Cross-Lingual Propagation of Sentiment Information Based on Bilingual Vector Space Alignment

Cross-Lingual Propagation of Sentiment Information Based on Bilingual Vector Space Alignment

Flavio Giobergia
Politecnico di Torino
Turin, Italy
flavio.giobergia@polito.it

Luca Cagliero
Politecnico di Torino
Turin, Italy
luca.cagliero@polito.it

Paolo Garza
Politecnico di Torino
Turin, Italy
paolo.garza@polito.it

Elena Baralis
Politecnico di Torino
Turin, Italy
elena.baralis@polito.it

ABSTRACT

Deep learning methods have shown to be particularly effective in inferring the sentiment polarity of a text snippet. However, in cross-domain and cross-lingual scenarios there is often a lack of training data. To tackle this issue, propagation algorithms can be used to yield sentiment information for various languages and domains by transferring knowledge from a source language (usually English). To propagate polarity scores to the target language, these algorithms take as input an initial vocabulary and a bilingual lexicon. In this paper we propose to enrich lexicon information for cross-lingual propagation by inferring the bilingual semantic relationships from an aligned bilingual vector space. This allows us to exploit the underlying text similarities that are not made explicit by the lexicon. The experiments show that our approach outperforms the state-of-the-art propagation method on multilingual datasets.

1 INTRODUCTION

In the last decade, an increasing amount of opinionated data has been recorded in digital forms (e.g., reviews, tweets, blogs). This has fostered the joint use of Natural Language Processing (NLP) and Machine Learning (ML) techniques to extract people's opinions, sentiments, emotions, and attitude from text, i.e., the sentiment analysis (or opinion mining) problems [16].

The recently proposed approaches (e.g., [1, 8, 9]) aim to predict the sentiment polarity of the analyzed text by means of deep learning techniques. However, Deep Neural Networks require a sufficiently large corpus of labeled data in order to train accurate sentiment predictors [11]. Meeting such a requirement could be challenging while coping with multilingual and cross-domain data. In particular, the majority of the annotated text is written in English whereas small amounts of data are available for less commonly spoken languages. Furthermore, the sentiment of a text snippet strongly depends on its surrounding context. For example, a word may have different connotations in different domains. Hence, tailoring DNN models to the right domain and language is crucial for developing accurate and portable sentiment analyzers.

A promising strategy to overcome the lack of multilingual training data has recently been proposed by [9]. They propose an approach to propagate sentiment information, encoded into high-dimensional embedding vectors [17], across languages. The idea behind this is to consider an initial vocabulary for which sentiment embeddings are known (usually in English) and a lexicon that maps English words to those in the target language. The mapping indicates the semantic relationship between pairs of words. First, the word-level sentiment polarity in various domains is extracted in the source language using a supervised transfer learning process. Then, the vector scores for the target language are induced using stochastic gradient descent. A more detailed description of [9] is given in Section 3.

Challenge. The quality of the sentiment score propagation strongly depends on the richness of the bilingual lexicon. When a bilingual lexicon is either not available or partly incomplete, the induction phase is unable to effectively propagate the sentiment polarity scores from the original language to the target one.

Research goals. The goal of this work is to improve the quality of the sentiment propagation phase across languages. The key idea is to enrich lexicon information in the propagation phase by deriving the semantic links among word pairs from an aligned bilingual vector space. This allows us to exploit the underlying text similarities that are not made explicit in the bilingual lexicon. We use an established model for vector representation of words, i.e., fastText [4]. A deep learning approach to generate aligned fastText word vectors has recently been proposed [13]. Once trained, the bilingual vector spaces not only embed lexicon information but also allow us to derive non-trivial semantic text relationships directly from the latent space. This simplifies the procedure of cross-lingual induction and exploits the vector representation of text in the latent space to infer missing word relationships. The authors of [13] have also published the pre-trained aligned vectors for a large number of languages. Hence, a promptly usable, general-purpose vector representation of text is currently available.

Approach. To propagate the multi-domain sentiment polarity scores of a word in the original language (e.g., English), we explore the bilingual aligned vector space. Specifically, an arbitrary word in the original vocabulary is described by two high-dimensional vectors: a latent vector in the original embedding space and a sentiment vector describing the sentiment polarity of the word in different domains. Thanks to the bilingual model, we project words from the original word embedding to the vector space of the target language and look for its nearest neighbors. Neighbors are likely to be semantically related to the original word (regardless of the presence of an explicit link in the bilingual lexicon). The semantic links and the similarity scores between the projected word and the neighbors are used to drive the sentiment propagation phase towards the target language.

