Distributed Representations of Words and Documents for Discriminating Similar Languages

Marc Franco-Salvador[1], **Paolo Rosso**[1], and **Francisco Rangel**[1,2]
[1] Universitat Politècnica de València, Spain
[2] Autoritas Consulting, S.A., Spain
mfranco@prhlt.upv.es, prosso@dsic.upv.es,
francisco.rangel@autoritas.es

Abstract

Discriminating between similar languages or language varieties aims to detect lexical and semantic variations in order to classify these varieties of languages. In this work we describe the system built by the Pattern Recognition and Human Language Technology (PRHLT) research center - Universitat Politècnica de València and Autoritas Consulting for the Discriminating between similar languages (DSL) 2015 shared task. In order to determine the language group of similar languages, we first employ a simple approach based on distances with language prototypes with 99.8% accuracy in the test sets. For classifying intra-group languages we focus on the use of distributed representations of words and documents using the continuous Skip-gram model. Experimental results of classification of languages in 14 categories yielded accuracies of 92.7% and 90.8% when classifying unmodified texts and text with hidden named entities, respectively.

1 Introduction

Automatic language identification is considered a solved problem in a regular scenario. McNamee (2005) demonstrated how even the most simple of the methods, based on language prototypes of term frequencies, is able to achieve almost 100% accuracy of classification. However, it is far to be solved if we consider the classification of short text, mixed content and when discriminating between language varieties and similar languages. Carter et al. (2013) investigated the language identification of short and noisy text of several European languages using Twitter data, and justified the difficulty of classification in this domain. Gottron and Lipka (2010) studied the identification of European languages in news headlines and single unambiguous words. They demonstrated the impact of the length in the accuracy of classification.

The identification of varieties of the same language has been related to author profiling (Rangel et al., 2013; Rangel et al., 2014; Rangel and Rosso, 2015), which aims to identify the linguistic profile of an author on the basis of his writing style, and to determine author's traits such as gender, age and personality. Variety identification differs from the aforementioned language identification works in terms of difficulty due to the high syntactic and semantic similarities. Accuracy of classification is reduced from 90-100% to values closer to 80%. In (Zampieri and Gebre, 2012) the authors investigated varieties of Portuguese applying different features such as word and character n-grams. Similarly, in (Sadat et al., 2014) the authors differentiate between six different varieties of Arabic in blogs and forums using character n-grams. Concerning Spanish language varieties, in (Maier and Gómez-Rodríguez, 2014) the authors employed meta-learning to classify tweets from Argentina, Chile, Colombia, Mexico and Spain. Zubiaga et al. (2014) overviews the results of the shared task of tweet language identification organized at SEPLN'2014. A more recent work (Franco-Salvador et al., 2015), explored the use of techniques based on embeddings to model semantics and evaluated using the HispaBlogs[1] dataset, a new collection of Spanish blogs from five different countries: Argentina, Chile, Mexico, Peru and Spain. The proposed approach demonstrated to achieve remarkable performance and to be less sensitive to over-fitting than the compared state-of-the-art approaches.

In order to illustrate that language identification is not a solved problem, the Discrimi-

[1]The HispaBlogs dataset can be downloaded at: https://github.com/autoritas/RD-Lab/tree/master/data/HispaBlogs

nating between similar languages (DSL) shared task (Zampieri et al., 2014; Zampieri et al., 2015) is organized. This task encourages participants to submit systems in order to identify the language of short texts of several groups of similar and varieties of languages (Tan et al., 2014). Goutte et al. (2014) achieved the best results of the 2014 edition with a combination of different kernels using Support Vector Machines (SVM) (Chang and Lin, 2011) and word and character n-gram features. This year, the task aims to identify the language of six groups of texts containing similar and varieties of languages (see Table 1) and a group containing texts written in a set of other languages.

In this work we evaluate the 2015 shared task by adapting the approach presented in (Franco-Salvador et al., 2015). We first use an approach based on distances with language prototypes to determine the language group, and next we classify the language using the continuous Skip-gram model to generate distributed representations of words, i.e., n-dimensional vectors –applying further refinements in order to be able to use them in documents. In addition, we use the Sentence Vector variation to directly generate representations of documents. Motivations behind evaluating this approach in the DSL shared task are: i) analyse the performance when classifying not only varieties of languages but also similar ones; and ii) determine the validity of the approach to work with considerably shorter texts (sentences) compared to the blogs with 10 post per user that were used as single instance in the past.

