions vectors trained by Trans-gram with the Europarl-v7 as parallel corpus and the Wikipedia as mono-lingual corpus. The previous state of the art results were detained (Gouws et al., 2015) with BilBOWA and (Lauly et al., 2014) with their Bilingual Auto-encoder model. This model learns word embeddings during a translation task that uses an encoder-decoder approach. We also report the scores from Klementiev et al. who introduced the task and the BiCVM model scores from (Hermann and Blunsom, 2013).

The results show an overall significant improvement over the other methods, with the added advantage of being computationally efficient.

5.2 P@k Word Translation

Next we evaluated our method on a word translation task, introduced in (Mikolov et al., 2013b) and used in (Gouws et al., 2015). The words were extracted from the publicly available WMT11[2] corpus. The experiments were done for two sets of translation: English to Spanish and Spanish to English. (Mikolov et al., 2013b) extracted the top $6K$ most frequent words and translated them with Google Translate. They used the top $5K$ pairs to train a translation matrix, and evaluated their method on the remaining $1K$. As our English and

[1]Thanks to S. Gouws for providing this implementation
[2]http://www.statmt.org/wmt11/

Method	En → De	De → En	Speed-up in training time
Klementiev et al.	77.6%	71.1%	×1
Bilingual Auto-encoder	**91.8%**	72.8%	×3
BiCVM	83.7%	71.4%	×320
BilBOWA	86,5%	75%	×800
Trans-gram	87,8%	**78,7%**	×600
Trans-gram (size 300 vectors EP+WIKI)	91,1%	**78,4%**	

Table 1: Comparison of Trans-gram with various methods for Reuters English/German classification

Method	En → Es P@1	En → Es P@5	Es → En P@1	Es → En P@5
Edit distance	13%	18%	24%	27%
Bing	55%		71%	
Translation Matrix	33%	35%	51%	52%
BilBOWA	39%	44%	**51%**	55%
Trans-gram	**45%**	**61%**	47%	**62%**

Table 2: Results on the translation task

Spanish vectors are already aligned we don't need the $5K$ training pairs and use only the $1K$ test pairs.

The reported score, the translation precision $P@k$, is the fraction of test-pairs where the target translation (Google Translate) is one of the k translations proposed by our model. For a given English word, w, our model takes its target vectors $\vec{w}$ and proposes the k closest Spanish word using the co-similarity of their vectors to $\vec{w}$. We compare ourselves to the "translation matrix" method and to the BilBowa aligned vectors. We also report the scores obtained by a trivial algorithm that uses edit-distance to determine the closest translation and by the Bing Translator service.

6 Interesting properties

6.1 Cross-lingual disambiguation

We now present the task of cross-lingual disambiguation as an example of possible uses of aligned multilingual vectors. The goal of this task is to find a suitable representation of each sense of a given polysemous word. The idea of our method is to look for a language in which the undesired senses are represented by unambiguous words and then to perform some arithmetic operation.

Let's illustrate the process with a concrete example: consider the French word "train", $\mathrm{train}_{\mathrm{fr}}$. The three closest Polish words to $\vec{\mathrm{train}}_{\mathrm{fr}}$ translate in English into "now", "a train" and "when". This seems a poor matching. In fact, $\mathrm{train}_{\mathrm{fr}}$ is polysemous. It can name a line of railroad cars, but it is also used to form progressive tenses. The French "Il est *en train de* manger" translates into "he is eat*ing*", or in Italian "*sta* mangiando".

As the Italian word "sta" is used to form progressive tenses, it's a good candidate to disambiguate $\mathrm{train}_{\mathrm{fr}}$. Let's introduce the vector $\vec{v} = \vec{\mathrm{train}}_{\mathrm{fr}} - \vec{\mathrm{sta}}_{\mathrm{it}}$. Now the three polish words closest to $\vec{v}$ translate in English into "a train", "a train" and "railroad". Therefore $\vec{v}$ is a better representation for the railroad sense of $\mathrm{train}_{\mathrm{fr}}$.

6.2 Transfer of linguistic features

Another interesting property of the vectors generated by Trans-gram is the transfer of linguistic features through a pivot language that does not possess these features.

Let's illustrate this by focusing on Latin languages, which possess some features that English does not, like rich conjugations. For example, in French and Italian the infinitives of "eat" are $\mathrm{manger}_{\mathrm{fr}}$ and $\mathrm{mangiare}_{\mathrm{it}}$, and the first plural persons are $\mathrm{mangeons}_{\mathrm{fr}}$ and $\mathrm{mangiamo}_{\mathrm{it}}$. Actually in our models we observe the following alignments: $\vec{\mathrm{manger}}_{\mathrm{fr}} \approx \vec{\mathrm{mangiare}}_{\mathrm{it}}$ and $\vec{\mathrm{mangeons}}_{\mathrm{fr}} \approx \vec{\mathrm{mangiamo}}_{\mathrm{it}}$. It is thus remarkable to see that features not present in English match in languages aligned through English as the only pivot language. We also found similar transfers for the genders of adjectives and are currently studying other similar properties captured by Trans-gram.

7 Conclusion

In this paper we provided the following contributions: Trans-gram, a new method to compute cross-lingual word-embeddings in a single word space; state of the art results on cross-lingual NLP tasks; a sketch of a cross-lingual calculus to help disambiguate polysemous words; the exhibition of linguistic features transfers through a pivot-language not possessing those features.

We are still exploring promising properties of the generated vectors and their applications in other NLP tasks (Sentiment Analysis, NER...).

References

Rami Al-Rfou, Bryan Perozzi, and Steven Skiena. 2013. Polyglot: Distributed word representations for multilingual nlp. *arXiv preprint arXiv:1307.1662*.

Manaal Faruqui and Chris Dyer. 2014. Improving vector space word representations using multilingual correlation. Association for Computational Linguistics.

Stephan Gouws, Yoshua Bengio, and Greg Corrado. 2015. Bilbowa: Fast bilingual distributed representations without word alignments. In *Proceedings of the 25th international conference on Machine learning*, volume 15, pages 748–756.

Zellig Sabbettai Harris. 1968. Mathematical structures of language.

Karl Moritz Hermann and Phil Blunsom. 2013. Multilingual distributed representations without word alignment. *arXiv preprint arXiv:1312.6173*.

Geoffrey E Hinton. 1984. Distributed representations.

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

Philipp Koehn. 2005. A parallel corpus for statistical machine translation. In *MT Summit*.

Stanislas Lauly, Hugo Larochelle, Mitesh Khapra, Balaraman Ravindran, Vikas C Raykar, and Amrita Saha. 2014. An autoencoder approach to learning bilingual word representations. In *Advances in Neural Information Processing Systems*, pages 1853–1861.

Omer Levy and Yoav Goldberg. 2014. Neural word embedding as implicit matrix factorization. In *Advances in Neural Information Processing Systems*, pages 2177–2185.

Y.; Rose T.; Lewis, D. D.; Yang and F. Li. 2004. Rcv1: A new benchmark collection for text categorization research. In *Journal of Machine Learning Research*.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Tomas Mikolov, Quoc V Le, and Ilya Sutskever. 2013b. Exploiting similarities among languages for machine translation. *arXiv preprint arXiv:1309.4168*.