Single and Cross-domain Polarity Classification using String Kernels

Rosa M. Giménez-Pérez[1], **Marc Franco-Salvador**[1,2], and **Paolo Rosso**[1]
[1] Universitat Politècnica de València, Valencia, Spain
[2] Symanto Research, Nuremberg, Germany
rogipe2@upv.es, marc.franco@symanto.net, prosso@upv.es

Abstract

The polarity classification task aims at automatically identifying whether a subjective text is positive or negative. When the target domain is different from those where a model was trained, we refer to a cross-domain setting. That setting usually implies the use of a domain adaptation method. In this work, we study the single and cross-domain polarity classification tasks from the string kernels perspective. Contrary to classical domain adaptation methods, which employ texts from both domains to detect pivot features, we do not use the target domain for training. Our approach detects the lexical peculiarities that characterise the text polarity and maps them into a domain independent space by means of kernel discriminant analysis. Experimental results show state-of-the-art performance in single and cross-domain polarity classification.

1 Introduction

The polarity classification task, also known as (binary) polarity or sentiment categorisation, aims at identifying whether a subjective text is positive or negative depending on the overall sentiment detected. Single domain polarity classification (Pang et al., 2002) refers to the standard text classification setting (Sebastiani, 2002). The cross-domain level (Blitzer et al., 2007) refers to classify a different domain from that or those where a model was trained.

These tasks have become especially important for business purposes. The vastness and accessibility of the Internet produced a new generation of event and product reviewers. These reviewers employ channels such as blogs, fora or social media. In consequence, companies are highly interested into identifying reviewers' opinions on, for instance, new products in order to improve marketing campaigns.

Although polarity classification tasks can be tackled with text classification methods, it has been proven to be a more challenging task (Pang et al., 2002): sentiment may be expressed more subtly (Reyes and Rosso, 2013) than categories generally recognised with keywords alone. In addition, the cross-domain variant has the additional difficulty of using a different vocabulary among domains. This problem is usually drawn by means of domain adaptation techniques (Ben-David et al., 2007). Most of these techniques exploit pivot features that allow to map vocabularies among domains.

String kernels are known for their good performance in text classification (Lodhi et al., 2002). Recent works with this representation demonstrated its excellent capacity to capture lexical peculiarities of text (Popescu and Grozea, 2012; Ionescu et al., 2014). In this work we study the single and cross-domain polarity classification tasks from the string kernels perspective. The research questions we aim to answer are:

- *What is the performance of string kernels for single and cross-domain polarity classification?* We are interested in the performance of this representation in these specially challenging classification tasks. Despite the use of string kernels is not new at single-domain level (Bespalov et al., 2011), this is, to the best of our knowledge, the first attempt to use them at cross-domain level. This leads us to our next research question.

- *Can this representation classify at cross-domain level without learning from texts of the target domain?* We employ Kernel Discriminant Analysis (Mika et al., 1999) for the classification, which is based on a non-linear space transformation. We aim to clarify if

Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers, pages 558–563,
Valencia, Spain, April 3-7, 2017.

the lexical peculiarities captured by this approach characterise the polarity of the texts independently of the domain.

In order to answer these questions, we compare our approach with several state-of-the-art methods with the well-known Multi-Domain Sentiment Dataset (Blitzer et al., 2007). Experimental results show state-of-the-art performance in single and cross-domain polarity classification. In addition, the stability of the proposed approach is remarkable among the different evaluated domains.

2 Related Work

In this section we review the state-of-the-art methods which have been evaluated in the Multi-Domain Sentiment dataset. Focused on single-domain polarity classification, the Confidence-Weighted Learning (CWL) (Dredze et al., 2008) is based on updating more aggressively the weights of features with higher confidence. The Structural Correspondence Learning with Mutual Information (SCL-MI) (Blitzer et al., 2007) was the first model evaluating the dataset at cross-domain level. The mutual information was used to select pivot features which are subsequently used for measuring co-occurrence with the rest of the features. Chen et al. (2012) addressed this task, considering the scalability and the computational cost of the approach, with marginalized stacked denoising autoencoders. The use of neural networks has also been proven to be useful for cross-domain classification tasks where unlabeled data from the test domain is employed to extract domain independent features (Ganin et al., 2016). Some approaches have proven to excel both at single and cross-domain levels. Bollegala et al. (2013) proposed the Sentiment-Sensitive Thesaurus (SST) model that groups together words expressing the same sentiment. Recently, the Knowledge-Enhanced Meta classifier (KE-Meta) (Franco-Salvador et al., 2015) combined surface and word sense disambiguation features derived from a semantic network.