The rest of the paper is structured as follows. Section 2 presents the approach we adapted for the shared task, Section 3 details our evaluation, and in Section 4 we provide our conclusions and future works. Additional analysis and comparison with the other submitted systems are available in the 2015 shared task overview (Zampieri et al., 2015).

2 Discriminating Similar Languages

In this section we detail the approach we used for discriminating between similar languages. We first describe the pre-processing we employed, next we present the method for classifying sentences among language groups of similar languages (inter-group classifier), and finally we review the distributed vector-based approach for identifying the language within groups of similar languages (intra-group classifier).

2.1 Data Pre-processing

For both inter- and intra-group classifiers, we pre-processed the text with tokenization, removed the tokens of length one, and those including numbers or punctuation. In addition, to ease the learning with the considerably low number of text available for generating the distributed vectors and to reduce ambiguity, we lowercased the input words and performed phrase detection for the intra-group classifier.

2.2 Inter-group Classifier

To classify sentences among groups of similar languages, we used a similar and simplified version of McNamee (2005). Having a training set Tr containing sentences belonging to one of the Lg language groups, we first generated the set of prototypes $proto_{Lg}$ of each language group using a bag-of-words representation. Next, for each input sentence $t = (w_1, w_2, ..., w_n)$ of the test set Te, we compute the language group g as follows:

$$g = \text{argmax}_{pr_g \in proto_{Lg}} \sum_{i}^{n} |w_i \cap pr_g|, \qquad (1)$$

where basically we determine the language group of a sentence as the group with the higher number of common words. Note that the sentence is represented as a list and, consequently, we allow for word repetitions, contrary to the prototypes. Using this method with the development partition, we achieved a 99.99% of accuracy in the inter-group classification, and demonstrated again that the task is trivial among considerably different languages.

2.3 Intra-group Classifier

To identify the language of sentences of similar and varieties of languages, we adapted the approach of our previous work (Franco-Salvador et al., 2015). We generated vector representations of sentences in two different ways. In Section 2.3.1 we describe how creating sentence vectors as a combination of distributed word vectors. Next, in Section 2.3.2 we describe an alternative and related approach to directly generate distributed representations of sentences. In Section 2.3.3 we describe the algorithms we chose for classification.

2.3.1 Generating Sentence Vectors from Word Vectors

The use of log-linear models has been proposed (Mikolov et al., 2013a) as an efficient al-

ternative to generate distributed representations, since they reduce the complexity of the hidden layer thereby improving efficiency. In this section we use the continuous Skip-gram model (Mikolov et al., 2013a; Mikolov et al., 2013b) to generate distributed representations (e.g. vectors) of words. It is an iterative algorithm which attempts to maximize the classification of the context surrounding a word. Formally, given a word w_t, and its surrounding words $w_{t-c},\ w_{t-c+1}, ...,\ w_{t+c}$ inside a window of size $2c+1$, the training objective is to maximize the average of the log probability:

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{-c\leq j\leq c, j\neq 0}\log p(w_{t+j}|w_t) \quad (2)$$

Although $p(w_{t+j}|w_t)$ can be estimated using the softmax function (Barto, 1998), its normalization depends on the vocabulary size W which makes its usage impractical for high values of W. For this reason, more computationally efficient alternatives are used instead. In this work we used the negative sampling (Mikolov et al., 2013b), a simplified version of the Noise Contrastive Estimation (NCE) (Gutmann and Hyvärinen, 2012; Mnih and Teh, 2012), which basically uses logistic regression to distinguish the target word from a noise distribution, having k negative samples for each word. Experimental results in Mikolov et al. (2013a) show that the Skip-gram model obtains better results at semantic level than other log-linear alternatives such as the continuous Bag-of-words model, and Mikolov et al. (2013b) offered identical conclusions for the negative sampling compared to NCE and Hierarchical softmax (Morin and Bengio, 2005), hence the election of our models.

In order to combine the word vectors generated with the Skip-gram model, having a list of vectors $(\vec{w}_1, \vec{w}_2, ..., \vec{w}_n)$ belonging to the words of a sentence, we generated a vector representation $\vec{v}$ of its content by estimating the average of their dimensions: $\vec{v} = n^{-1}\sum_{i=1}^{n}\vec{w}_i$. We refer to such document vector representation as *Skip-gram* in the evaluation section.

