Transfer Learning in Large-Scale Short Text Analysis

Yan Chu[1(✉)], Zhengkui Wang[2], Man Chen[1], Linlin Xia[3], Fengmei Wei[1], and Mengnan Cai[1]

[1] College of Computer Science and Technology, Harbin Engineering University, Harbin, China
[2] Information Communications and Computing, Singapore Institute of Technology, Singapore, Singapore
[3] School of Automation Engineering, Northeast Dianli University, Chuanying, China
chuyan@hrbeu.edu.cn

Abstract. Transfer learning has emerged as a new learning technique facilitating an improved learning result of one task by integrating the well learnt knowledge from another related task. While much research has been devoted to develop the transfer learning algorithms in the field of long text analysis, the development of the transfer learning techniques over the short texts still remains challenging. The challenge of short text data analysis arises due to its sparse nature, noise words, syntactical structure and colloquial terminologies used. In this paper, we propose AutoTL(Automatic Transfer Learning), a transfer learning framework in short text analysis with automatic training data selection and no requirement of data priori probability distribution. In addition, AutoTL enables an accurate and effective learning by transferring the knowledge automatically learnt from the online information. Our experimental results confirm the effectiveness and efficiency of our proposed technique.

Keywords: Transfer learning · Long text analysis · Short text analysis · Latent semantic analysis

1 Introduction

Transfer learning is a new approach of improving the data learning result by utilizing the knowledge from different tasks and domains. The traditional machine learning or data mining approaches require the training and test data to be under the same feature space and the same distribution. Transfer learning, in contrast, allows the domains, tasks and distribution used in training and testing to be different. Specifically, when the training data in the target task are insufficient for a good data modeling, it transfers the useful knowledge from the related auxiliary data from another task to enrich the data features. In this case, more data characteristics are integrated into the data learning facilitating an improved learning results [1,2].

Much research has been devoted into the transfer learning in the domain of analyzing the long text data. To name a few, [3] proposed source free transfer

S. Zhang et al. (Eds.): KSEM 2015, LNAI 9403, pp. 499–511, 2015.
DOI: 10.1007/978-3-319-25159-2_45

learning to transfer knowledge from long texts to the long and [4] proposed latent dirichlet allocation to analyze two sets of topics on short and long texts. As the rapid development of Internet, more and more blog-sphere and social networking applications come into being, such as Microblog, Twitter, QQ news and online advertising. These applications exhibit two important features that differs themselves from traditional applications. First, data generated from these applications contain a lot of short texts, which contains rich useful information. Second, the text data vary dramatically every day, in terms of data size and data distribution. On one hand, these new features eventually challenge the traditional data mining and machine learning approaches, as the assumptions made do not hold in these new applications.On the other hand, the existing transfer learning algorithms tailored for long text analysis can not be directly applied in these application as well. The long text data analysis aims to analyze the long text data by utilizing the knowledge learnt from other long text datasets. The techniques are designed to handle the data that is well labeled, naturally compact and structured. However, the short text differs from the long text due to the sparse nature, noise words, syntactical structure and colloquial terminologies used, which result in unsatisfactory analysis results by directly using the transfer learning algorithms in the long text analysis domain.

In order to better utilize the short text data, it is essential to develop new transfer learning techniques in short text analysis. Given the fact that the result learnt from the long text analysis is enriched, one promising approach is to transfer the long text knowledge into the short text analysis. Several algorithms have been proposed under the similar methodology of utilizing the long text information to help the short text analysis. In their work, a major assumption is that source data are provided by the problem designers. This, however, would reduce the usability of these algorithms, as it requires the designers to have a well understanding of the source data. In addition, the prior probability distribution is required. In the big data era, it is significantly difficult to obtain such a data prior probability distribution. Therefore, this calls for the new algorithms that can release the dependency of specific source data and data prior probability distribution knowledge.

In this paper, we propose a novel framework, called AutoTL (Automatic Transfer Learning), which enables an automatic knowledge transferring. AutoTL differs itself by utilizing the informative online information to strengthen the short text analysis without the need of specifying the source training data, when the short text is not well labeled and without knowing the priori probability distribution. Specifically, using the latent semantic analysis techniques, it first extracts the semantic related keywords as the seed feature set between the online web (long text) data and the target data. This can be done by employing the online search engine via inputting the tags extracted from the target data to obtain the most relevant web data. It then builds one undirected graph for the online web data where the nodes represent the tags/labels. Within this graph, it further extracts one subgraph which is able to cover all the seed feature set. In addition, an improved Laplacian Eigenmaps is adopted to map the high-dimensional feature representation to a low-dimensional one. Finally, it classifies the target data through one constraint function of minimizing the mutual information between the instance and feature representation.