3 String Kernels

String Kernels (SK) are functions that measure the similarity of string pairs at lexical level. Their dual representation allows to work with a huge number of character n-grams while keeping the feature space reduced.

In this work, we follow the implementation and formulation of Ionescu et al. (2014).[1] A simple measure of the similarity of two strings s,t is the number of shared substrings of length p. The p-grams kernel is estimated as follows:

$$\mathrm{k}_p(s,t) = \sum_{v \in L^p} f(\mathrm{num}_v(s), \mathrm{num}_v(t)), \quad (1)$$

where $\mathrm{num}_v(s)$ is the number of occurrences of string v as a substring of s, p is the length of v, and L is the alphabet used to generate v. The function $f(x,y)$ variates depending on the type of kernel:

1. $f(x,y) = x \cdot y$ in the p-spectrum kernel;
2. $f(x,y) = \mathrm{sgn}(x) \cdot \mathrm{sgn}(y)$ in the p-grams presence bits kernel;[2]
3. $f(x,y) = \min(x,y)$ in the p-grams intersection bits kernel.

As we can see, the values of $f(.)$ are the highest with the spectrum kernel and the lowest with the presence kernel. This gives us an idea about what these kernels capture. The spectrum kernel offers high values even when the texts are only partially related. The intersection kernel employs the n-gram frequency to provide with a precise lexical similarity measure. Finally, the presence kernel captures the lexical *core meaning* of the texts by smoothing the n-gram repetitions.

Our kernels combine different n-gram lengths[3] (see Section 4.2 for details about our parameter selection) and are normalised as follows:

$$\hat{\mathrm{k}}(s,t) = \frac{\mathrm{k}(s,t)}{\sqrt{\mathrm{k}(s,s) \cdot \mathrm{k}(t,t)}} \quad (2)$$

We perform the classification with Kernel Discriminant Analysis (KDA) (Baudat and Anouar, 2000),[4] which returns the eigenvector matrix U. We compute the feature matrices $Y = KU$ and $Y_\mathrm{t} = K_\mathrm{t}U$, where K and K_t are the training and test instance kernels. For each class c, we create the prototype Y_c as the average of all vectors of Y that correspond to the instances of class c.

[1] http://string-kernels.herokuapp.com/

[2] sgn is the sign function.

[3] We combine the n-gram lengths by adding the kernel values obtained for each n.

[4] We use the following KDA implementation: http://www.cad.zju.edu.cn/home/dengcai/Data/DimensionReduction.html

Finally, we classify each test instance by identifying the class of the prototype with the lowest mean squared error between $Y_t(i)$ and Y_c. Key to our cross-domain classification, without learning from texts of the target domain, is the KDA's space transformation. It employs *the kernel trick* (Schölkopf, 2001) and formulates the task as an eigenvalue problem resolution to learn non-linear mappings which transform our features to a new space that captures the most relevant lexical peculiarities for polarity classification.

4 Evaluation

In this section we evaluate and compare our approach in the single and the cross-domain polarity classification tasks.

4.1 Dataset and Tasks Setting

Dataset We employ the Multi-Domain Sentiment Dataset (v. 2.0) (Blitzer et al., 2007).[5] It contains Amazon product reviews of four different domains: Books (B), DVDs (D), Electronics (E) and Kitchen appliances (K). Each review contains information including a rating in a range of 0 to 5 stars. Reviews rated with more than 3 stars were labeled as positive, and those with less than 3 as negative. There are 1,000 positive and 1,000 negative reviews for each domain.

