Transfer Learning Using Twitter Data for Improving Sentiment Classification of Turkish Political News

Mesut Kaya, Guven Fidan and I Hakkı Toroslu

Abstract In this paper, we aim to determine the overall sentiment classification of Turkish political columns. That is, our goal is to determine whether the whole document has positive or negative opinion regardless of its subject. In order to enhance the performance of the classification, transfer learning is applied from unlabeled Twitter data to labeled political columns. A variation of self-taught learning has been proposed, and implemented for the classification. Different machine learning techniques, including support vector machine, maximum entropy classification, and Naive-Bayes has been used for the supervised learning phase. In our experiments we have obtained up to 26 % increase in the accuracy of the classification with the inclusion of the Twitter data into the sentiment classification of Turkish political columns using transfer learning.

1 Introduction

Social Media has become a global forum for people to express their subjective thoughts, opinions, and feelings. People express their opinions about almost anything like products, social events, news etc. People are curious about other peoples opinions. In the news domain, in general, people are still more interested in the opinions of a special group of experts, namely newspaper columnists rather than ordinary peoples comments on social media. With the rapid growth of Twitter among people,

M. Kaya · I. H. Toroslu (✉)
Department of Computer Engineering, Middle East Technical University, Ankara, Turkey
e-mail: e1502509@ceng.metu.edu.tr|mesut.kaya@agmlab.com

Guven Fidan
R&D Department AGMLab, Ankara, Turkey
e-mail: guven.fidan@agmlab.com

I. H. Toroslu
e-mail: toroslu@ceng.metu.edu.tr

E. Gelenbe and R. Lent (eds.), *Information Sciences and Systems 2013*,
Lecture Notes in Electrical Engineering 264, DOI: 10.1007/978-3-319-01604-7_14,

almost all of the columnists and journalists have Twitter accounts and share their personal opinions informally on Twitter. Therefore, it is possible to analyze columnists' opinions and feelings both from Twitter data and from their newspaper columns.

In our previous work [16], through several experiments we have shown that it is possible to obtain better sentiment classification results for Turkish political columns by providing a list of effective words and increasing the weight of these features within the model created by using the training data. Besides, one of the difficulties of the sentiment classification of news data is the lack of tagged data. Since the columns are not short texts, the annotation task is difficult and expensive. In order to have a good performance from sentiment classification, large amount of annotated data is needed.

In order to provide a wide effective words list and overcome the lack of tagged data problem, in this work, we adapt the *shape transfer learning* approach, aiming to extract the knowledge from source tasks to be applied to a target task [1]. In this paper, features are transferred from Twitter domain to news domain in an unsupervised way. The idea is to extract important features (such as, unigrams) from columnists' Twitter accounts and use them in the training phase of the sentiment classification of political columns. By using unlabeled data from Twitter domain, the need and the effort to collect more training data can be reduced and the performance of classifiers can be increased.

The content of this paper is as follows: In Sect. 2 related works and the literature are reviwed. In Sect. 3, transfer learning methodology is explained with the details of the algorithms used and background information is given. In Sect. 4, experimental setup and the evaluation metrics used are covered, and detailed analyses of the evaluations are given. Finally, In Sect. 5, the work is concluded.

2 Related Work

In this section, we briefly summarize the related work on sentiment classification and its applications on the news domain.

In their book, Opinion Mining and Sentiment Analysis Pang and Lee provide a detailed survey of sentiment analysis from natural language processing (NLP) and Machine Learning (ML) perspectives [2], and they also describe several application domains. News is one of them.

Viondhini and Chandrasekaran [17] states that in the text categorization Machine Learning techniques like naive bayes (NB), support vector machine (SVM) and maximum entropy(ME) have achieved great success. They also state that other used ML techniques in the NLP area are: K-nearest neighborhood (KNN), N-gram model.

There are some works on the application of the sentiment analysis to the news domain with different approaches [10–15]. One of the recent works in this domain is our recent work on the sentiment analysis of Turkish political columns [16]. As an initial work on Turkish political domain, sentiment classification techniques are incorporated into the domain of political news from columns in different Turkish news sites. The performances of different machine learning methods are measured and the problem of sentiment classification in news domain is discussed in detail.

There are lots of sentiment analysis techniques and different areas that these techniques are applied. Transfer learning and domain adaptation techniques are used widely in ML [3]. Transfer learning has been applied in many different research areas containing NLP problems, learning data across domains, image classification problems etc. [1].