2.3.2 Learning Sentence Vectors

Sentence vectors (SenVec) (Le and Mikolov, 2014) follows Skip-gram architecture to train a special vector $\vec{v}$ representing the complete sentence. Basically, the model uses all the words of the sentence as context to train the vector representing its content. In contrast, the original Skip-gram model employs a fixed size window to determine the context (surrounding words) of the iterated words of a sentence.

2.3.3 Classifying distributed vectors

To classify the distributed vectors of the combination of words, we used a logistic classifier (Skip-gram + LG (run1)). For that model we employed also an SVM classifier (Skip-gram + SVM (run2)) with radial basis function kernel and cost 10. Finally, SenVec vectors were classified using a logistic classifier (SenVec + LG (run3)). At this point, we must point out that the test sentences contain words which are not present in the training set. Obviously, for those words we have not learned a distributed vector but we have the initial random vector we could use to train it. Despite we could directly ignore and remove those words, the experiments with the development partition showed that there is not loss of performance when we include those vectors, and even in some configurations, e.g. Skip-gram + LG, provided a very slight improvement (0.6%). We hypothesize that this insertion of noise in the vectors may help the classifiers to determine the frontiers among languages, and we kept them in our experiments.

3 Evaluation

In this section we evaluate our systems for the DSL 2015 shared task. Given a labelled collection of training sentences Tr belonging to a set of L languages, and a collection of test sentences Te, the task is to classify each sentence $t \in Te$ into one of the languages $l \in L$ using the labelled sentences of Tr.

3.1 Dataset and Methodology

We evaluated our system with the DSL Corpus Collection (Tan et al., 2014) of this edition (DSLCC v. 2.0). This dataset contains sentences in Bulgarian, Macedonian, Serbian, Croatian, Bosnian, Czech, Slovak, Argentinian Spanish, Peninsular Spanish, Brazilian Portuguese, European Portuguese, Malay, Indonesian and a group containing texts written in a set of other languages. In Table 1 we can see how they are grouped according to their similarities. Groups A, C and F contain similar languages and groups B, D and E include language varieties. There are 18,000 training, 2,000 development and 1,000 test instances/sentences per language. In addition, the

Language groups	Unmodified texts (**Test set A**)			Named entities substituted with #NE# (**Test set B**)		
	Skip-gram + LG (run1)	**Skip-gram + SVM (run2)**	**SenVec + LG (run3)**	**Skip-gram + LG (run1)**	**Skip-gram + SVM (run2)**	**SenVec + LG (run3)**
Bulgarian	**1.000**	**1.000**	0.985	**1.000**	**1.000**	0.998
Macedonian	**1.000**	**1.000**	0.999	**1.000**	**1.000**	0.998
Overall (group A)	**1.000**	**1.000**	0.992	**1.000**	**1.000**	0.998
Bosnian	**0.803**	0.795	0.744	**0.751**	0.750	0.641
Croatian	**0.859**	0.837	0.847	**0.858**	0.853	0.769
Serbian	0.751	0.802	**0.912**	0.747	0.772	**0.871**
Overall (group B)	0.804	0.811	**0.834**	0.785	**0.791**	0.760
Czech	**0.999**	**0.999**	0.998	**1.000**	**1.000**	**1.000**
Slovak	**1.000**	**1.000**	0.993	**1.000**	**1.000**	0.951
Overall (group C)	**0.999**	**0.999**	0.995	**1.000**	**1.000**	0.976
Spanish (Spain)	0.821	**0.878**	0.863	0.806	**0.853**	0.796
Spanish (Argentina)	**0.903**	0.870	0.876	**0.847**	0.770	0.816
Overall (group D)	0.862	**0.874**	0.869	**0.826**	0.806	0.806
Portuguese (Brazil)	**0.945**	0.926	0.876	**0.904**	0.900	0.783
Portuguese (Portugal)	0.832	0.879	**0.900**	0.780	0.832	**0.866**
Overall (group E)	0.888	**0.902**	0.888	0.842	**0.866**	0.824
Malay	0.992	0.994	**0.998**	0.987	**0.990**	0.917
Indonesian	0.993	**0.996**	0.994	0.989	0.994	**0.996**
Overall (group F)	0.992	0.995	**0.996**	0.988	**0.992**	0.956
Other languages	**0.998**	**0.998**	**0.998**	**0.998**	**0.998**	**0.998**
Overall (all groups)	0.921	**0.927**	**0.927**	0.905	**0.908**	0.885

Table 1: Accuracy results in discrimination between similar languages using test set A and B.

dataset is provided in two variants. The test set A includes unmodified journalistic texts. Test set B used different instances and substituted named entities for the #NE# tag to study the bias they provide. Results measure the accuracy of language identification of the Skip-gram + LG, Skip-gram + SVM and SenVec + LG classifiers[2] in both datasets.