Methodology We evaluate our approach using the presence ($k_p^{0/1}$), intersection ($k_p^{\cap}$), and spectrum (k_p) kernels. We compare with SST and KE-Meta at single and cross-domain levels (see Section 2). In addition, we compare with CWL at single-domain and with SCL-MI at cross-domain level.[6] Finally, we include as a baseline the combination of word unigram, bigram, and trigram features using a support vector machine classifier with linear kernel (henceforth referred to as word n-g). We perform our evaluation with a stratified 10-fold cross-validation. We use the accuracy of classification as the evaluation metric. Statistically significant results according to a χ^2 test are highlighted in bold.

4.2 Parameter Selection

We adjusted the kernel n-gram length and the KDA's regularisation factor α with a 80-20% splitting over the nine training folds of each cross-validation iteration. We first set α to its default value (0.2) and explored different combinations of n-gram lengths, for $2 \leq n \leq 10$. The best results where obtained when we combined all the n-grams in $5 \leq n \leq 8$. Using that combination, we tested for $\alpha \in [0.01, 1]$. The results notably differed depending on the task setting, training domain, and kernel (see Figure 1). We use the parameters adjusted in this section for the rest of our evaluation.

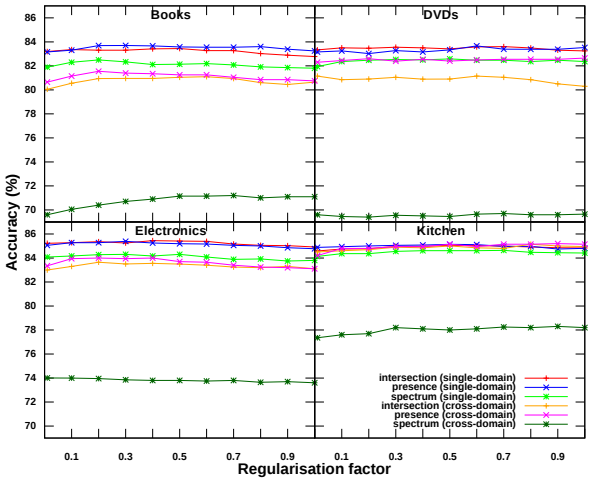

Figure 1: Avg. accuracy among all the fold values depending on the KDA's regularisation factor.

Method	Books	DVDs	Electronics	Kitchen
KE-Meta	**83.5**	82.3	82.6	84.2
SST	80.4	82.4	84.4	**87.7**
CWL	**82.6**	80.9	**85.9**	85.7
word n-g	80.5	81.7	80.3	81.9
SK($k_p^{0/1}$)	**83.8**	**84.8**	**86.2**	85.5
SK($k_p^{\cap}$)	**83.8**	**84.6**	**86.6**	85.4
SK(k_p)	**82.7**	82.8	84.7	85.3

Table 1: Single-domain polarity classification accuracy (in %).

4.3 Single-domain Polarity Classification

In Table 1 we show the single-domain results. As we can see, the state-of-the-art performance differs depending on the domain. The combination of word n-grams makes word n-g the baseline in all the domains. KE-Meta excels with book reviews, SST with kitchen appliance reviews, and CWL with book and electronic reviews. Franco-Salvador et al. (2015) analysed this fact and justified it with the difference in review length and

[5] https://www.cs.jhu.edu/~mdredze/datasets/sentiment/

[6] The results of the compared approaches are taken from Franco-Salvador et al. (2015).

Method	Books	DVDs	Electronics	Kitchen
KE-Meta	77.9	**80.4**	78.9	82.5
SST	76.3	78.3	**83.9**	**85.2**
SCL-MI	74.6	76.3	78.9	82.0
word n-g	74.4	79.8	77.1	76.9
SK($k_p^{0/1}$)	**82.0**	**81.9**	**83.6**	**85.1**
SK($k_p^{\cap}$)	**80.7**	**80.7**	**83.0**	**85.2**
SK(k_p)	71.2	69.0	73.7	78.0

Table 2: Multi-source cross-domain polarity classification accuracy (in %).