Achievements. The proposed approach produces sentiment embeddings that outperform the state-of-the-art embeddings by [9] on various multilingual benchmark datasets (e.g., using the embeddings with an SVM classifier, it achieves 10% average macro F_1 score improvement on the Italian datasets).

Paper outline. Section 2 overviews the related literature. Section 3 summarizes the cross-lingual propagation method presented by [9] and introduces the mathematical notation used throughout the paper. Section 4 presents the proposed approach. Section 5 reports the outcomes of the empirical evaluation. Finally, Section 6 draws conclusions and presents the future works.

2 RELATED WORK

To predict the sentiment polarity of textual reviews, news, and posts, several deep learning-based sentiment analysis approaches have been proposed. Most of them (e.g., [2, 12, 18]) are language- or domain-specific, i.e., they are specifically tailored to a given context (e.g., movie reviews, Twitter posts) and language. Hence, model learning assumes that a large enough training set is available. Unfortunately, in many real contexts and for various languages this is not the case.

To extend the applicability of existing sentiment analysis solutions towards other languages, the use of automated machine translation tool has been investigated [3, 7, 10, 20]. The main drawbacks of using automatic text translation tools are that the process is computationally intensive, the generated translations are prone to errors, and the tools often miss the word semantic differences according to the context of use [10]. Parallel strategies entail (i) building sentiment lexicons tailored to different languages and domains of interest and exploiting them to train supervised models [5] and (ii) integrate syntax-based rules in unsupervised models [19]. However, all the aforementioned approaches require a significant human effort, which has already been accomplished only for the major languages and the most popular domains.

To propagate sentiment information across different languages and domains, a deep learning approach has recently been presented [9] by Dong and De Melo. They consider an initial vocabulary of English words for which sentiment embeddings are known and a translation lexicon representing semantic relationships between pairs of words (both non-English and English words) such as translation, synonym, orthographic variants, and other semantic, morphological, and etymological word relationships is given. In [9], links between words are extracted from a multilingual Wiktionary dump [6]. However, for the languages and domains not yet supported by the Etymological Wordnet Project, still the method is unable to propagate sentiment information. Furthermore, some relevant semantic links could be missing in the input lexicon. Our proposal is to rely on an aligned bilingual vector space, e.g., [13], from which explicit and implicit semantic relationships among words can be inferred.

3 SENTIMENT PROPAGATION BASED ON TRANSLATION LEXICON

To propagate sentiment information to various languages, the approach proposed by [9] generates a sentiment embedding vector v_x for each word x in a multilingual vocabulary V. A sentiment embedding is a high-dimensional vector reflecting the distribution of the word's sentiment polarities across a large range of domains[1]. If the same word is used in multiple languages, each instance is treated as a distinct word in V.

The sentiment vector associated with each word x in the initial vocabulary V_0 is derived via transfer learning. Specifically, for each domain d_j a linear Support Vector Machine classifier [15] $M_j(x) = w_j \cdot x + b$ is trained from a set of domain-specific textual documents. The classifier assigns a polarity to each word, denoting whether the word is peculiar to the domain under analysis. The j-th component of the embedding vector v_x incorporates the coefficient w_j of the linear model M_j. Hereafter, we will assume sentiment information to be known for an initial vocabulary $V_0 \subset V$, which usually consists of a subset of English words (i.e., the most popular language used in electronic documents).

The translation lexicon T_L is a set of triples $\{(x_1, x'_1, we_1), \ldots, (x_m, x'_m, we_m)\}$ $[x_1, x'_1, \ldots, x_m, x'_m \in V]$ providing evidence of the semantic relationships holding between pairs of words in the multilingual vocabulary. The lexicon maps words in the original language to the corresponding translations. Notice that each word may have multiple translations. To incorporate relationships such as synonyms, orthographic variants, and etymological connections, the lexicon includes also links between pairs of words of the same language. In [9] the translation lexicon is extracted from a multilingual Wiktionary dump [6]. The links between pairs of words represent translations, synonyms, morphology, derivation, and etymological links. The weight associated with a triple (x_q, x'_q, we_q) $[1 \leq q \leq m]$ denotes the relevance of the semantic relationship. In [6] it indicates the number of semantic links occurring in the data source.

The sentiment vectors of all the words in V are populated by propagating the cross-domain polarity scores of the words in V_0 via an iterative optimization approach, i.e., Stochastic Gradient Descent (SGD). The optimization problem addressed by the SGD entails assigning values to the sentiment vector v_x for all words x in the multilingual vocabulary V according to the following objective function:

$$C \cdot \sum_{x \in V_0} ||v_x - \tilde{v}_x||_2 + \\ - \sum_{x \in V} v_x^T \left[\frac{1}{\sum_{(x,x',we) \in T_L} we} \cdot \sum_{(x,x',we) \in T_L} we \; v_{x'} \right]$$

where, given a word $x \in V_0$, $\tilde{v}_x$ represents its initial sentiment vector (learned through transfer learning).