One of the problems that transfer learning and domain adaptation are applied is sentiment classification. Ave and Gommon conducted an initial study to customize sentiment classifiers to new domains [4]. Bliter et al. extend structural correspondence learning (SCL) to sentiment classification to investigate domain adaptation for sentiment classifiers by focusing on online reviews for different types of products [5]. Li et al. outline a novel sentiment transfer mechanism based on constructed non-negative matrix tri-factorizations of term document matrices in the source and target domains [3].

In our work, different than the other domain adaptation and transfer learning methods applied in sentiment classification tasks, we use unlabeled data with unsupervised feature construction, and transferring knowledge from short text (Tweets) to long text (columns), which is not a common technique applied in transfer learning. Besides, our work is an initial work for applying transfer learning for sentiment classification of Turkish texts.

3 Transfer Learning Methodology

3.1 Background

3.1.1 Transfer Learning

Transfer Learning's main goal is to extract useful knowledge from one or more source tasks and to transfer the extracted information into a target task where the roles of source and target tasks are not necessarily the same [1].

In our work, we aim to solve sentiment classification of Turkish political columns (target task) by extracting and transferring features from unlabeled Twitter data in an unsupervised way (source task). Source domain is Twitter and contains unlabeled data; target domain is news and contains labeled data. Notice that source and target data does not share the class labels. Besides, the generative distribution of the labeled data is not the same as unlabeled data's distribution.

Our main motivation is the assumption that even unlabeled Twitter data collected from columnist's verified accounts may help us to learn important features in the politic news domain. By using this assumption, we use transfer learning. This kind of transfer learning is categorized as self-taught learning which is similar to inductive transfer learning. Self-taught learning was first proposed by Raina et al [6].

3.1.2 F-Score for Feature Ranking

In order to measure the importance of a feature for a classifier, we use F-Score (Fisher score) [8, 9]. F-Score has been chosen, since it is independent of the classifiers, so that we can use it for 3 different classifiers we use in experiments.

Given the training instances $x_i, i = 1, 2, 3, \ldots, l$ the F-score of the jth feature is defined as:

$$F(j) = \frac{(\bar{x}_j^{(+)} - \bar{x}_j)^2 + (\bar{x}_j^{(-)} - \bar{x}_j)^2}{\frac{1}{n_+ - 1}\sum_{i=1}^{n_+}(x_{i,j}^{(+)} - \bar{x}_j^{(+)})^2 + \frac{1}{n_- - 1}\sum_{i=1}^{n_-}(x_{i,j}^{(-)} - \bar{x}_j^{(-)})^2} \tag{1}$$

where n_+ and n_- are the number of positive and negative instances in the data set respectively; $\bar{x}_j$, $\bar{x}_j^+$ and $\bar{x}_j^-$ represents the averages of the jth feature of the whole positive-labeled and negative-labeled instances; $\bar{x}_{i,j}^+$ and $\bar{x}_{i,j}^-$ represent the jth feature of ith positive and negative instance. Larger F-Score means that the feature has more importance for the classifier.

3.1.3 TF-IDF Weighting

Term frequency-inverse document frequency measures how important a feature (word) to a document and it is used as weighting factor in text mining applications. Variations of tf-idf calculations are available, and in this work we use the following formulations:

Given a corpus D, a document d and a term t in that document term frequency, inverse term frequency and tf-idf are calculated by multiplying tf and idf, where tf is the number of times t occurs in d over total number of terms in d and idf is the logarithm of number of docs in D divided by number of documents in D that t occurs in.

3.2 Data Sets

In our experiments we use three different data sets, one from the news domain and the other two from Twitter domain. Articles from the news domain are collected via specific crawlers and annotated, we have 400 annotated columns. Tweets from columnists' Twitter accounts are collected by using Twitter4J API [1]. Search API used to collect all accessible tweets of the columnists. 123,074 tweets of columnists are collected. In order to collect random tweets, more than 100,000, Streaming API is used. The formulation of labeled news data that will be used in the rest of the paper is as follows:

[1] http://twitter4j.org/en/index.html

$$T = \left\{(x_l^{(1)}, y^{(1)}), (x_l^{(2)}, y^{(2)}) \ldots, (x_l^{(m)}, y^{(m)})\right\}$$

A news data is represented as $(x_l^{(j)}, y^{(j)})$ where $x_l^{(j)} = (f_1, f_2 \ldots, f_k)$ is a term vector of the text and each f_k is tf-idf value for features of the sample data and $y^{(j)} \in \{pos, neg\}$.