vocabulary richness among the evaluated domains. In addition, they highlighted the KE-Meta stability among domains, i.e., their higher lower-bound in accuracy. However, the results of our presence and intersection string kernels are more stable. What is more, depending on the domain, their results are statistically superior or equal to the best obtained by the state of the art. The exception is SST, which obtains the best results in the kitchen domain, where the shorter average review length could penalise other methods. We note that there are not statistically significant differences between the presence and intersection kernels. However, the spectrum kernel obtains lower results in all the cases. In contrast to the other two kernels, the spectrum one assigns a high score even when only one of the texts has a high frequency for a particular n-gram (see Section 3). This produces similar kernel representations for texts which may be not so close at lexical level and, consequently, penalises the model precision.

4.4 Cross-domain Polarity Classification

Following recent works in cross-domain polarity classification (Bollegala et al., 2013; Franco-Salvador et al., 2015), in Table 2 we compare with the state of the art using a multi-source cross-domain setting, i.e., we train with all the domains but the one we classify. Similarly to the single-domain results, word n-g is the baseline, KE-Meta offers higher results in book and DVD reviews, and SST in electronic and kitchen appliance reviews. We note that SCL-MI was designed for single-source cross-domain classification (Blitzer et al., 2007). Therefore, the use of multiple training domains may be the reason of its lower, but still competitive, performance.

Interestingly, despite not using target domain texts for training, the presence and intersection kernels obtain statistically superior or equal results to the best ones obtained by the state of the art. This proves that the non-linear mappings learned by KDA capture the lexical peculiarities that characterise polarity in a domain-independent way. We note again the stability of the results of these kernels and the non-existent statistically significant difference between them. In contrast, the spectrum kernel obtains the lowest results of the table. In order to analyse this fact, we perform an additional experiment where we use a single-source setting to train our cross-domain classifiers. We can see the results in Table 3.

The comparison of the multi-source and the single-source results shows that the presence and intersection kernels are occasionally able to exploit different domain characteristics to obtain better results, e.g. the presence and intersection kernels with kitchen reviews, and the presence kernel with DVDs reviews. Even in cases when the combination of domains do not lead to better results, the results remain close to those of the most compatible training domain; specially with the presence kernel. We note the relevance of the multi-source setting for the industry: it is easier to use multiple domains to learn a domain-independent classifier than to detect each time which is the most appropriated training domain. Finally, we observe that the spectrum kernel has competitive results when the most compatible domain is used for training. However, the aforementioned score characteristics of that kernel (see Sections 3 and 4.3) exponentially increase its error in the multi-source setting.

5 Conclusions

In this paper we studied the single and the cross-domain polarity classification tasks from the string kernels perspective. We analysed the performance of the presence, intersection, and spectrum kernels when classifying with kernel discriminant analysis. Experimental results compared to several state-of-the-art approaches in the Multi-Domain Sentiment Dataset showed state-of-the-art performance for the presence and intersection kernels in both tasks. In addition, these two kernels provided with the most stable results among domains. What is more, we showed that the non-linear space transformations of kernel discriminant analysis captured the lexical peculiarities that characterise polarity in a domain-independent way. This fact

Method	D→B	E→B	K→B	B→D	E→D	K→D
SK($k_p^{0/1}$)	82.0	72.4	72.7	81.4	74.9	73.6
SK($k_p^{\cap}$)	82.1	72.4	72.8	81.3	75.1	72.9
SK(k_p)	81.1	69.9	71.4	80.0	73.5	71.8
	B→E	**D→E**	**K→E**	**B→K**	**D→K**	**E→K**
SK($k_p^{0/1}$)	71.3	74.4	83.9	74.6	75.4	84.9
SK($k_p^{\cap}$)	71.8	74.5	84.4	74.9	75.1	84.9
SK(k_p)	70.7	72.6	83.9	74.2	74.9	84.5

Table 3: SK single-source cross-domain polarity classification accuracy (in %), where each column header follows the "training domain → test domain" format.

allowed our approaches to excel at cross-domain level without learning from texts of the target domain. Finally, the analysis of the single-source and the multi-source cross-domain results proved that the presence kernel tolerates better the inclusion of new training domains in the multi-source cross-domain setting. This fact makes it the recommended option for cross-domain polarity classification.

