Do Neighbours Help? An Exploration of Graph-based Algorithms for Cross-domain Sentiment Classification

Natalia Ponomareva
Statistical Cybermetrics Research group,
University of Wolverhampton, UK
nata.ponomareva@wlv.ac.uk

Mike Thelwall
Statistical Cybermetrics Research group,
University of Wolverhampton, UK
m.thelwall@wlv.ac.uk

Abstract

This paper presents a comparative study of graph-based approaches for cross-domain sentiment classification. In particular, the paper analyses two existing methods: an optimisation problem and a ranking algorithm. We compare these graph-based methods with each other and with the other state-of-the-art approaches and conclude that graph domain representations offer a competitive solution to the domain adaptation problem. Analysis of the best parameters for graph-based algorithms reveals that there are no optimal values valid for all domain pairs and that these values are dependent on the characteristics of corresponding domains.

1 Introduction

The sentiment classification (SC) is an active area of research concerned automatic identification of sentiment strength or valence of texts. SC of product reviews is commercially important and widely researched but it typically needs to be optimised separately for each type of product (i.e. domain). When domain-specific data are absent or insufficient the researchers usually seek solution in semi-supervised, unsupervised or cross-domain approaches. In this paper, we focus on cross-domain methods in order to take advantage of the huge amount of annotated sentiment data available on the Internet. Our aim is to find out to what extent it is possible to learn sentiment phenomena from these data and transfer them to new domains rather than induce them from scratch for each new domain. Previous research has shown that models trained on one data usually give much worse results on another, especially when both data sets belong to completely different domains. This is largely because the sentiment words and their valences depend a lot on the domain where they are expressed. The first problem concerns the words that can convey opposite sentiments with respect to the context or domain. For example, a word "ridiculous" in book reviews may express a negative meaning when talking about a book content, however for reviews on electronics this word can bear a positive meaning when talking about prices. Another and more common problem is related to sentiment words that are specific for each domain. For instance, words like "boring", "inspiring", "engaging" are very common in book reviews but it is almost impossible to find them in reviews on electronics. At the same time, the electronics domain can contain words like "defective", "refund", "return", "customer service", which are very unusual for book reviews.

Several cross-domain approaches have been suggested recently to solve the problem of accuracy loss in cross-domain sentiment classification, namely Structural Correspondence Learning (SCL) (Blitzer et al., 2007), the graph-based approach (Wu et al., 2009) and Spectral Feature Alignment (SFA) (Pan et al., 2010). In this paper, we explore graph-based algorithms which refer to a group of techniques that model data as a graph of documents. This data representation takes into account not only document contents but also document connectivity which is modeled as document sentiment similarity rather than content similarity. Our interest in graph

Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, pages 655–665, Jeju Island, Korea, 12–14 July 2012.

algorithms is two-fold. First, graph-based domain representations can benefit from two independent sources of information: scores given by a machine learning technique which indicate the probability of a document to belong to a sentiment class and similarity relations between documents. Second, unlike other suggested methods, this approach can be easily adapted to multiple classes, which makes it possible to classify documents using finer-grained sentiment scales.

Different graph-based algorithms have been applied to several SA tasks (Pang and Lee, 2005; Goldberg and Zhu, 2006; Wu et al., 2009), but no comparison has been made to find the most appropriate one for SC. Moreover, in the framework of the domain adaption task, we come across the problem of choosing the best set of parameters, which, as we further demonstrate, depends on the characteristics of a corresponding domain pair. Unfortunately, no study has investigated this problem. (Pang and Lee, 2005; Goldberg and Zhu, 2006) exploited the graph-based approach for a semi-supervised task and experimented with data belonging to one domain and, therefore did not come across this issue. The work of (Wu et al., 2009) lacks any discussion about the choice of the parameter values; the authors set some values equal for all domains without mentioning how they obtained these numbers.

The present research brings several contributions. First, we compare two graph-based algorithms in cross-domain SC settings: the algorithm exploited in (Goldberg and Zhu, 2006), which seeks document sentiments as an output of an optimisation problem (OPTIM) and the algorithm adopted by (Wu et al., 2009), that uses ranking to assign sentiment scores (RANK). Second, as document similarity is a crucial factor for satisfactory performance of graph-based algorithms, we suggest and evaluate various sentiment similarity measures. Sentiment similarity is different from topic similarity as it compares documents with respect to the sentiment they convey rather than their topic. Finally, we discover the dependency of algorithm parameter values on domain properties and, subsequently, the impossibility to find universal parameter values suitable for all domain pairs. We discuss a possible strategy for choosing the best set of parameters based on our previous study (Ponomareva and Thelwall, 2012), where we introduced two domain characteristics: domain similarity and domain complexity and demonstrated their strong correlation with cross-domain accuracy loss.