Our major contributions are summarized as follows:

- We propose AutoTL, an transfer learning algorithm of effective short text analysis. AutoTL is superior to other algorithms, as it automatically identifies the related source data from the rich online information and does not require the system to know the priori probability distribution of the data in advance.
- We provide the techniques to integrate the latent semantic analysis into the short text analysis which facilitates an effective learning.
- We conduct extensive experimental evaluations and experimental result indicate that our proposed technique is effective, efficient and practical.

The reminder of the paper is organized as follows. Section 2 introduces the automatic transfer learning algorithm. In Section3, we provide experimental evaluation. Section4 presents the existing work. In Section 5, we conclude the paper.

2 Automatic Transfer Learning Based on Latent Semantic Analysis

In this section, we introduce the proposed transfer learning framework for short text analysis. We will first define the short text analysis problem, then introduce the solution of constructing the feature representation for the target data based on the latent semantic analysis followed by the introduction of the classifier generation.

2.1 Problem Definition

The target domain or target data is referred to a large amount of short texts data $X = \{X_1, X_2, ..., X_n\}$, where X_i is one short text instance. Among the target domain, the known label space is referred to $L = \{l_1, l_2, ..., l_m\}$ related to X. In the short text analysis, the label space is normally very small and not sufficient to conduct an accurate classifying. In addition, no specific source data are given to the learning, in which case the traditional data mining and machine learning approaches are unable to be applied here. Furthermore, the data priori probability distribution is unknown as well. The problem that we study is given the target domain and limited labels, how to provide an accurate classification over the target domain.

To tackle this problem, in this paper, we propose AutoTL (automatic transfer learning) to increase the short text classification performance by automatically transferring the knowledge obtained from other online long text resources, also called source domain (e.g. the web information or social media). Intuitively, AutoTL adopts the latent semantic analysis approach to dig the semantics to both the target domain and source domain. Based on this semantic meaning, it formalizes the important features and make the connection between these two different type of data. It tries to find the best feature representation in order to keep the text semantics for a good classification. The key techniques are introduced in below.

2.2 Keyword Extraction

As the related source data are not provided, we have to figure out which online resources are most related to the target data first. In order to do so, a set of keywords are extracted from the target domain and are supplied to a search engine to search the related source data. For instance, a simple way is to utilize the top k related web pages as the source data. Therefore, the first step that AutoTL is to extract the representative keywords. It is insufficient to simply use the labels as the keywords, as this would lead the topic distillation. To overcome this, we adopt the mutual information to help for the source data selection. The mutual information captures the degree of mutual information between two objects which is defined as follows:

$$I(P;Q) = \sum_{x \in P} \sum_{y \in Q} p(x,y) log \frac{p(x|y)}{p(x)} = \sum_{x \in P} \sum_{y \in Q} p(x,y) log \frac{p(x,y)}{p(x)p(y)} \tag{1}$$

A bigger mutual information indicates a higher correlation between two objects. Using the mutual information as the measure, the target domain is preprocessed to calculate the target feature seed sets which share the biggest mutual information with the target label space. Specifically, the mutual information is calculated as *I(x,c)*, where I is the feature seed and c is the label. When $I(x_i, c_j) > \epsilon$ where ϵ is the threshold, it indicates that the feature x_i is highly related to c_j. We can choose the x_i as the keywords.

2.3 Feature Weight Calculation

After selecting the source data, the next step is to identify the useful labels/features from the source data, which can be used to strengthen the target data classification. To do so, a naive approach is to calculate the similarity between different set of the features between the target domain and the source domain. According to the similarity among the words, the useful features can be selected. However, this approach treats each word as an individual, which ignores the relations between the text and the semantic relationship hidden in the context keywords. Hence, we utilize the latent semantic analysis approach instead [5]. Semantic analysis shows its superiority on this as it organizes the text into a space semantic structure that keeps the relation between the text and words.