The first term of the loss function ensures that the sentiment vectors of the words in the initial vocabulary V_0 do not diverge significantly from the original ones, for a large enough constant C. The second term guarantees that the inferred sentiment vectors of words that are linked together (to some extent) in the translation lexicon are kept similar. A drawback of the aforesaid loss function is that the dot product in the second term allows for arbitrarily large magnitudes for the inferred sentiment vectors. Indeed, the dot product can grow by indefinitely increasing the magnitude of the vectors that are being learned. This issue will be addressed by the proposed approach.

4 PROPOSED APPROACH

Instead of propagating sentiment information directly by means of the translation lexicon, we aim to link the semantically related words indirectly according to their similarity in a bilingual word

[1]The vectors used in the experiments have 26 dimensions, one for each domain plus an extra dimension combining all domains together.

embedding space. The idea behind this is to improve the quality of the cross-lingual propagation phase by projecting polarity scores extracted from a richer text representation based on latent spaces.

4.1 Bilingual embedding space

Each word in a dictionary is mapped to a vector in the latent space. The application of word embeddings to address many natural language processing tasks is established. A pioneering work in this field is the Word2Vec model [17]. fastText is a famous extension of Word2Vec, which has been presented in [4]. It provides a more effective vector representation by incorporating sub-words in the input dictionary. The vectors associated with the sub-words can be conveniently combined in order to generate the embeddings of new words that are not present in the dictionary.

Vector representations of text are generated, separately for each language, using deep learning architectures. However, pre-trained vector models (learned from the Wikipedia corpus) are also available for a large number of languages[2]. To links words in different languages, the per-language models need to be aligned first. The procedure to align bilingual fastText vector spaces is thoroughly described in [13]. Notably, a large number of pre-trained aligned models is available[3]. This allows users to exploit the general-purpose, multilingual vectors (characterized by 300 dimensions and trained from Wikipedia for 44 languages) without the need for retraining them from scratch.

4.2 Sentiment propagation strategy

Let E_o be the fastText embedding space in the original language (e.g., English) and let E_T be the aligned embedding space in the target language (i.e., a language other than English). Each word x in the original language has a corresponding vector v_x^{eo} in E_o. Thanks to the aligned bilingual vector space, we can project v_x^{eo} to the target vector space in order to get the corresponding target vector v_x^{et} in E_T. Notice that the new vector does not necessarily correspond to any real word in the target language.

We exploit word similarities in the latent space to propagate sentiment information. Specifically, let v_x^s be the sentiment vector of an arbitrary word $x \in V_0$. We aim to propagate sentiment information to other words in $V \backslash V_0$. This issue is addressed in two steps: (i) first, we create a word graph representing the most significant pairwise word similarities. (ii) Next, we propagate the sentiment scores to the other words using gradient descent. Unlike [9], we adopt a new loss function tailored to the problem under analysis. Notice that step (i) allows us to get a richer word representation compared to a bilingual lexicon.

Word Graph Creation. The word graph $\mathcal{G} = (V, E)$ is a undirected weighted graph connecting pairs of words in V. Edges in E are triples $(x, x', w_{xx'})$, where $x, x' \in V$ are the connected vertices and $w_{xx'}$ is the edge weight. For each word $x \in V$ we explore the neighborhood of vector v_x^{et} in the target latent space to look for the neighbor words that are most semantically related to x. More specifically, we look for the K nearest vectors (where K is a user-specified parameter) corresponding to the words of the target languages that are closest to v_x^{et} and select these words.

Given the set NN_x of x's nearest neighbors, we create a weighted edge $e \in E$ connecting every $x' \in NN_x$ to x. The weight of the edge connecting words x and x' indicates the pairwise word similarity in the latent space and is computed using the cosine similarity [15]. To avoid introducing unreliable word relationships and to limit graph connectedness, we filter out the edges (links) with weight below a given (user-specified) threshold α. The effect of parameters K and α on the performance and complexity of the proposed approach will be discussed in Section 5.

x	x's nearest neighbors	Cosine similarity
excellent	eccellente	0.575
	ottimo	0.513
	apprezzabile	0.369
	buon	0.367
	adatto	0.322

Table 1: Example: K nearest neighbors of *excellent*. Original language = English, target language = Italian, $K = 5$

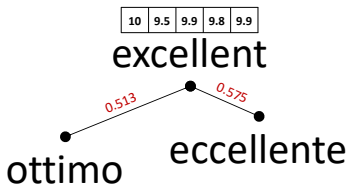

Figure 1: Example: word sub-graph associated with *excellent*. Original language = English, target language = Italian, $K = 5$, $\alpha = 0.4$