The rest of the paper is structured as follows. In Section 2 we give a short overview of related works on cross-domain SC. Section 3 describes and compares the OPTIM and RANK algorithms. In Section 4 we discuss an issue of document similarity and select document representation that correlates best with document sentiments. Experimental results are described in Section 5 followed by a discussion on the strategy for choosing the best parameter values of the algorithms (Section 6). Finally, in Section 7 we summarise our contributions and discuss further research.

2 Related work

Cross-domain sentiment analysis has received considerable attention during the last five years and, since then, several approaches to tackle this problem have emerged. The most straightforward approach is to use an ensemble of classifiers as tested in several works (Aue and Gamon, 2005; Li and Zong, 2008). It is a well-explored technique in machine learning concerned with training classifiers on domains where annotated data are available and then, combining them in ensembles for the classification of target data. Aue and Gamon (2005) studied several possibilities to combine data from domains with known annotations and came up with the conclusion that an ensemble of classifiers in a meta-classifier gives higher performance than a simple merge of all features.

Structural Correspondence Learning (SCL) (Blitzer et al., 2007) is another domain transfer approach, which was also tested on parts of speech (PoS) tagging (Blitzer et al., 2006). Its underlying idea is to find correspondences between features from source and target domains through modeling their correlations with pivot features. Pivot features are features occurring frequently in both domains, which, at the same time, serve as good predictors of document classes, like the general sentiment words “excellent” and “awful”. The extraction

of pivot features was made on the basis of their frequency in source and target corpora and their mutual information with positive and negative source labels. The correlations between the pivot features and all other features were modeled using a supervised learning of linear pivot predictors to predict occurrences of each pivot in both domains. The proposed approach was tested on review data from 4 domains (books, DVDs, kitchen appliances and electronics) and demonstrated a significant gain of accuracy for most domain pairs compared to the baseline. However, for a few domains the performance degraded due to feature misalignment: the narrowness of the source domain and diversity of the target domain created false projections of features in the target domain. The authors proposed to correct this misalignment with a small amount of annotated in-domain data.

Spectral Feature Alignment (SFA), introduced by Pan et al. (2010), holds the same idea as SCL, i.e., an alignment of source and target features through their co-occurrences with general sentiment words. But instead of learning representations of pivots in source and target domains the authors used spectral clustering to align domain-specific and domain-independent words into a set of feature-clusters. The constructed clusters were then used for the representation of all data examples and training the sentiment classifier. This new solution yields a significant improvement on cross-domain accuracy compared with SCL for almost all domain pairs.

The method suggested by Bollegala et al. (2011) also relies on word co-occurrences. In particular, the authors presented a method for automatic construction of a sentiment-sensitive thesaurus where each lexical element (either unigram or bigram) is connected to a list of related lexical elements which most frequently appear in the context expressing the same sentiment. This thesaurus is then used on the training step to expand feature vectors with related elements to overcome the feature mismatch problem. The method was tested on the same data set as SCL and SFA but unlike previous works the authors used a combination of domains to create sentiment-sensitive thesauri and to train the cross-domain classifier. They compare the accuracy of their approach with an average accuracy over the results with the same target domain given by SCL and SFA, and concluded that their method surpasses all existing approaches. However, we think that such a comparison is not optimal. Indeed, using the approach described in (Ponomareva and Thelwall, 2012) we can choose the most appropriate data for training our classifier rather than averaging the results given by all data sets. Therefore, instead of average accuracies, the best accuracies with respect to the same target domain should be compared. This comparison leads to opposite conclusions, namely that SCL and SFA significantly outperform the sentiment-sensitive thesaurus-based method.

Unlike the approaches mentioned above, graph-based algorithms exploit relations between documents for finding the correct document scores. We describe them in more details in the next section.

3 Graph-based algorithms

In this section we present and compare 2 graph-based algorithms which use similar graph structures but completely different methods to infer node scores. The RANK algorithm (Wu et al., 2009) is based on node ranking, while OPTIM (Goldberg and Zhu, 2006) determines solution of graph optimisation problem. Initially OPTIM was applied for the rating-inference problem in a semi-supervised setting. This study, for the first time, analyses its behaviour for cross-domain SC and compares its performance with a similar approach.

3.1 OPTIM algorithm

The OPTIM algorithm represents graph-based learning as described in (Zhu et al., 2003). Let us introduce the following notation:

- $G = (V, E)$ is an undirected graph with $2n$ nodes V and weighted edges E.
- L stands for labeled data (source domain data) and U for unlabeled data (target domain data).
- x_i is a graph node which refers to a document, $f(x_i)$ is a true label of a document which is supposed to be unknown even for annotated documents, allowing for noisy labels. Each $x_i \in L$ is connected to y_i which represents a given rating of a document. The edge weight between $x - i$ and y_i is a large number