Future work will investigate further how to employ string kernels for single and cross-domain classification tasks.

Acknowledgments

We thank Ionescu et al. (2014) for their support and comments. The work of the third author was partially supported by the SomEMBED TIN2015-71147-C2-1-P MINECO research project and by the Generalitat Valenciana under the grant ALMAMATER (PrometeoII/2014/030).

References

Gaston Baudat and Fatiha Anouar. 2000. Generalized discriminant analysis using a kernel approach. *Neural computation*, 12(10):2385–2404.

Shai Ben-David, John Blitzer, Koby Crammer, Fernando Pereira, et al. 2007. Analysis of representations for domain adaptation. *Advances in neural information processing systems*, 19:137.

Dmitriy Bespalov, Bing Bai, Yanjun Qi, and Ali Shokoufandeh. 2011. Sentiment classification based on supervised latent n-gram analysis. In *Proceedings of the 20th ACM International Conference on Information and Knowledge Management (CIKM'11)*, pages 375–382, Glasgow, Scotland, UK. ACM.

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*, pages 440–447, Prague, Czech Republic, June. Association for Computational Linguistics.

Danushka Bollegala, David Weir, and John Carroll. 2013. Cross-domain sentiment classification using a sentiment sensitive thesaurus. *IEEE transactions on knowledge and data engineering*, 25(8):1719–1731.

Minmin Chen, Zhixiang Xu, Kilian Weinberger, and Fei Sha. 2012. Marginalized denoising autoencoders for domain adaptation. In *Proceedings of the 29th International Conference on Machine Learning (ICML'12)*, pages 767–774, Edinburgh, Scotland.

Mark Dredze, Koby Crammer, and Fernando Pereira. 2008. Confidence-weighted linear classification. In *Proceedings of the 25th International Conference on Machine Learning (ICML'08)*, pages 264–271, Helsinki, Finland. ACM.

Marc Franco-Salvador, Fermín L. Cruz, José A. Troyano, and Paolo Rosso. 2015. Cross-domain polarity classification using a knowledge-enhanced meta-classifier. *Knowledge-Based Systems*, 86:46 – 56.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. 2016. Domain-adversarial training of neural networks. *Journal of Machine Learning Research*, 17(59):1–35.

Radu-Tudor Ionescu, Marius Popescu, and Aoife Cahill. 2014. Can characters reveal your native language? A language-independent approach to native language identification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1363–1373, Doha, Qatar, October. Association for Computational Linguistics.

Huma Lodhi, Craig Saunders, John Shawe-Taylor, Nello Cristianini, and Chris Watkins. 2002. Text classification using string kernels. *Journal of Machine Learning Research*, 2(Feb):419–444.

Sebastian Mika, Gunnar Rätsch, Jason Weston, Bernhard Schölkopf, and Klaus-Robert Müller. 1999. Fisher discriminant analysis with kernels. In *Proceedings of IEEE Neural Networks for Signal Processing Workshop (NNSP'99)*, pages 41–48, Madison, Wisconsin, USA.

Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up? sentiment classification using machine learning techniques. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing*, pages 79–86, Philadelphia, Pennsylvania, USA, July. Association for Computational Linguistics.

Marius Popescu and Cristian Grozea. 2012. Kernel methods and string kernels for authorship analysis. In *Online Working Notes/Labs/Workshop Papers of the CLEF 2012 Evaluation Labs (CLEF'12)*, Rome, Italy.

Antonio Reyes and Paolo Rosso. 2013. On the difficulty of automatically detecting irony: beyond a simple case of negation. *Knowledge and Information Systems*, pages 1–20.

Bernhard Schölkopf. 2001. The kernel trick for distances. *Advances in neural information processing systems*, 13:301–307.

Fabrizio Sebastiani. 2002. Machine learning in automated text categorization. *ACM computing surveys (CSUR)*, 34(1):1–47.