The formulation of unlabeled Twitter data collected from columnist's Twitter account U_1 and from random Twitter accounts U_2 that will be used in the rest of the paper are: $U_1 = \left\{z_{u1}^1, z_{u1}^2 \ldots, z_{u1}^a\right\}$ and $U_2 = \left\{z_{u2}^1, z_{u2}^2 \ldots, z_{u2}^b\right\}$. In U_1 and U_2, each z_{u1}^a corresponds to an unlabeled tweet of columnists' verified Twitter accounts and each z_{u2}^b corresponds to an unlabeled tweet of random Twitter accounts and they contain number of occurrences of each feature within tweet.

In both of the algorithms explained in detail below, by using less frequent and most frequent features in U_2 noisy features are eliminated from U_1. Then, by using filtered U_2, a list L_u of sorted features according to their occurrences in the all documents is generated. Then, the number of occurrences of the features are normalized using the log_x function (best x is chosen after several experiments). Actually the normalized list L_u contains feature and value pairs. Notice that, after normalization some features in L_u are eliminated. For the second algorithm described below, by using the labeled training set T, we calculate F-score of each feature and a list L_v of sorted features according to their F-scores is generated.

3.3 Algoritms

We propose two different approaches in the unsupervised construction of the transferred features:

3.3.1 Algorithm-1

Unsupervised feature construction without using the knowledge of the feature rankings within the classifier used. The algorithm used is given below:

Algorithm 1: Unsupervised feature construction without feature rankings

```
Input: $T$, $U_1$ and $U_2$
Output: Learned Classifier $C$ for Classification Task
Construct $L_u$ by using $U_1$ and $U_2$
Construct new labeled set $\bar{T} = \left\{(\bar{x}_l^{(i)}, y^{(i)})\right\}_{i=1}^{m}$ by $L_u$ and $T$
Learn a classifier $C$ by applying supervised learning algorithm (SVM, Naive Bayes
or Maximum Entropy).
return $C$
```

Without transfer learning tf-idf values for T are as follows:

$$\forall i \forall j \left(x_l^{(j)}(f_j)\right) = \frac{count\left((x_l^{(j)}(f_j))\right)}{j} \times idf \tag{2}$$

After applying steps 1 and 2, with transfer learning (we simply increase the term frequency of transferred features) the tf-idf in $\bar{T}$ are as follows:

$$\forall i \forall j \left(x_l^{(j)}(f_j)\right) = \frac{count\left((x_l^{(j)}(f_j))\right) + log_x\left(valueoff_j inL_u\right)}{j} \times idf \tag{3}$$

while constructing $\bar{T}$ only the transferred features of L_u, are included into $\bar{T}$. Namely, features in the target domain that do not appear in L_u are eliminated.

3.3.2 Algorithm-2

Unsupervised feature construction with using the knowledge of feature rankings within the classifier used. The algorithm is given below:

Algorithm 2: Unsupervised feature construction with feature rankings

```
Input: $T$, $U_1$ and $U_2$
Output: Learned Classifier $C$ for Classification Task
Construct $L_u$ by using $U_1$ and $U_2$
Use $T$ to calculate f-score of each feature and $L_v$.
Combine $L_u$ and $L_v$ and have a list $L_{u+v}$.
Transfer knowledge in $L_{u+v}$ to obtain $\bar{T} = \left\{(\bar{x}_l^{(i)}, y^{(i)})\right\}_{i=1}^{m}$
Learn a classifier $C$ by applying supervised learning algorithm (SVM, Naive Bayes or
  Maximum Entropy).
return $C$
```

By combining L_u and L_v a third list L_{u+v} is constructed as follows: $L_{u+v} = c_1 L_u + c_2 L_v$. In order to decide on optimal c_1 and c_2 values several experiments are conducted. Different than Algorithm-1, after transferring information the tf-idf in $\bar{T}$ become as follows:

$$\forall i \forall j \left(x_l^{(j)}(f_j)\right) = \frac{count\left((x_l^{(j)}(f_j))\right) + log_x\left(valueoff_j inL_{u+v}\right)}{j} \times idf \tag{4}$$

In Algorithm-2 $\bar{T}$ is constructed by using only the transferred features from L_{u+v}.