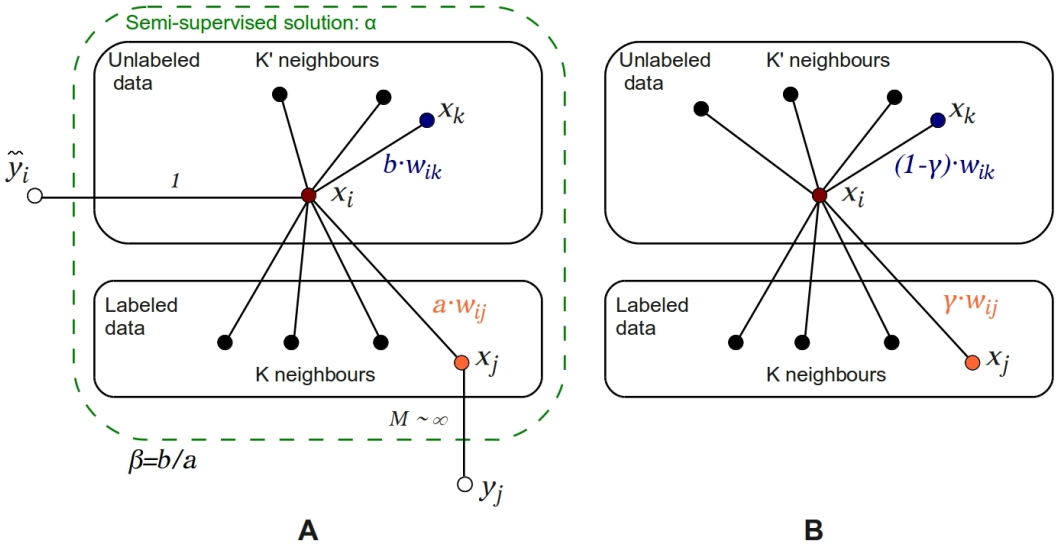

Figure 1: Graph models for the OPTIM (A) and RANK (B) algorithms

M introducing the hard constraints between labeled documents and their ratings. Each $x_i \in U$ is connected to $\hat{y}_i$ that stands for predicted rating of a document. The edge weight between x_i and $\hat{y}_i$ is equal to 1.

- Each unlabeled document x_i is connected to its k nearest labeled documents $kNN_L(i)$ (source domain neighbours). The weight between x_i and $x_j \in kNN_L(i)$ is measured by a given similarity w and denoted $a \cdot w_{ij}$.
- Each unlabeled document x_i is connected to its k' nearest unlabeled documents $k'NN_U(i)$ (target domain neighbours). The weight between x_i and $x_j \in k'NN_U(i)$ is denoted by $b \cdot w_{ij}$.

Figure 1A illustrates the graph structure described. The algorithm is based on the assumption that the rating function $f(x)$ is smooth with respect to the graph, so there are no harsh jumps of sentiment between nearest neighbours. To satisfy the smoothness condition sentiment variability between the closest nodes should be minimised. Another requirement is to minimise the difference between each initial node rating and its final value, although in the case of unlabeled nodes this is optional. Taking into consideration the conditions mentioned the sentiment-inference problem can be formulated as an optimisation problem:

$$\mathcal{L}(f) = \sum_{i \in L} M(f(x_i) - y_i)^2 + \sum_{i \in U} (f(x_i) - \hat{y}_i)^2 + \sum_{i \in U} \sum_{j \in kNN_L(i)} a w_{ij} (f(x_i) - f(x_j))^2 + \sum_{i \in U} \sum_{j \in k'NN_U(i)} b w_{ij} (f(x_i) - f(x_j))^2 \rightarrow min \quad (1)$$

After the substitutions $\alpha = ak + bk'$ and $\beta = \frac{b}{a}$ the final optimisation problem can be written as:

$$\mathcal{L}(f) = \sum_{i \in L} M(f(x_i) - y_i)^2 + \sum_{i \in U} [(f(x_i) - \hat{y}_i)^2 + \frac{\alpha}{k + \beta k'} (\sum_{j \in kNN_L(i)} w_{ij} (f(x_i) - f(x_j))^2 + \sum_{j \in k'NN_U(i)} \beta w_{ij} (f(x_i) - f(x_j))^2)] \rightarrow min \quad (2)$$

where β defines the relative weight between labeled and unlabeled neighbours, while α controls the weight of the graph-based solution with respect to the primarily obtained supervised sentiment scores.

The minimum-loss function which gives the solution of the optimisation problem can be found by setting the gradient to zero. For more details on the problem solution see (Goldberg and Zhu, 2006).

3.2 RANK algorithm

The RANK algorithm has a similar graph structure (Figure 1B): nodes represent labeled and unlabeled documents and there is a parameter (in this case γ) that controls the relative importance of labeled data over unlabeled data and is an analogue of β in OPTIM. The weight of edges between different nodes is also measured by document similarity. However, there are no edges between nodes and their initial sentiments because RANK is an iterative algorithm and each iteration gives new scores to unlabeled nodes while labeled nodes remain constant. More precisely, on each iteration sentiment scores of unlabeled documents are updated on the basis of the weighted sum of sentiment scores of the nearest labeled neighbours and the nearest unlabeled neighbours. The process stops when convergence is achieved, i.e. the difference in sentiment scores is less than a predefined tolerance.