Example. Suppose that the original language is English and the target language is Italian. Let us consider the following input parameters: $K = 5$ and $\alpha = 0.4$. Table 1 and Figure 1 report an example related to the English word *excellent*. Specifically, Table 1 reports the five nearest neighbors of *excellent* while Figure 1 shows the word sub-graph associated with that word. Only the first two neighbors of *excellent* are characterized by a cosine similarity greater than or equal to 0.4. Hence, only the Italian words *eccellente* and *ottimo* are connected to *excellent* in the word graph $\mathcal{G}$. This is semantically correct because *eccellente* is the Italian translation of *excellent* and *ottimo* has a similar meaning. The three discarded neighbors are other "positive" adjectives but they do not have the same meaning of excellent (the translations of the other three neighbors are appreciable, good, and suitable, respectively). Hence, the enforcement of the minimum similarity threshold helps us to remove noisy connections.

The English word *excellent*, which is one of the words in V_0, is characterized by a sentiment embedding (i.e., a vector of cross-domain polarity scores). The sentiment vectors of the Italian words are populated by propagating the cross-domain polarity scores of the English words via the iterative optimization approach described in the following.

Gradient Descent with Updated Loss Function. The Gradient Descent is used to propagate sentiment information through the word graph. As discussed in Section 3, the iterative propagation process should both preserve the values of the vectors in the initial vocabulary V_0 and guarantee a high degree of similarity

[2]https://fasttext.cc/
[3]https://fasttext.cc/docs/en/aligned-vectors.html

between the sentiment vectors of linked words. To achieve these goals, we adopt the following objective function:

$$C \cdot \sum_{x \in V_0} ||v_x - \tilde{v}_x||_2 + \\ - \sum_{x \in V} \left\| v_x - \left[\frac{1}{\sum_{(x,x',w_{xx'}) \in E} w_{xx'}} \cdot \sum_{(x,x',w_{xx'}) \in E} w_{xx'} \, v_{x'} \right] \right\|_2$$

where, for each word x in the initial vocabulary V_0, the first term minimizes the deviation from its initial sentiment embedding vector $\tilde{v}_x$. The second term minimizes the deviation from the sentiment vectors of neighbors, represented as connected words in the word graph. Adopting the L2-norm in the second terms allows the propagation of the vector dimensions without altering the vector magnitude. Therefore, words in the initial vocabulary keep, to a good approximation, the same original vectors, whereas new words get sentiment polarity scores similar to those of their neighbors in the target latent space.

5 EXPERIMENTS

The experiments presented in this section are aimed at evaluating the quality of the sentiment vectors resulting from the application of the proposed methodology. The evaluation process is formulated as a binary sentiment analysis problem. The sentiment embeddings are compared, in terms of macro-F_1 score, with those produced by the method presented by [9].

All the experiments were run on a machine equipped with Intel® Xeon® X5650, 32 GB of RAM and running Ubuntu 18.04.1 LTS.

The rest of this section is organized as follows. Subsection 5.1 describes the settings used in the experimental validation as well as the analyzed datasets. Subsection 5.2 summarizes the main results. Subsection 5.3 discusses the influences of the main parameters of the performance of the proposed approach. Finally, Subsection 5.4 analyzes the spatial complexity of the proposed approach.

5.1 Experimental setting

To validate the quality of the generated sentiment vectors, we set up a binary sentiment classification task over multilingual datasets. Specifically, given a set of short text snippets labelled as *positive* or *negative* according to their sentiment polarity, we aim at predicting the sentiment polarities of a subset of related snippets for which the polarities are assumed to be unknown.

To accomplish the classification task, we train two popular classification models, i.e., the Support Vector Machines (SVM) and the Random Forest (RF) classifiers [15]. Classifiers are first trained separately on each multilingual training dataset and then applied to a the corresponding test set. More specifically, each dataset is split into a training set (80% of the data) used for training the models and for tuning of the hyper-parameters, and a test set (20%), which is used for performance evaluation. Classifier settings are set up according to the outcomes of a grid search based on a 5-fold Cross-Validation. Separately for each language and dataset, we evaluate the performance of each classification model in terms of macro-F_1 score. The F_1 score is a popular metric that indicates the harmonic mean of precision and recall of the generated model [15]. Unlike the traditional F_1 score, in the macro-F_1 score the precision values of each class are multiplied with the recall values of all other classes. Hence, the metric is more deemed as more suitable for evaluating imbalanced datasets, i.e., datasets for which the class labels are unevenly distributed in the training data.