Table 1 Baseline Results

	NB	ME	SVM
Unigram	71.81	75.85	71.12
Unigram+adjective	71.81	75.59	72.95
Unigram+effective words	71.81	76.31	73.70

4 Evaluation

4.1 Experimental Setup

In the experiments, K-fold-cross-validation [7] is conducted by adopting K to be 3. 200 positive and 200 negative news items are used to make a 3-fold-cross-validation in the data experiments. Two experiments are adopted by using 3 different machine learning methods: namely NB, ME and SVM.

In order to have a baseline, sentiment classification of news columns are generated by using unigrams as features without transferring any knowledge from Twitter domain(for this purpose, we use the values from our previous work [16]). Table 1 shows the baseline results.

Transfer learning, adopted with and without the feature ranking information are applied to sentiment classification, and the results are compared with the results of the baseline. To evaluate the performance of the different experiments the following typical accuracy metric, which is commonly used in text classification, is used:

$$\text{Accuracy} = \frac{tp + tn}{tp + tn + fp + fn}$$

4.2 Results

In the first set of experiments, unsupervised feature construction for transfer learning is applied without using the feature ranking knowledge. The amount of transferred features is from 1 to 100 %. The classifier C is learned by using only transferred features.

Figure 1 shows the performances of 3 different machine learning methods for varying amount of transferred features. Notice that these results are for the cases in which only the transferred features are included. In other words, while transferring the knowledge, for creating the classifier features, only those which are in the L_u list are used. Features in the labeled data that are not in L_u are eliminated. In this case SVM performed better than NB and ME. When compared with the baseline results, for SVM there is a 5.67 % improvement. For NB there is small change and for ME there is a 4 % decrease. Therefore, by using only the transferred features without feature ranking knowledge provides significant information only for SVM.

Fig. 1 Accuracy values of classifiers with only transferred features

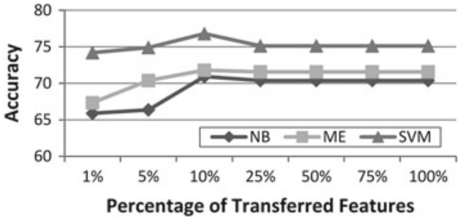


Fig. 2 Accuracy values for f-score of features included for different values of c1–c2

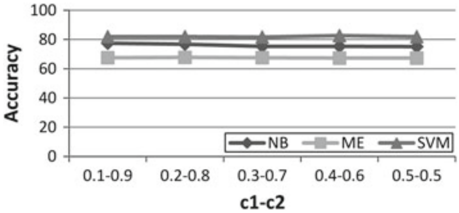
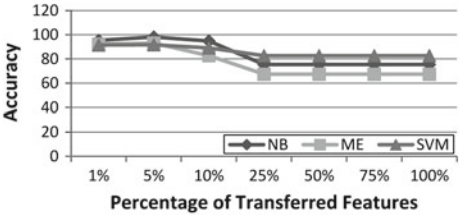


Fig. 3 Accuracy values for f-score of features included for c1 = 0.4 and c2 = 0.6



In the second set of experiments, feature rankings obtained by using F-scores are used for unsupervised feature construction. In the previous section, the details of the method used are explained. In Algorithm-2 features are combined as $L_{u+v} = c_1 L_u + c_2 L_v$ list. In Fig. 2, accuracy values for experiments are shown by using different c_1 and c_2 values. We observe that varying c_1 and c_2 values do not make significant changes. Therefore, we took $c_1 = 0.4$ and $c_2 = 0.6$ in the rest of the experiments.

Figure 3 shows the accuracy values when the amount of features transferred from constructed list $L_{(u+v)}$ varies. Combining two lists (L_u obtained from unlabeled data and L_v obtained from F-scores of the features) produces a very good performance gain. We can see from the Figure that especially transferring 5 % of constructed $L_{(u+v)}$ list provides very useful information for the classification task for all techniques that we have tried. For NB up to 98.116 % accuracy values are obtained. Comparing with the baseline results for NB, this corresponds to a 26.306 % performance gain. We observe a 19.435 % performance gain with a 93.135 % accuracy value for ME. Finally, for SVM a 15.43 % performance gain with a 91.74 % accuracy value is reached. However, if the amount of the transferred features are in 10–25 % range of list $L_{(u+v)}$, then the accuracy performance decreases for all techniques, and after that the results does not change. Roughly, we can say that features that carry important information for these classifiers are in the first 10 % of transferred features.