Using the same notation as for OPTIM we can formulate the iterative procedure in the following way:

$$f_k(x_i) = \sum_{j \in kNN_L(i)} \gamma w_{ij} f(x_j) + \sum_{j \in k'NN_U(i)} (1-\gamma) w_{ij} f_{k-1} x_j) \quad (3)$$

where $f_k(x_i)$ is the node sentiment score on the k-th iteration. Document scores are normalised after each iteration to ensure convergence (Wu et al., 2009). It is worth noting that initially the authors did not consider having a different number of neighbours for the source and target domains.

Analysing differences in the graph structures and assumptions of both models we can say that they are almost identical. Even the smoothness condition holds for the RANK algorithm as the score of a node is an averaged sum of the neighbours. The only principal difference concerns the requirement of closeness of initial and final sentiment scores for OPTIM. This condition gives more control on the stability of the algorithm performance.

4 Measure of document similarity

A good measure of document similarity is a key factor for the successful performance of graph-based algorithms. In this section we propose and evaluate several measures of document similarity based on different vector representations and the cosine of document vectors.

Following (Goldberg and Zhu, 2006) and (Pang and Lee, 2005) we consider 2 types of document representations:

- feature-based: this involves weighted document features. The question here concerns the features to be selected. When machine learning is employed the answer is straightforward: the most discriminative features are the best ones for our task. However, we assume that we do not know anything about the domain when measuring sentiment similarity and, thus, we should establish the appropriate set of features only relying on our prior knowledge about sentiment words. According to previous studies, adjectives, verbs and adverbs are good indicators of sentiment (Pang and Lee, 2008), therefore, we keep only unigrams and bigrams that contain these PoS. We test two feature weights - tfidf and idf (F_{tfidf} and F_{idf} in Table1 respectively). The evident drawback of such a vector representation concerns the discarding of nouns, which in many cases also bear sentiments. To overcome this issue we introduce a new measure that uses sentiment dictionaries to add nouns expressing sentiments ($F_{idf+SOCAL}$).

- lexicon-based: uses sentiment dictionaries to assign scores to lexical elements of two types: words or sentences. The dimension of the corresponding document vector representation conforms with the granularity of the sentiment scale. For example, in case of binary sentiment scales, a document vector consists of two dimensions, where first component corresponds to the percentage of positive words (sentences) and the second component - to the percentage of negative words (sentences). To assign sentiment scores to lexical elements we exploit different sentiment resources, namely

domain	F_{tfidf}	F_{idf}	$F_{idf+SOCAL}$	W_2	W_{10}	S_2
BO	0.61	0.62	0.64	0.49	0.50	0.44
DV	0.61	0.61	0.64	0.56	0.56	0.51
EL	0.62	0.66	0.68	0.47	0.49	0.46
KI	0.65	0.67	0.68	0.51	0.54	0.53

Table 1: Correlation for various similarity measures with sentiment scores of documents across different domains.

SentiWordNet (Esuli and Sebastiani, 2006), SO-CAL (Taboada et al., 2010) and SentiStrength (Thelwall et al., 2012). The scores of sentences are averaged by the number of their positive and negative words. Preliminary experiments show a big advantage of SO-CAL-dictionaries comparing with other resources. SentiWordNet demonstrates quite an unsatisfactory performance, while SentiStrength, being very precise, has an insufficient scope and, therefore, finds no sentiment in a substantial number of documents.

The best document representation is selected on the basis of its correlation with the sentiment scores of documents. To compute correlations for feature-based measures, we take 1000 features with highest average tfidf weights. Table 1 gives the results of a comparison for two document representations and their different settings. Here W_2 and S_2 stand for word-based and sentence-based representations of dimension 2 and W_{10} - for word-based representation of dimension 10. All use SO-CAL-dictionaries to assign scores to words or sentences. Feature-based representations demonstrate significantly better correlations with document sentiments although for some domains, like DV, the lexical element-based representation produces a similar result. Integration of SO-CAL-dictionaries gives insignificant contribution into the overall correlation, which maybe due to the limited number of features participated in the analysis. In our further experiments we use both F_{idf} and $F_{idf+SOCAL}$ document representations.

5 Experimental results

Our data comprises Amazon product reviews on 4 topics: books (BO), electronics (EL), kitchen (KI) and DVDs (DV), initially collected and described by Blitzer et al. (2007). Reviews are rated using a binary scale, 1-2 star reviews are considered as negative and 4-5 star reviews as positive. The data within each domain are balanced: they contain 1000 positive and 1000 negative reviews.