Text matrix is used in the latent semantic analysis. It does not only capture the word frequency in the text and also the capability of distinguishing the texts. Typically, in latent semantic analysis, the feature weight is calculated as the multiplication between the local weight (*LW(i,j)* indicating the weight of word i in text j) and the global weight ($GW(i)$ indicate the weight of word i in all the texts). Particularly, the feature weight can be calculated as follows:

$$W(i,j) = LW(i,j) * GW(i) = log(tf(i,j)+1) * (1 - \sum_j \frac{p_{ij} log(P_{ij})}{logN}) \tag{2}$$

where $P_{ij} = \frac{lf(i,j)}{gf(i)}$, $lf(i,j)$ is the frequency of word i in text j, and $gf(i)$ is the frequency of word i in the all the texts.

This traditional method works well in the context where the target and source domains belong to the same type of data with the same data distribution. Unfortunately, the above method can not be directly applied to our context where the target and source data are completely different data types and most likely have different data distribution. The reason is the traditional method does not consider the difference between the source and target domains which may result in poor classification performance. Therefore, in this paper, we propose a new latent semantic analysis approach to enable an accurate classification by utilizing the word frequency and the entropy.

Word Frequency Weight. The word frequency weight is referred to the frequency of the feature appearing in different labels, which captures the capability of distinguishing the labels using the feature. In other words, if one feature appears frequently in one text, it indicates that this feature play an important role in this text. Meanwhile, if this feature has high frequency in other texts as well, we shall reduce its weight as it can not distinguish the texts much. Assume the labels we obtained from the source data represent the categories based on the keywords. So the word frequency weight can be calculated as below:

$$\begin{aligned} FW(C_i,j) &= logcf(C_i,j) \times \frac{1}{log(\sum_{k \neq i}^{cf(C_k,j)})} \\ &= log\frac{\sum_{j,t=1}^{m} tf(t,j)}{m} \times \frac{n(c-1)}{log(\sum_{k \neq i}^{c-1} \sum_{s=1}^{n} tf(s,j))} \end{aligned} \tag{3}$$

where, $cf(C_i,j)$ is the frequency of feature j appearing in category C_i, $\sum_{k \neq i} cf(C_k,j)$ is frequency of feature j appearing in other categories, $\sum_{j,t=1}^{m} tf(t,j)$ is the frequency of feature j appearing in all the documents belonging to the category C_i, m is the number of documents in C_i and $c-1$ is the number of labels of the documents.

Entropy Weight. In this paper, we use the entropy to represent the weight of the classification labels which is defined as $CW(c|i)$. The entropy weight represents degree of the importance of one feature to the classification labels. The entropy (*H(X)*) is the degree of the uncertainty to one signal X, which is calculated as:

$$H(X) = -\sum p(x_i) logp(x_i) \tag{4}$$

The conditional entropy ($H(X|Y)$) is the uncertainty degree of X when Y is confirmed, which is calculated as follows:

$$H(X|Y) = -\sum p(x_i|Y)logp(x_i|Y) = -\sum p(x_i, Y)log(x_i, Y) \tag{5}$$

Hence, the entropy weight can be calculated as the certainty degree of X when Y is confirmed, such as:

$$CW(C_i|j) = H(C_i) - H(C_i|j) \tag{6}$$

Normally, $H(C_i)$ is hard to calculate and satisfies the following condition: $H(C_i|j) \leq H(C_i) \leq log(c)$. So when the source documents contain similar length, $H(C_i)$ is close to $log(c)$. Thus, the entropy weight can be adjusted as follows:

$$\begin{aligned} CW(C_i|j) = H(C_i) - H(C_i|j) &= log(c) + \sum p(t,j)log(t,j) \\ &= log(c) + \sum \frac{tf(t,j)}{gf(j)} log(\frac{tf(t,j)}{gf(j)}) \end{aligned} \tag{7}$$

To this end, the weight in our proposed approach is calculated as follows:

$$W(i) = FW(C_i, j) \times CW(C_i|j) \tag{8}$$

Different to the traditional latent semantic analysis that builds the feature-document weight matrix, AutoTL builds the feature-classification labels weight matrix. In the matrix, the weight w_{ij} in the i^{th} row and j^{th} column represents the correlation between the feature and the classification labels. Assume the matrix obtained from the documents is M. After the SVD decomposition, we can get matrix M_k. In addition, via the feature similarity $M_k M_k^T$, we can obtain the features that are not labeled in the target domain, but highly related to the classification. So the best features are chosen as the feature seed set.