Dataset	Cardinality	#Positive	#Negative
cs	2,458	1,660	798
de	2,407	1,839	568
es	2,951	2,367	584
fr	3,912	2,080	1,832
it	3,559	2,867	692
nl	1,892	1,232	660
ru	3,414	2,500	914

Table 2: Cardinality and class distribution for each of the datasets presented in [9]

Dataset	Cardinality	#Positive	#Negative
IT_1	10,024	5,012	5,012
IT_2	13,888	6,942	6,946

Table 3: Key statistics for the new Italian datasets

Classifiers are trained on a vector representation of the input text snippets. The vectors associated with each snippet are computed by averaging the values of the vector dimensions of the words included in the snippet. Notice that the aforesaid task could be alternatively addressed using Recurrent Neural Networks and Convolutional Neural Networks [14]. The comparison between different Deep Learning techniques is out of the scope of the present work.

Classifier performance achieved on the sentiment vectors produced by our method are compared with that of the vectors produced by [9]. The comparison is aimed at showing the higher effectiveness of the proposed approach compared to state-of-the-art solutions.

Data. The list of datasets used for the experiments comprises all the datasets adopted by [9] plus a couple of larger external datasets with different data distributions. Specifically, Table 2 enumerates the characteristics of the existing datasets.

The datasets provided by [9] have been extracted from several websites and collect the reviews left by users on a specific topic (e.g. places, movies, food). The target binary class (*positive* or *negative*) is derived from the user rating. However, users ratings are not necessarily binary values (i.e., they usually comply with the 5-star system). To generate the binary sentiment polarities we have discretized the 5-star ratings as follows: reviews with 3 stars are discarded (as they are considered neutral). 1's and 2's are assigned to the negative class, 4's and 5's to the positive one.

The statistics reported in Table 2 clearly show a strong class imbalance in the analyzed data. This may hinder the training of robust classifiers, as the minority class may be not sufficiently represented by the trained models. To evaluate the performance of the proposed approach on more balanced data as well, we have considered also two additional datasets for the Italian language (i.e., the language for which the imbalance ratio of the corresponding datasets is maximal). Table 3 describes the two new Italian datasets. Data was extracted from reviews of TripAdvisor[4] users in different Italian cities.

[4]https://www.tripadvisor.com

Dataset	Our method		Dong and De Melo	
	SVM	RF	SVM	RF
cs	**0.7403**	0.7198	0.7227	0.7297
de	0.6847	**0.6981**	0.6495	0.6756
es	**0.6131**	0.531	0.4451	0.4892
fr	0.7021	**0.7291**	0.6389	0.6764
it	**0.8256**	0.794	0.6805	0.6644
nl	**0.6869**	0.6369	0.5903	0.6022
ru	0.6840	0.6112	**0.7221**	0.7009
IT_1	**0.8439**	0.8424	0.7435	0.7311
IT_2	**0.8441**	0.8427	0.7415	0.7494

Table 4: Comparison, in terms of macro-F_1 score, between the embeddings produced by the proposed methodology (Our method) and those generated by [9] (Dong and De Melo)

5.2 Performance comparison

Table 4 summarizes the results obtained on the various datasets. For each dataset, the performance for SVM and RF are reported for both the proposed methodology (denoted as Our method) and for the sentiment embeddings produced by [9] (denoted as Dong and De Melo). The outcomes of the proposed methodology were achieved by setting K to 5 and α to 0.4. Subsection 5.3 discusses the effect of the input parameters on the performance of the proposed method.

The proposed methodology for cross-lingual sentiment propagation performs better than the method proposed by [9] in terms of macro-F_1 score on the majority of the analyzed datasets (Russian reviews are the only exception).

To gain insight into classifiers' performance, we explore also the abilities of the classifier to correctly assign each class label (i.e., precision) as well as to recognize the largest extend of the test samples labeled with each class (i.e., recall). Table 5 reports the macro-precision and macro-recall values (indicating the means of per-class precision and recall values, respectively) [15]. Based on the achieved results, we can conclude that classifier performance is not biased towards any of the aforesaid metrics. Interestingly, the embeddings produced by [9] show higher precision for multiple languages, but the recall is often worse than those achieved by the proposed method (relying on the unified latent space model).

5.3 Parameter analysis

We study also the effect of setting different values for parameters K and α on the quality of the generated embeddings. To do so, we separately analyze their impact on the macro-F_1 scores achieved by the binary classifiers. Hereafter, for the sake of brevity, we will report only the results achieved on a representative dataset (IT_2). It is the largest and more balanced dataset among all the tested ones. Similar results were achieved on the other datasets.