First, we compute a baseline for each domain pair by training a Support Vector Machines (SVMs) classifier using one domain as training data and another as test data. We choose SVMs as our main learning technique because they have proved to be the best supervised algorithm for SC (Pang and Lee, 2008). In particular, we use the LIBSVM library (Chang and Lin, 2011) and a linear kernel function to train the classifier. For the feature set we experiment with different features and feature weights and conclude that unigrams and bigrams weighted with binary values yield the best performance.

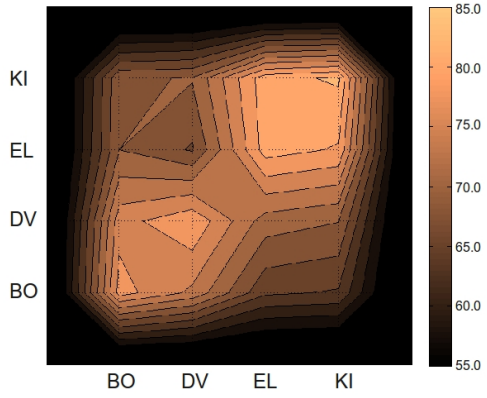

Figure 2: Baseline accuracy for cross-domain SC. (x-axis - source domains, y-axis - target domains).

Figure 2 presents an isoline image of cross-domain accuracies for all domain pairs.[1] Products on the x-axis represent source domains and products

[1]We should point out that in the images the shading between points is not intended to suggest interpolation but is used to highlight the overall pattern. Of course the pattern depends on a domain order on the axes, therefore, similar domains are placed together to make the regions with high and low accuracies evident.

on the y-axis represent target domains. The isolines image of the baseline accuracy delivers a good representation of domain relations. In particular, we can observe two regions with the highest accuracy (EL-KI, KI-EL) and (BO-DV, DV-BO) and two regions with a big performance drop (EL-BO, EL-DV, KI-BO, KI-DV) and (BO-EL, BO-KI, DV-EL, DV-KI). As shown in our previous study (Ponomareva and Thelwall, 2012) the first two regions conform with the most similar domain pairs BO, DV and EL, KI.

OPTIM and RANK require the setting of several parameters: (k, k', α, β) for OPTIM and (k, k', γ) for RANK. As it is computationally expensive to iterate over all possible values of parameters we first run the algorithms on a small matrix of parameters and then apply the gradient descent method which takes the values with highest accuracy as its starting points. We execute both algorithms with different similarity measures, F_{idf} and $F_{idf+SOCAL}$. In Table 2 OPTIM and RANK run with F_{idf}, while OPTIM+SOCAL and RANK+SOCAL run with $F_{idf+SOCAL}$. We give the best accuracies achieved by these algorithms for each domain pair. Unlike the correlations, the accuracies increase significantly with the integration of SO-CAL-dictionaries, the average improvement is about 3% for RANK and 1.5% for OPTIM. In general, RANK consistently outperforms OPTIM for all domain pairs, OPTIM shows competitive performance only for the pairs of similar domains BO-DV, KI-EL and EL-KI. We should also point out that OPTIM is more time-consuming as it requires expensive matrix operations. Due to these advantages of the RANK algorithm, we mostly focus on its analysis in the rest of the paper.

It is interesting to examine the performance of RANK on the basis of the 3D isolines image (Figure 3B). The isolines stretch from left to right indicating that accuracy is almost independent of the source domain. Such behaviour for RANK suggests a positive answer to our question stated in the title: even if domains are quite different, neighbours from the same domain will fix these discrepancies. This property is definitely a big advantage of the RANK algorithm in the context of the cross-domain task as it minimises the importance of the source domain. Obviously more experiments with different data must be accomplished to prove this conclusion with a higher level of confidence.

We also compare graph-based algorithms with other state-of-the-art approaches, such as SCL and SFA (Table 2, Figure 3). The best results in Table 2 are highlighted and if the difference is statistically significant with $\alpha = 0.05$ the corresponding accuracy is underlined. Note that we compare graph-based approaches against the others but not each other, therefore, if the result given by RANK is underlined it means that it is statistically significant only in comparison with SCL and SFA and not with OPTIM. According to Table 2, RANK surpasses SCL for almost all domain pairs with an average difference equal to 2%. Interestingly, without using SO-CAL-dictionaries RANK loses to both SCL and SFA for almost all domain pairs. The advantage of RANK over SFA is disputable as there is not much consistency about when one algorithm outperforms another, except that SFA is better overall for close domains. However Figure 3 suggests an interesting finding: that for domains with different complexities swapping source and target also changes the method that produces the best performance. A comparison of RANK and SCL on the Chinese texts given by (Wu et al., 2009) shows the same phenomenon. It seems that RANK works better when the target domain is simpler, maybe because it can benefit more from in-domain neighbours of the less rich and ambiguous domain. In the future, we plan to increase the impact of lexically different but reliably labeled source data by implementing the SFA algorithm and measuring document similarity between feature clusters rather than separate features.