2.4 New Feature Space Construction

Consider that the features may contain many relations in a real life. In order to improve the classification quality, we try to capture the relations among these features. To do so, the approach we propose is to construct the source domain labels as one undirected graph, where the nodes capture the labels and the edges are their relations. To build the relation from the feature seed sets, we try to extract the subgraph that contains all feature seed sets from it. This eventually build the connections between the labels in the source domain and target domain.

Since the label graph is normally high-dimensional, we adopt the the Laplacian Eigenmaps algorithm [6] to map all the nodes in the sub-graph into one low-dimensional space. This effectively alleviates many problems (e.g. data over fitting, low efficiency and so on), which are caused by the high-dimension. The Laplacian Eigenmaps algorithm assumes that if the points are close in the high-dimensional space, the distances between them should be short when embedded into a low-dimensional space. As in the algorithm, it does not consider the category information of the samples when calculate the neighbor distance. No matter the point inside or outside the category, it gives the points same weight if the distances are the same. This, however, is not preferred for the target domain containing both labeled data and unlabeled data. In the paper, we improve the Laplacian Eigenmaps algorithm, using different methods to calculate the weight of labeled data and unlabeled data. Intuitively, we make point distance inside the category be less than the distance whose points are outside the category.

To construct a relative neighborhood graph, we use the unsupervised learning approach (e.g. Euclidean distance) to calculate the distance between the unlabeled data. Meanwhile, we use the supervised learning to calculate the distance between the labeled data, which is provided as follows:

$$D(x_i, x_j) = \begin{cases} \sqrt{1 - \exp(-d^2(x_i, x_j)/\beta} & c_i = c_j \\ \sqrt{\exp(d^2(x_i, x_j))/\beta} & c_i \neq c_j \end{cases} \tag{9}$$

where, c_i and c_j are categories of the samples x_i and x_j, $d(x_i, x_j)$ is the Euclidean distance between x_i and x_j. Parameter β can prevent $D(x_i, x_j)$ too large when $d(x_i, x_j)$ become larger which can effectively control the noises. If the distance between sample points x_i and x_j is smaller than the threshold ε, the two points are neighbor points.

Furthermore, the weight matrix W can be calculated, where if x_i and x_j are neighbor points, $W_{ij} = 1$, otherwise, $W_{ij} = 0$. The Laplacian generalized eigenvectors can be simply calculated by solving the following problem:

$$\begin{cases} \min \sum_{i,j} \|Y_i - Y_j\| w_{ij} \\ s.t. \quad Y^T D Y = I \end{cases} \tag{10}$$

where, D is a diagonal matrix. With the improved Laplacian Eigenmaps algorithm, we can map each high-dimensional node into a low-dimensional space. To this end, the data can get a new feature representation.

2.5 The Target Domain Classification

After getting the new feature representations of the target data, we can classify the target domain using the mutual information as what has been discussed in section 2.2. This can be done based on the existing classifier, such as SVM classifier. For the space limitation, we omit the details here. To better appreciate the framework, Figure 1 provides the main steps of the entire AutoTL framework.

3 Experiments

This section provides the experimental evaluation. All the experiments are conducted on a machine with Dual Core E5300 and 1.86GHz CPU and 16GB memory running in Windows 7. In order to evaluate the efficiency of the AutoTL, we

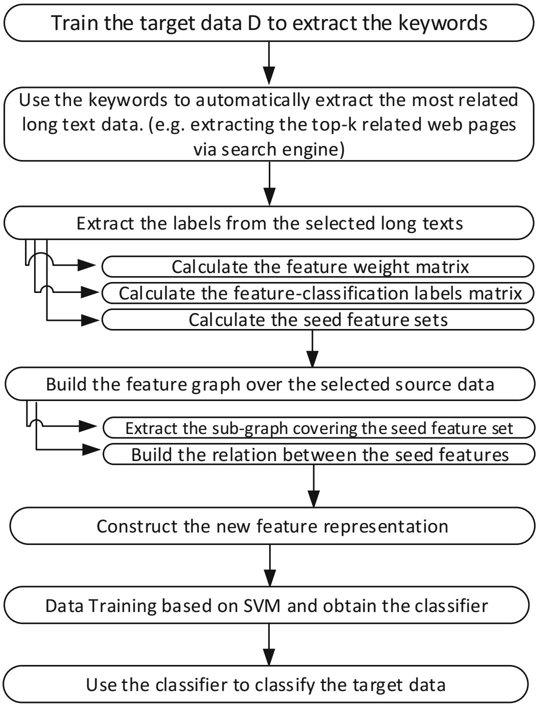


Fig. 1. AutoTL Framework.