The parameter K indicates the number of neighbors considered while linking the words in the original to those in the target language. The higher K, the more word relationships are included in the word graph. As a drawback, when K is relatively high, the model may include less relevant or unreliable links. Furthermore, since the connectedness of the graph increases, the complexity of the sentiment propagation process gets worse (see Section 5.4).

Figure 2 shows how the macro-F_1 score varies as K increases, for $\alpha = 0.4$. The plot highlights a knee in the curve for $K = 5$. This implies that, for the purpose of sentiment classification, using a larger value of K does not yield significant performance

Dataset	Metric	Our method		Dong and De Melo	
		SVM	RF	SVM	RF
cs	Precision	0.7347	0.7326	0.7203	**0.7547**
	Recall	**0.7593**	0.712	0.7474	0.7177
de	Precision	0.6797	0.7481	0.6563	**0.7735**
	Recall	**0.7372**	0.6766	0.7131	0.6507
es	Precision	0.6111	**0.7747**	0.4010	0.6181
	Recall	**0.6154**	0.5428	0.5	0.5172
fr	Precision	0.7025	**0.7301**	0.6488	0.6784
	Recall	0.7019	**0.7309**	0.6403	0.6760
it	Precision	**0.8494**	0.8168	0.6750	0.8030
	Recall	**0.8071**	0.7765	0.7637	0.6336
nl	Precision	**0.6868**	0.6651	0.6059	0.6491
	Recall	**0.704**	0.6317	0.6162	0.6022
ru	Precision	0.6805	0.6623	0.7151	**0.7362**
	Recall	0.7221	0.6025	**0.7634**	0.6845
IT_1	Precision	**0.8441**	0.8425	0.7442	0.7314
	Recall	**0.8439**	0.8424	0.7436	0.7312
IT_2	Precision	**0.8442**	0.8428	0.7416	0.7495
	Recall	**0.8441**	0.8427	0.7415	0.7495

Table 5: Results in terms of macro-precision and macro-recall, for embeddings generated by the proposed methodology (Our method) and with those introduced in [9] (Dong and De Melo)

improvements. Notice that, to remove the less reliable links, the word graph is early pruned by enforcing the cut-off threshold value α. The impact of the pruning phase is higher while setting high K values.

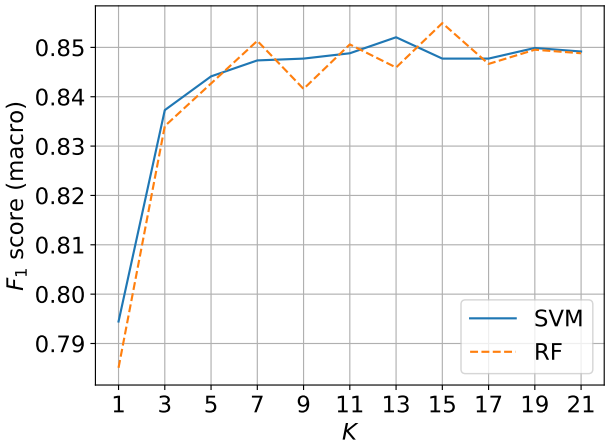

Figure 2: Macro-F_1 score as a function of K, on dataset IT_2

We separately analyze also the impact of parameter α. Enforcing low α values potentially introduces a bias in the graph due to the presence of "noisy" links, whereas setting high α values limits word graph connectedness. Given an edge $(x, x', w_{xx'}) \in E$, the edge weight $w_{xx'}$ is computed as the cosine similarity between x and x' [15]. The cosine similarity takes (absolute) values between 0 (orthogonal vectors) and 1 (parallel vectors). Hence, α has the same value range. Figure 3 shows how the macro-F_1 score varies as α increases. The lower bound set for α (approximately 0.4) is the best we can manage using the hardware resources currently in use (i.e., setting lower values requires more computational

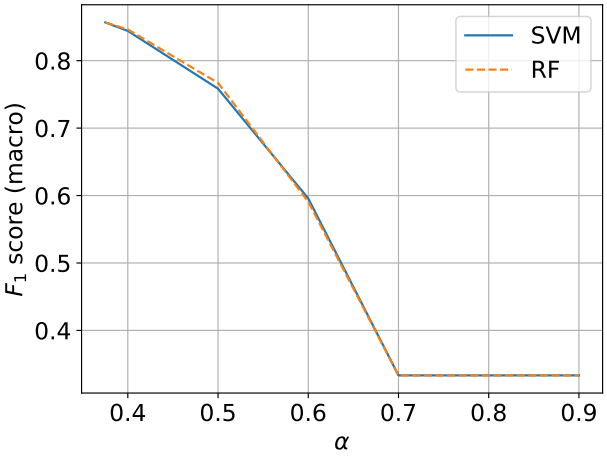

Figure 3: Macro-F_1 score as a function of α, on dataset IT_2

memory). Specifically, the empirical results show that, by setting α to 0.4, the sentiment propagation process converges to a satisfactory solution with the hardware resources available for this study. Subsection 5.4 provides a more detailed analysis of the space complexity of the problem. Setting lower α (limited graph pruning) values yields sentiment embeddings with a higher quality. Conversely, when high α values are set (specifically, for values of $\alpha \geq 0.7$), the pruning phase is not beneficial.