6 Strategy for choosing optimal parameters

The results of the RANK and OPTIM algorithms presented in the previous section represent the highest accuracies obtained after running gradient descent method. Table 3 lists the best parameter values of the RANK algorithm over several domain pairs. Our attempt to establish some universal values valid for all domain pairs was not successful as the choice of the parameters depends upon the domain properties. Of course, in real life situations we do

source-target	baseline	OPTIM	RANK	OPTIM+ SOCAL	RANK+ SOCAL	SCL	SFA
BO-EL	70.0	74.0	77.2	74.4	**79.8**	77.5	72.5
BO-DV	76.5	78.6	77.4	79.9	79.8	75.8	**81.4**
BO-KI	69.5	74.6	78.6	77.3	**82.8**	78.9	78.8
DV-BO	74.4	78.8	78.9	80.5	**82.1**	79.7	77.5
DV-EL	67.2	73.6	78.8	74.4	**80.9**	74.1	76.7
DV-KI	70.2	75.6	80.4	77.3	**83.2**	81.4	80.8
EL-BO	65.5	67.8	69.9	69.5	73.6	75.4	**75.7**
EL-DV	71.3	74.2	72.6	75.6	77.0	76.2	**77.2**
EL-KI	81.6	83.6	83.2	85.7	85.3	85.9	**86.8**
KI-BO	64.7	68.4	70.9	69.7	**74.8**	68.6	**74.8**
KI-DV	70.1	72.3	72.4	73.4	**78.4**	76.9	77.0
KI-EL	79.7	82.6	81.9	83.7	83.7	**86.8**	85.1
average	71.7	75.3	76.9	76.8	**80.1**	78.1	78.7

Table 2: Comparison of different cross-domain algorithms

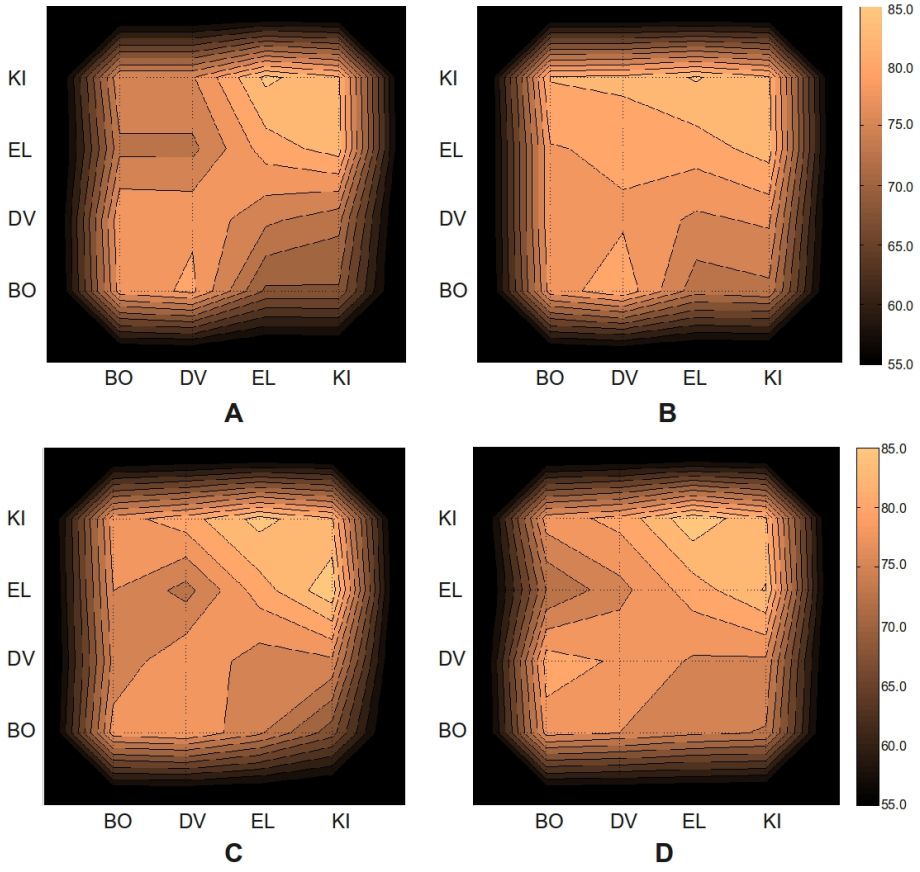

Figure 3: Accuracy obtained with different cross-domain algorithms over various domains: A) OPTIM, B) RANK, C) SCL, D) SFA. (x-axis - source domains, y-axis - target domains).

parameter	BO-EL	BO-DV	BO-KI	EL-BO	EL-DV	EL-KI
γ	0.34	0.78	0.30	0.50	0.55	0.9
k	50	100	25	75	50	200
k'	220	50	40	100	150	10