3.2 Results

As we can see in Table 1, similar languages were easier to distinguish, with accuracies close to 100%. A similar trend is appreciated to identify the "other languages" group, which contains instances of several alternative languages such as French or Catalan. The language varieties were more difficult, obtaining values in the range 80–90%, the most difficult being the group of the Serbo-Croatian language, followed by the Spanish and Portuguese. Regarding the substitution of named entities with the #NE# tag, we appreciated a small reduction in accuracy, more elevated for the SenVec model. In general, the differences between the models and classifiers were reduced. In Table 2 we can see the evaluation of statistical significance among the different models. SVM provided slight improvements (increasing several times the training time) with respect to the logistic classifier, and inferred more accurate frontiers among languages (see language variety group inner values). The Skip-gram approach was less sensitive to the substitution of named entities and offered the best performance in average. That model is a few points below compared to the best participant in the task which achieved 95.54% and 94.01% in the test set A and B respectively.

	R<#run> <(test set) {A,B}>					
	R1A	**R2A**	**R3A**	**R1B**	**R2B**	**R3B**
R1A		=	=	*	*	*
R2A			=	*	*	*
R3A				*	*	*
R1B					=	*
R2B						*
R3B						

Table 2: Pairwise statistical test of significance among submitted runs (= not significant $p > 0.05$; * significant $0.05 \geq p > 0.01$).

Comparing the results with those obtained for language variety identification in Franco-Salvador et al. (2015), closer to 70%, with respect the previous experiments carried out on the HispaBlogs dataset we would like to highlight that: i) there is a further difficulty when processing noisy social media texts than more formal journalistic ones; ii) the length of the texts in HispaBlogs is of 10 posts for user blog (that could introduce ambiguity and

[2]We used 300-dimensional vectors, context windows of size 10, and 20 negative words for each sample. We used the word2vec toolkit to perform the phrase detection and the vector training:
`https://code.google.com/p/word2vec/`

noise) whereas in the DSLCC dataset is of a single sentence per instance; iii) the number of classes in HispaBlogs is five whereas in DSLCC is three per group in the worst case; and iv) we think that the overfitting may have a significant impact on the results: whereas in HispaBlogs a different author is given in each instance, in DSLCC there is no such restriction. Therefore, models may profile the author's writing style to classify the test instances of the same authors they already saw in the training set.

4 Conclusions

In this work we evaluated the Discriminating between similar languages 2015 shared task. We employed the continuous Skip-gram model to generate distributed representations of words and sentences with interesting insights about the identification of languages. As expected, groups of language varieties were more difficult to classify. In addition, the substitution of named entities with the #NE# tag slightly reduced the accuracy. Finally, the combination of word vectors (Skip-gram) offered better results on average than the use of directly generated vectors of sentences (SenVec). As future work we will investigate further how to apply distributed representations to other author profiling tasks. We will continue working also to improve the current model in order to generate better distributed representations for discriminating between similar languages.

Acknowledgments

This research has been carried out within the framework of the European Commission WIQ-EI IRSES (no. 269180) and DIANA - Finding Hidden Knowledge in Texts (TIN2012-38603-C02) projects, and the VLC/CAMPUS Microcluster on Multimodal Interaction in Intelligent Systems. The work of the third author was partially funded by Autoritas Consulting SA and by Spanish Ministry of Economics under grant ECOPORTUNITY IPT-2012-1220-430000.

References

Andrew G Barto. 1998. *Reinforcement learning: An introduction*. MIT press.

Simon Carter, Wouter Weerkamp, and Manos Tsagkias. 2013. Microblog language identification: Overcoming the limitations of short, unedited and idiomatic text. *Language Resources and Evaluation*, 47(1):195–215.

Chih-Chung Chang and Chih-Jen Lin. 2011. Libsvm: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27.

Marc Franco-Salvador, Francisco Rangel, Paolo Rosso, Mariona Taulé, and M. Antònia Martí. 2015. Language variety identification using distributed representations of words and documents. In *Proceeding of the 6th International Conference of CLEF on Experimental IR meets Multilinguality, Multimodality, and Interaction (CLEF 2015)*, volume LNCS(9283). Springer-Verlag.