5.4 Complexity analysis

The most computationally intensive step of the proposed method is sentiment propagation on the word graph based on Stochastic Gradient Descent. It entails computing the gradient of the loss function described in Section 4 and then iteratively updating the sentiment polarities until a local minimum is reached.

To exploit the hardware optimizations available for matrix computations, the gradient can be computed on the entire matrix, rather than separately for each weight. Specifically, to process the information embedded in the word graph, an adjacency matrix A is defined. Each matrix value A_{ij} indicates the weight of the edge linking two arbitrary words x_i and x_j. If the edge does not exist, the corresponding matrix value is zero. Since graph connectedness is bounded by the cut-off threshold α, the adjacency matrix is rather sparse (the higher α, the sparser A). To compute the gradient of the adopted loss function, the adjacency matrix of the word graph is loaded into main memory. Large sparse matrices can be efficiently loaded into main memory by storing only the non-zero elements. However, this limits the types of operations that can be performed. Hence, in most cases, a denser in-memory representation would be needed.

Let N be the size of the initial vocabulary V_o in the original language (English, in our case). For each word in V_o, K neighbor words are selected from the target language. Hence, in the worst case, the word graph contains $(K + 1)N$ words. Since part of the neighbors in the target languages are overlapped, we can assume, to a good approximation, that each word in the original language has one translation in the target language, yielding $2N$ words in the resulting graph. The corresponding adjacency matrix consists of $4N^2$ cells. Let us assume to use B bytes to represent each floating point number (where $B = 4$ or $B = 8$ in modern systems), the total adjacency matrix size is $4BN^2$. On the other hand, the size N of the initial English vocabulary ranges between 80,000 and 100,000. Thus, the required memory allocation ranges between 100 and 300 GB.

A possible way to optimize the process is to identify connected sub-graphs and to run the Stochastic Gradient Descent separately on each sub-graph. It is potentially feasible because nodes and edges external from the connected sub-graph would not influence sentiment propagation within the sub-graph. This reduces the size of the processed adjacency matrices, which are stored into main memory. However, when graphs are highly connected (as in our case), the optimization is not very beneficial. Therefore, as discussed in Section 5.3, in the experimental evaluation reported in this study we have decided to limit the computational complexity of the propagation process by properly setting the K and α parameters.

6 CONCLUSIONS AND FUTURE WORKS

This paper presents a in-progress research study on the use of a bilingual latent space to propagate sentiment information across multiple languages. The proposed approach overcomes the limitations of the solutions previously proposed in literature due to the dependence of the propagation phase on the bilingual lexicon. Our claim is that relying on latent word relationships (embedding lexicon information as well) would enhance the process of sentiment propagation in cross-lingual and multi-domain contexts.

We have empirically compared the sentiment embeddings generated by the proposed methodology with those produced by the approach presented in [9]. Specifically, the embeddings have been exploited to tackle a binary sentiment analysis problem. The results confirm the initial claim: for most of the considered languages, the propagated information yields better results.

The presented study leaves room for several extensions. Firstly (and most importantly), we aim at extending the Deep Learning process (based on a dual-channels CNN) presented by [9] by embedding the enhanced sentiment vector propagation phase. This allows us to fully explore the potential of the new methodology in a state-of-the-art Deep Neural Network Architecture for sentiment analysis.

A further exploration will be devoted to identifying the optimal setting of the α parameter. We plan not only to increase the computational power but also to study more sophisticated strategies to optimize the propagation phase as well as to design greedy strategy able to overcome the limitations due to the iterative optimization process.

Finally, we plan to test further multilingual datasets. Since most of the publicly available datasets are small- or medium-sized and quite imbalanced, we aim at crawling, releasing, and testing new data related to various domains and written in different languages.

7 ACKNOWLEDGEMENTS

This work has been partially supported by the SmartData@PoliTO center on Big Data and Data Science.