Table 3: Best number of labeled and unlabeled neighbours for the RANK algorithm over various domain pairs

source-target	similarity	complexity variance	γ
BO-EL	1.23	-1.93	0.34
BO-DV	1.75	0.06	0.76
BO-KI	1.17	-1.26	0.48
DV-BO	1.75	-0.06	0.75
DV-EL	1.22	-1.99	0.52
DV-KI	1.18	-1.32	0.44
EL-BO	1.23	1.93	0.62
EL-DV	1.22	1.99	0.68
EL-KI	1.87	0.67	0.75
KI-BO	1.17	1.26	0.64
KI-DV	1.18	1.32	0.54
KI-EL	1.87	-0.67	0.76

Table 4: Similarity, complexity variance and γ averaged over the best results (confidence level of 95%) of the RANK algorithm. The values are given on various domain pairs

not have a knowledge of the parameter values which produce the best performance and, therefore, it would be useful to elaborate a strategy for choosing the optimal values with respect to a corresponding domain pair. In our previous work (Ponomareva and Thelwall, 2012) we introduced two domain characteristics: domain similarity and domain complexity variance and proved their impact into the cross-domain accuracy loss. Domain similarity and complexity are independent properties of a domain pair as the former measures similarity of data distributions for frequent words, while the latter compares the tails of distributions. In Ponomareva and Thelwall (2012), we tested various metrics to estimate these domain characteristics. As a result, inversed χ^2 was proved to be the best measure of domain similarity as it gave the highest correlation with the cross-domain accuracy drop. The percentage of rare words (words that occur less than 3 times) was found to be the closest approximation to domain complexity as it showed the highest correlation with the in-domain accuracy drop.

It is naturally to assume that if domain similarity and complexity are responsible for the cross-domain accuracy loss, they might influence on the parameter values of domain adaptation algorithms. This is proved to be true for the γ parameter, whose values averaged over the top results of the RANK algorithm are listed in Table 4. We use the confidence interval of 95% to select the top values of γ. Table 4 shows that γ is the lowest for dissimilar domains with a simpler target (negative values of domain complexity variance), which means that the RANK algorithm benefits the most from unlabeled but simpler data. γ grows to values close to 0.6 for dissimilar domains with more complex target (positive values of domain complexity variance), which shows that the impact of simpler source data, though different from target, increases. Finally γ reaches its maximum for similar domains with the same level of complexity. Unfortunately, due to comparable amount of data for each domain, no cases of similar domains with different complexity are observed. We plan to study these particular cases in the future.

High dependency of γ on both domain characteristics is proved numerically. The correlation between γ and domain similarity and complexity reaches 0.91, and decreases drastically when one of these characteristics is ignored.

Concerning the optimal number of labeled and unlabeled neighbours, no regularity is evident (Table 3). In our opinion, that is an effect of choosing the neighbours on the basis of the quantitative threshold. Nevertheless, different domains have distinct pairwise document similarity distributions. Figure 4 demonstrates similarity distributions for BO, EL and DV inside and across domains. Therefore, taking into account only the quantitative threshold we ignore discrepancies

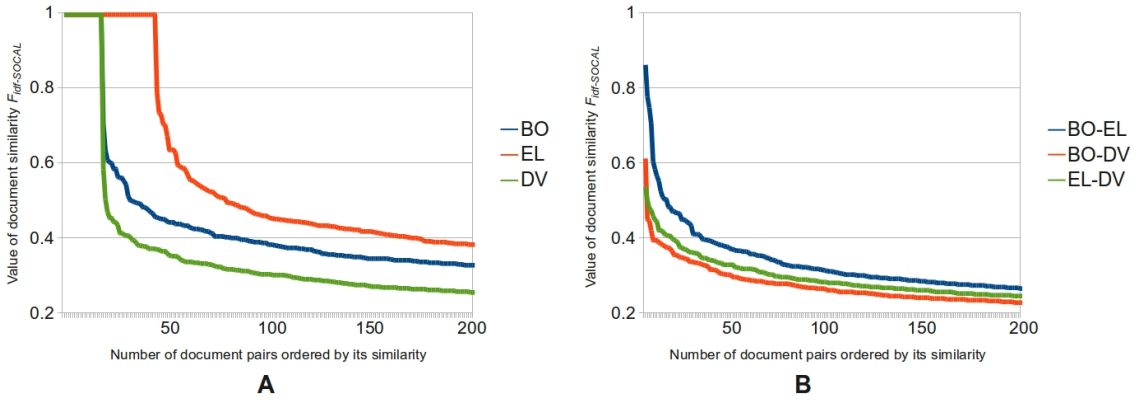

Figure 4: Pairwise document similarity distributions inside domains (A) and across domains (B)

in graph connectivities inside and across domains and may bring "bad" neighbours to participate in decision-making. In our further research we plan to explore the idea of a qualitative threshold, which chooses neighbours according to their similarity and uses the same similarity levels for in-domain and cross-domain graphs.