use 20Newsgroups, SRAA (Simulated Real Auto Aviation) and Reuter-21578 as three main document classification tasks in the experiments. The 20Newsgroups includes 18774 news reports, which consists of 7 big categories, 20 small categories and 61188 vocabularies. SRAA includes more than 700,00 UseNet articles, which consists of 2 big categories and 4 small categories. Reuter-21578 includes 22 files, which consists of 5 categories. From these three tasks, we extract 7 different datasets/categories including: comp, sci, talk, rec, aviation, auto and topics. Meanwhile, we compare our framework with three classical algorithms: TrAdaboost[7], DATAT[8], TrSVM[9].

3.1 Analysis of Experimental Results

There are two important factors that would impact the algorithm performance: the mutual information threshold ϵ of determining whether two features are correlated and the number of web pages selected as the source data. Hence, we first run two sets of experiments to study how these two factors impact the performance and then figure out the right one as the default setting in the following experiments.

Impact of Mutual Information Threshold. First, we study how the mutual information threshold impacts the performance. This set of experiments is conducted using four different datasets: comp, talk, aviation and topics. Figure 2 presents the result of AutoTL when we vary the threshold from 0.2 to 0.8.

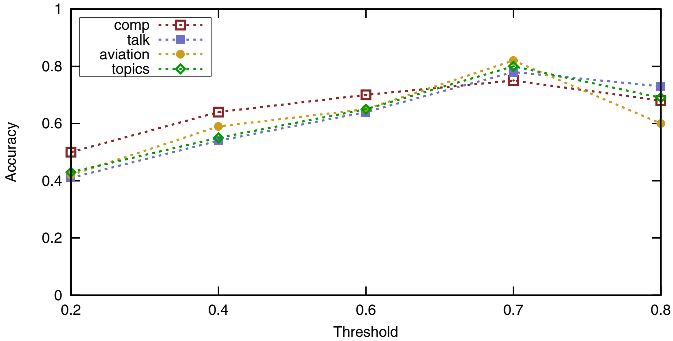


Fig. 2. Impact of the mutual information threshold.

From the result, we obtain two insights. First, selecting different mutual information threshold to determine whether two features are correlated impacts the performance. Second, AutoTL achieves a better performance when the threshold is set around 0.7. The performance decreases when the threshold is set too small or too large. For example, the performance of point 0.2 and 0.8 is worse than that of 0.7. This is within expectation, as a small or large threshold would either result in too many unrelated features or too less correlated features which all lead a worse learning result.

Impact of the Number of Web Pages. Next, we study how the number of web pages selected as the source data impacts the AutoTL performance. This set of experiments is conducted using four different datasets: sci, rec, auto and topics. Figure 3 provides the accuracy of AutoTL, when we vary the number of selected web pages as the source data from 5 to 20.

From the result, we observe that AutoTL performs better when the number is around 10. When the number of selected web pages is too small or too large, the performance decreases. This is reasonable. Since when the number of selected web pages is too small, the source data can not get enough feature information in the training which may decrease the performance. On the other hand, when the number of selected web pages is too large, the source data may involve more noises that may also decreases the performance. So according to the source data quality, choosing the right number of selected pages does impact the performance.

Based on these study, in the following experiments, we use 0.7 as the mutual information threshold and 10 web pages as the source data for AutoTL by default.

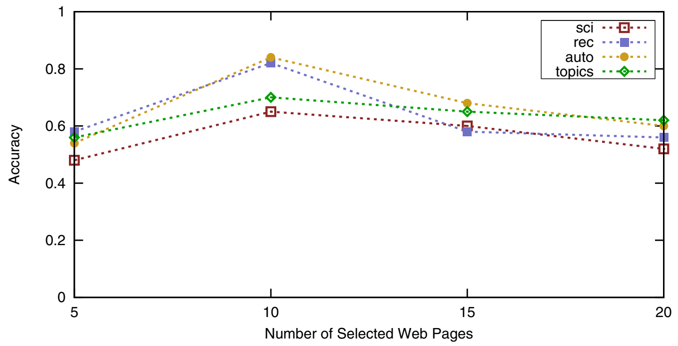


Fig. 3. Impact of the number of web pages.