REFERENCES

[1] Oscar Araque, Ignacio Corcuera-Platas, J. Fernando Sánchez-Rada, and Carlos A. Iglesias. 2017. Enhancing deep learning sentiment analysis with ensemble techniques in social applications. *Expert Systems with Applications* 77 (2017), 236 – 246. https://doi.org/10.1016/j.eswa.2017.02.002

[2] Oscar Araque, Ignacio Corcuera-Platas, J. Fernando Sánchez-Rada, and Carlos A. Iglesias. 2017. Enhancing deep learning sentiment analysis with ensemble techniques in social applications. *Expert Systems with Applications* 77 (2017), 236 – 246. https://doi.org/10.1016/j.eswa.2017.02.002

[3] Carmen Banea, Rada Mihalcea, Janyce Wiebe, and Samer Hassan. 2008. Multilingual subjectivity analysis using machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Honolulu, Hawaii, USA, 127–135.

[4] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching Word Vectors with Subword Information. *Transactions of the Association for Computational Linguistics* 5 (2017), 135–146.

[5] Yanqing Chen and Steven Skiena. 2014. Building sentiment lexicons for all major languages. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Association for Computational Linguistics, Baltimore, Maryland, USA, 383–389.

[6] Gerard de Melo. 2014. Etymological Wordnet: Tracing The History of Words. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*. European Language Resources Association (ELRA), Reykjavik, Iceland, 1148–1154. http://www.lrec-conf.org/proceedings/lrec2014/pdf/1083_Paper.pdf

[7] Erkin Demirtas and Mykola Pechenizkiy. 2013. Cross-lingual polarity detection with machine translation. In *Proceedings of the Second International Workshop on Issues of Sentiment Discovery and Opinion Mining*. ACM, Chicago, USA, 9.

[8] Hai Ha Do, PWC Prasad, Angelika Maag, and Abeer Alsadoon. 2019. Deep Learning for Aspect-Based Sentiment Analysis: A Comparative Review. *Expert Systems with Applications* 118 (2019), 272 – 299. https://doi.org/10.1016/j.eswa.2018.10.003

[9] Xin Dong and Gerard de Melo. 2018. Cross-Lingual Propagation for Deep Sentiment Analysis. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18)*. AAAI Press, New Orleans, Louisiana, USA, 5771–5778.

[10] Kevin Duh, Akinori Fujino, and Masaaki Nagata. 2011. Is machine translation ripe for cross-lingual sentiment classification?. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2*. Association for Computational Linguistics, Portland, Oregon, USA, 429–433.

[11] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. 2016. *Deep Learning*. MIT Press. http://www.deeplearningbook.org.

[12] Hitkul Jangid, Shivangi Singhal, Rajiv Ratn Shah, and Roger Zimmermann. 2018. Aspect-Based Financial Sentiment Analysis Using Deep Learning. In *Companion Proceedings of the The Web Conference 2018 (WWW '18)*. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, 1961–1966. https://doi.org/10.1145/3184558.3191827

[13] Armand Joulin, Piotr Bojanowski, Tomas Mikolov, Hervé Jégou, and Edouard Grave. 2018. Loss in Translation: Learning Bilingual Word Mapping with a Retrieval Criterion. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*.

[14] Ji Young Lee and Franck Dernoncourt. 2016. Sequential Short-Text Classification with Recurrent and Convolutional Neural Networks. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, San Diego, California, 515–520. https://doi.org/10.18653/v1/N16-1062

[15] Jure Leskovec, Anand Rajaraman, and Jeffrey David Ullman. 2014. *Mining of Massive Datasets* (2nd ed.). Cambridge University Press, New York, NY, USA.

[16] Bing Liu. 2015. *Sentiment Analysis - Mining Opinions, Sentiments, and Emotions*. Cambridge University Press. http://www.cambridge.org/us/academic/subjects/computer-science/knowledge-management-databases-and-data-mining/sentiment-analysis-mining-opinions-sentiments-and-emotions

[17] Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. 2013. Linguistic Regularities in Continuous Space Word Representations. In *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 9-14, 2013, Westin Peachtree Plaza Hotel, Atlanta, Georgia, USA*. 746–751. https://www.aclweb.org/anthology/N13-1090/

[18] Aliaksei Severyn and Alessandro Moschitti. 2015. Twitter Sentiment Analysis with Deep Convolutional Neural Networks. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '15)*. Association for Computing Machinery, New York, NY, USA, 959–962. https://doi.org/10.1145/2766462.2767830

[19] David Vilares, Carlos Gomez-Rodriguez, and Miguel A. Alonso. 2017. Universal, unsupervised (rule-based), uncovered sentiment analysis. *Knowledge-Based Systems* 118 (2017), 45 – 55. https://doi.org/10.1016/j.knosys.2016.11.014

[20] Xiaojun Wan. 2009. Co-Training for Cross-Lingual Sentiment Classification. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*. Association for Computational Linguistics, Suntec, Singapore, 235–243. https://www.aclweb.org/anthology/P09-1027