7 Conclusions and future work

This paper has studied the performance of two graph-based algorithms, OPTIM and RANK when applied to cross-domain sentiment classification. Comparison on their performance on the same data has revealed that, in spite of the similar graph structures, RANK consistently produces better results than OPTIM. We also have compared the graph-based algorithms with other cross-domain methods, including SCL and SFA, and concluded that RANK considerably outperforms SCL and obtains better results than SFA for half of the cases. Given that we consider only the best accuracies obtained with RANK, such comparison is not completely fair but it shows the potential of the RANK algorithm once the strategy for choosing its optimal parameters is established. In this paper, we also discuss some ideas about how to infer optimal parameter values for the algorithms on the basis of domain characteristics. In particular, the strong correlation for γ with domain similarity and complexity has been observed. Unfortunately we are not able to find any regularity in the number of source and target domain neighbours, which we think is the result of the qualitative approach to selecting the closest neighbours.

As a result of this research we have identified the following future directions. First, we plan to improve the RANK performance by choosing the number of neighbours on the basis of the document similarity threshold which we set equal for both in-domain and cross-domain neighbours. We expect that this modification will diminish the number of "bad" neighbours and allow us to reveal a dependency of similarity threshold on some domain properties. Another research direction will focus on the integration of SFA into the similarity measure to overcome the problem of lexical discrepancy in the source and target domains. Finally, as all our conclusions have been drawn on a data set of 12 domain pairs, we plan to increase a number of domains to verify our findings on larger data sets.

Acknowledgments

This work was supported by a European Union grant by the 7th Framework Programme, Theme 3: Science of complex systems for socially intelligent ICT. It is part of the CyberEmotions project (contract 231323).

References

Anthony Aue and Michael Gamon. 2005. Customizing sentiment classifiers to new domains: A case study. In *Proceedings of Recent Advances in Natural Language Processing (RANLP '05)*.

John Blitzer, Ryan McDonald, and Fernando Pereira. 2006. Domain adaptation with structural

correspondence learning. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing (EMNLP '06)*, pages 120–128.

John Blitzer, Mark Dredze, and Fernando Pereira. 2007. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics (ACL '07)*, pages 440–447.

Danushka Bollegala, David Weir, and John Carroll. 2011. Using multiple sources to construct a sentiment sensitive thesaurus for cross-domain sentiment classification. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL '11)*, pages 132–141.

Chih-Chung Chang and Chih-Jen Lin. 2011. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

Andrea Esuli and Fabrizio Sebastiani. 2006. Sentiwordnet: A publicly available lexical resource for opinion mining. In *Proceedings of the 5th Conference on Language Resources and Evaluation (LREC 06)*, pages 417–422.

Andrew B. Goldberg and Xiaojin Zhu. 2006. Seeing stars when there aren't many stars: graph-based semi-supervised learning for sentiment categorization. In *Proceedings of the First Workshop on Graph Based Methods for Natural Language Processing (TextGraphs '06)*, pages 45–52.

Shoushan Li and Chengqing Zong. 2008. Multi-domain sentiment classification. In *Proceedings of ACL-08: HLT, Short Papers*, pages 257–260.

Sinno Jialin Pan, Xiaochuan Niz, Jian-Tao Sunz, Qiang Yangy, and Zheng Chen. 2010. Cross-domain sentiment classification via spectral feature alignment. In *Proceedings of International World Wide Web Conference (WWW '10)*.

Bo Pang and Lillian Lee. 2005. Seeing stars: exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics (ACL '05)*, pages 115–124.

Bo Pang and Lillian Lee. 2008. Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, 2(1-2):1–135.

Natalia Ponomareva and Mike Thelwall. 2012. Bibliographies or blenders: Which resource is best for cross-domain sentiment analysis? In *Proceedings of the 13th Conference on Intelligent Text Processing and Computational Linguistics (CICLing '12)*.

M. Taboada, J. Brooke, M. Tofiloski, K. Voll, and M. Stede. 2010. Lexicon-based methods for sentiment analysis. *Computational Linguistics*, 37(2):267–307.

M. Thelwall, K. Buckley, and G. Paltoglou. 2012. Sentiment strength detection for the social web. *Journal of the American Society for Information Science and Technology*, 63(1):163–173.

Qiong Wu, Songbo Tan, and Xueqi Cheng. 2009. Graph ranking for sentiment transfer. In *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, pages 317–320.

Xiaojin Zhu, Zoubin Ghahramani, and John Lafferty. 2003. Semi-supervised learning using gaussian fields and harmonic functions. In *Proceedings of the 20th International Conference on Machine Learning (ICML '03)*, pages 912–919.