Thomas Gottron and Nedim Lipka. 2010. A comparison of language identification approaches on short, query-style texts. In *Advances in information retrieval*, pages 611–614. Springer.

Cyril Goutte, Serge Léger, and Marine Carpuat. 2014. The nrc system for discriminating similar languages. In *Proceedings of the First Workshop on Applying NLP Tools to Similar Languages, Varieties and Dialects*, pages 139–145, Dublin, Ireland, August. Association for Computational Linguistics.

Michael U. Gutmann and Aapo Hyvärinen. 2012. Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics. *The Journal of Machine Learning Research*, 13(1):307–361.

Quoc V. Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *Proceedings of the 31st International Conference on Machine Learning*.

Wolfgang Maier and Carlos Gómez-Rodríguez. 2014. Language variety identification in spanish tweets. In *Proceedings of the EMNLP'2014 Workshop on Language Technology for Closely Related Languages and Language Variants*, pages 25–35, Doha, Qatar, October. Association for Computational Linguistics.

Paul McNamee. 2005. Language identification: a solved problem suitable for undergraduate instruction. *Journal of Computing Sciences in Colleges*, 20(3):94–101.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. In *Proceedings of Workshop at International Conference on Learning Representations*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26*, pages 3111–3119.

Andriy Mnih and Yee Whye Teh. 2012. A fast and simple algorithm for training neural probabilistic language models. *arXiv preprint arXiv:1206.6426*.

Frederic Morin and Yoshua Bengio. 2005. Hierarchical probabilistic neural network language model. In *Proceedings of the international workshop on artificial intelligence and statistics*, pages 246–252. Citeseer.

Francisco Rangel and Paolo Rosso. 2015. On the impact of emotions on author profiling. *Information Processing & Management*, pages n/a, DOI: 10.1016/j.ipm.2015.06.003 (In press).

Francisco Rangel, Paolo Rosso, Moshe Moshe Koppel, Efstathios Stamatatos, and Giacomo Inches. 2013. Overview of the author profiling task at pan 2013. In *Forner P., Navigli R., Tufis D. (Eds.), CLEF 2013 Labs and Workshops, Notebook Papers*, volume 1179, pages 352–365. CEUR-WS.org.

Francisco Rangel, Paolo Rosso, Irina Chugur, Martin Potthast, Martin Trenkmann, Benno Stein, Ben Verhoeven, and Walter Daelemans. 2014. Overview of the 2nd author profiling task at pan 2014. In *Cappellato L., Ferro N., Halvey M., Kraaij W. (Eds.) CLEF 2014 Labs and Workshops, Notebook Papers.*, volume 1180, pages 898–827. CEUR-WS.org.

Fatiha Sadat, Farnazeh Kazemi, and Atefeh Farzindar. 2014. Automatic identification of arabic language varieties and dialects in social media. In *Proceeding of the 1st. International Workshop on Social Media Retrieval and Analysis SoMeRa*.

Liling Tan, Marcos Zampieri, Nikola Ljubešic, and Jörg Tiedemann. 2014. Merging comparable data sources for the discrimination of similar languages: The dsl corpus collection. In *Proceedings of the 7th Workshop on Building and Using Comparable Corpora*, pages 11–15, Reykjavik, Iceland.

Marcos Zampieri and Binyam Gebrekidan Gebre. 2012. Automatic identification of language varieties: The case of portuguese. In *KONVENS2012-The 11th Conference on Natural Language Processing*, pages 233–237. Österreichischen Gesellschaft für Artificial Intelligende (ÖGAI).

Marcos Zampieri, Liling Tan, Nikola Ljubešić, and Jörg Tiedemann. 2014. A report on the dsl shared task 2014. In *Proceedings of the First Workshop on Applying NLP Tools to Similar Languages, Varieties and Dialects*, pages 58–67, Dublin, Ireland, August. Association for Computational Linguistics.

Marcos Zampieri, Liling Tan, Nikola Ljubešić, Jörg Tiedemann, and Preslav Nakov. 2015. Overview of the dsl shared task 2015. In *Proceedings of the Joint Workshop on Language Technology for Closely Related Languages, Varieties and Dialects (LT4VarDial)*, Hissar, Bulgaria.

Arkaitz Zubiaga, Inaki San Vicente, Pablo Gamallo, José Ramom Pichel, Inaki Alegria, Nora Aranberri, Aitzol Ezeiza, and Vıctor Fresno. 2014. Overview of tweetlid: Tweet language identification at sepln 2014. *TweetLID@ SEPLN*.