Performance Comparison on Different Datasets. Furthermore, we compare the performance among four different algorithms: AutoTL, TrAdaBoost, DRDAT and TrSVM. The experiment is conducted based on seven datasets: comp, sci, talk, rec, aviation, auto and topics. Figure 4 shows the comparison results over different datasets. From the result, we can see that different algorithms perform different over different datasets. AutoTL outperforms other algorithms among the different datasets. This confirms the efficiency and effectiveness of AutoTL.

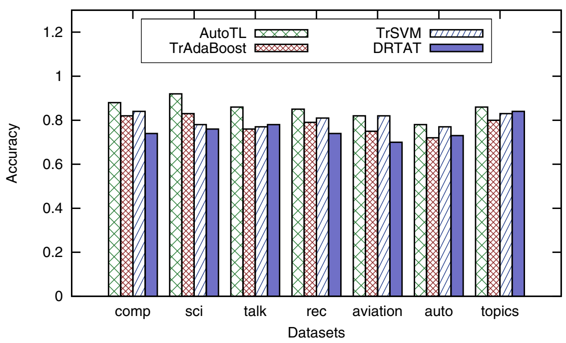
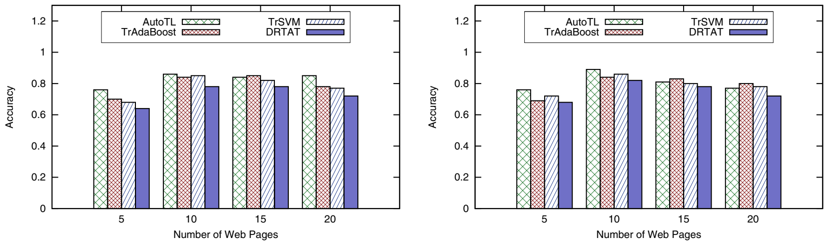


Fig. 4. Performance comparison over different datasets.

Performance Comparison Under Different Amount of Source Data. Finally, as an complete study, we also study the performance comparison among the four algorithms when we choose different number of web pages as the source data. Figure 5 (a) and (b) provides the comparison over comp and sci datasets while varying the number of selected web pages from 5 to 20 respectively. From the results, we can see that the number of selected web pages impact the algorithm performance. We can further obtain another two insights. The first one is all the algorithms follow the pattern that the algorithm performance would decrease when the number is too small or too large. The second one is when the number is set around 10, the algorithms achieve a better performance. The third one is that when in some of other settings, AutoTL may perform a little bit worse than other algorithms. For example, in Figure 5 (a), TrAdaBoost performs a little bit better than AutoTL. This could be because when the number is large, AutoTL affects by the noise more than TrAdaBoost.



(a) comp dataset. (b) sci dataset.

Fig. 5. Performance comparison over comp and sci datasets, when the number of web pages changes.

4 Related Work

Transfer learning has been widely used in long text analysis domain. To name a few, Dai et al. [7] proposed TrAdaboost, which improved the boosting technology in order to create an automatic weight adjustment mechanism. It filters out most of the data similar to the target areas from the source field so that it can enrich the training data to improve the accuracy of the classifier. Mei et al. [14] proposed WTLME which is based on maximum entropy model, using instance weighted technology. The algorithm transfers model parameters studied from the original field to the target domain and reduces the time of re-collection. Hong et al. [9] proposed TrSVM which requires weak similarity. All of these algorithms perform well when the source data and target data are in a very similar domain.

Dai et al. [10] proposed a CoCC algorithm, in which the co-occurrence of words in the source domain and the target domain were used as a bridge. The tag structures of the source field and the target domain were collaboratively clustering at the same time. By minimizing the mutual information between words and samples, it can achieve the goal that transfer the tag structure of the source domain to the target domain. Xue et al. [11] proposed a TPLSA algorithm which tried to bridge the relations between two related domains. Long et al. [12] proposed a GTL algorithm, which extracted the potential common themes between source and target domains and optimize maximum likelihood function to maintain the geometric structure of the documents. These algorithms are mainly used in the same language of the text files. Ling et al. [13] proposed an algorithm to handle the text analysis when they were in different languages by using the information bottleneck model. However, all these above mentioned algorithms are developed for analyzing the long text data.

Recently, some research has been conducted on the short text analysis by transferring the knowledge from the long texts. For example, Jin et al. [4] proposed a DLDA model, which extracts two sets of topics from the source and target domains and uses a binary switch variable to control the forming process of the documents. However, the algorithm requires the source data and the priori probability distribution to be known in advance. AutoTL differs itself from this algorithm by an automatic source data selection and no priori probability distribution requirement.