5 Discussion and Future Work

Although we transfer knowledge from short text to larger text and transfer features from unlabeled data in an unsupervised way, transfer learning method produced a very good improvement in the accuracy of the sentiment classification of Turkish political columns, over 90 %. In terms of relative performances, we see that in SVM, NB and ME, transferred information improves the performance. It is also observed that, transferring features that are not in the first 10 % of the transferred features decreases the performance.

We observe that the amount of transferred features do not make huge differences after a significant amount (25 %). Besides, in the second set of experiments conducted by using F-score information of the features the best results are obtained by transferring 1–10 % amount of features. This means that features carrying the most important information are the ones with higher frequency in the bag-of-words framework of transferred data.

An important outcome of this study is using feature ranking information (F score) combined with the unlabeled data turns out to be an effective method for transfer learning used in sentiment classification.

As future work, transferring from longer texts (different columns) can be analyzed, and using Transfer Learning in the Sentiment Classification of News Data with labeled data with Domain Adaptation techniques can be analyzed. Besides, using feature rankings together with unlabeled data can be adapted to different domains.

References

1. Pan SJ, Yang Q (2010) A Survey on Transfer Learning. IEEE Trans knowl data Eng 22(10):1345–1359
2. Pang B, Lee L (2008) Opinion mining and sentiment analysis. Found Trends Inf Retrieval 2(1):1–135
3. Li T, Sindhwani V, Ding C, Zhang Y (2010) Bridging domains with words: opinion analysis with matrix tri-factorizations. In: Proceedings of the Tenth SIAM Conference on Data Mining (SDM). pp 293–302
4. Aue A, Gamon M (2005) Customizing sentiment classifiers to new domains: a case study. http://research.micosoft.com/anthaue
5. Blitzer J, Dredze M, Pereira F (2007) Biographies, bollywood, boom-boxes and blenders: domain adaptation for sentiment classification. In: Proceedings of 45th Annual Meeting of the Association, Computational Linguistics. pp. 432–439.
6. Raina R, Battle A, Lee H, Pracker B, Ng AY (2007) Self-thaught learning: transfer learning from unlabeled data. In: Proceedings of 24th International Conference on, Machine Learning. pp 759–766.
7. Kohavi R (1995) A study of cross-validation and bootstrap for accuracy estimation and model selection. Proc 14th Int Joint Conf. Artif Intell 2(12):1137–1143
8. Chen YW, Lin CJ (2006) Combining SVMs with various feature selection strategies. In: Guyon I, Gunn S, Nikravesh M, Zadeh L (eds) Feature extraction, foundations and applications. Springer, Berlin

9. Chang YW, Lin CJ (2008) Feature ranking using linear SVM. In: JLMR, vol 3, WCCI2008 workshop on casuality, Hong Kong.
10. Fortuna B, Galleguillos C, Cristianini N (2009) Detecting the bias in the media with statistical learning methods. Theory and applications Yatlor and Francis Publisher, Text Mining
11. Evgenia B, van der Goot E (2009) News bias of online headlines across languages. Conference Proceedings, Lodz University Publishing House, The study of conflict between Russia and Georgia. Rhetorics of the media
12. Strapparava C, Mihalcea R (2007) (2007) Semeval 2007 task 14: affective text. In: Proceedings of ACL
13. Godbole N, Srinivasaiah M, Skiena S (2007) Large-scale sentiment analysis for news and blogs. In: Proceedings of the International Conference on Weblogs and Social media (ICWSM)
14. Balahur A, Steinberger R (2009) Rethinking sentiment analysis in the news: from theory to practice and back. WOMDA'09, pp 1–12.
15. Mullen T, Malouf R (2006) A prelimianry investigation into sentiment analysis of informal political discourse. Proceedings of the AAAI symposium on computational approaches to analyzing weblogs, In, pp 159–162
16. Kaya M, Toroslu IH, Fidan G (2012) Sentiment analysis of turkish political news. The 2012 IEEE/WIC/ACM International Conference on Web Intelligence.
17. Viondhini G, Chandrasekaran RM (2012) Sentiment analysis and opinion mining: a survey. Int J Advanced Res Comput Sci Softw Eng 2(6)