5 Conclusions

Transfer learning is a technique that finds useful knowledge and skills in the previous tasks and applies them to new tasks or domains. In this paper, we proposed AutoTL, an automatic transfer learning framework to analyze the short text data by utilizing the long text knowledge such as web data. AutoTL shows its superiority than other algorithms from different perspectives. First, it does not enforce the user to provide a specific source data for training, but conducts an automatic source data selection. Second, no priori probability distribution is required in advance. Third, AutoTL integrates the rich online information and latent semantic analysis in the short text learning task, which highly increases the learning accuracy. Extensive experimental evaluation indicates that AutoTL is practical, efficient and effective.

Acknowledgments. The work is funded by the Heilongjiang Scientific Research Foundation for Returned Scholars(Grant No.LC2015025), supported by Fundamental Research Funds for Central Universities(Grant No.HEUCFD1508 and HEUCF100602) and a special study of technological innovation fund of Harbin(Grant No.2013RFQXJ113 and 2013RFQXJ117). This work is also partially supported by China NSF(Grant No. 61402126, 61370083) and Postdoctoral Scientific Research Foundation of Heilongjiang Province.

References

1. Pan, S.J., Qiang, Y.: A Survey on Transfer Learning. IEEE Transactions on Knowledge and Data Engineering **22**(10), 1345–1359 (2010)
2. Yang, Q.: An introduction to transfer learning. In: Tang, C., Ling, C.X., Zhou, X., Cercone, N.J., Li, X. (eds.) ADMA 2008. LNCS (LNAI), vol. 5139, pp. 1–1. Springer, Heidelberg (2008)
3. Lu, Z., Zhu, Y., Pan, S.J., et al.: Source free transfer learning for text classification. In: Proceedings of the 28th AAAI Conference on Artificial Intelligence, pp. 122–128. AAAI Press, Québec (2014)

4. Jin, O., Liu, N.N., et al.: Transferring topical knowledge from auxiliary long texts for short text clustering. In: Proceedings of the 20th ACM Conference on Information and Knowledge Management, pp. 775–784. ACM Press, Glasgow (2011)
5. Dumais, S.T., Furnas, G.W., et al.: Using latent semantic analysis to improve information retrieval. In: Proceedings of the ACM Conference on Human Factors in Computing Systems, pp. 281–285. ACM Press, Washington D.C. (1988)
6. Belkin, M., Niyogi, P.: Laplacian eigenmaps for dimensionality reduction and data representation. Neural Computation **10**(5), 1373–1396 (2003)
7. Dai, W., Yang, Q., Xue, G., et al.: Boosting for transfer learning. In: Proceedings of The 24th Annual International Conference on Machine Learning, Corvallis, Oregon, USA, pp. 193–200 (2007)
8. Liu, W., Zhang, H.: Ensemble transfer learning algorithm based on dymaica dataset regroup. Computer Egineering and Applications **46**(12), 126–128 (2010)
9. Jiaming, H., Jian, Y., Yun, H., Yubao, Y., Jiahai, W.: TrSVM: A transfer learning method based on the similarity of domains. Computer Research and Development **48**(10), 1823–1830 (2011)
10. Dai, W., Xue, G.-R., et al.: Co-clustering based classification for out-of-domain documents. In: Proceedings of the Thirteenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Jose, California, USA, pp. 210–219 (2007)
11. Xue, G., Dai, W., et al.: Topic-bridged PLSA for cross-domain text classification. In: Proceedings of the 31st Annual International ACM SIGIR Conference, pp. 627–634. ACM Press, Singapore (2008)
12. Long, M., Wang, J., Ding, G., Shen, D., Yang, Q.: Transfer learning with graph co-regularization. In: Proceedings of the 26th AAAI Conference on Artificial Intelligence. AAAI Press, Toronto, Ontario (2012)
13. Ling, X., Dai, W., et al.: Can Chinese web pages be classified with english data source. In: Proceedings of the 17th International Conference on World Wide Web 2008, Beijing, China, pp. 969–978 (2008)
14. Mei, C., Zhang, Y., Xuegang, H., Li, P.: Transfer learning algorithms based on maximum entropy model. Computer Research and Development **48**(9), 1722–1728 (2011)