Yang and Zhang *EURASIP Journal on Wireless Communications and Networking* (2017) 2017:42
DOI 10.1186/s13638-017-0815-5

EURASIP Journal on Wireless Communications and Networking

RESEARCH Open Access

Automatic transfer learning for short text mining

Lei Yang and Jianpei Zhang*

Abstract
As a new emerging technique, transfer learning enjoys the advantage of integrating the well-learnt knowledge from another related work to facilitate an improved learning result of one task. Most of the existing transfer learning methods are designed for long texts and short texts. However, the latter one distinguishes from the former one in terms of its sparse nature, noise words, syntactical structure, and colloquial terminologies used. A transfer learning algorithm called automatic transfer learning (AutoTL) is proposed for short text mining. By transferring knowledge automatically learnt from the online information, the proposed method enables training data to be selected automatically. Furthermore, it does not make any a priori assumption about probability distribution. Our experimental results on 20Newsgroups, Simulated Real Auto Aviation, and Reuter-21578 validate the higher performance of the proposed AutoTL over several state-of-of-the-art methods.

Keywords: Transfer learning, Short text mining, Latent semantic analysis

1 Introduction

In the big data era, with the ever increasing complexity of machine learning models such as deep learning, the demand for large amounts of labeled data is growing at an unprecedented scale. The traditional machine learning approaches require not only the training data but also test data to be under the same feature space and the same distribution. The transfer learning, in contrast, allows the domains, tasks, and distribution used in training and testing to be different. It emerges as a new learning technique facilitating an improved learning result of one task by integrating the well-learnt knowledge from another related task. 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 which are 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 [1–6].

Much research has been devoted into the transfer learning in the domain of analyzing long texts. To name a few, Lu et al. [7] proposed source free transfer learning to transfer knowledge from long texts to the long and Jin et al. [8] proposed latent dirichlet allocation to analyze two sets of topics on short and long texts. Thanks to the advances of the Internet, more and more applications regarding blog-sphere and social network applications come into being, such as Twitter, microblog, and online advertising. Such applications result in two features that differ from traditional applications. First, data generated from these applications consist of a lot of short texts, which contains rich useful information. Second, the data are updated with a dramatic speed, in terms of data size and data distribution. These significant features eventually challenge the traditional data mining and machine learning approaches on the one hand, as the assumptions made do not hold in new applications any more. On the other hand, the existing transfer learning algorithms tailored for long text analysis can not be directly applied to these applications. The long text data analysis aims at analyzing long text data with 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, short texts differ from long texts 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. Thus, it is necessary to develop new transfer learning techniques for

*Correspondence: zhangjianpei@hrbeu.edu.cn
College of Computer Science and Technology, Harbin Engineering University, Nantong Street No. 145, 150001 Harbin, China

short text analysis. Given the fact that the results learnt from the long text analysis are enriched, one promising approach is to transfer the long text knowledge into the short text analysis. Several algorithms have been proposed under such a methodology [7, 8]. We can see that 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 especially in such a big data era. In addition, the prior probability distribution is required, which is significantly difficult to be obtained.

In this paper, we propose a novel algorithm, called AutoTL (automatic transfer learning). The AutoTL differs itself from the traditional methods by utilizing online information to strengthen the short text analysis without the need of specifying the source training data. It is especially fit for the short text which is not well labeled and without knowing the priori probability distribution. Specifically, using the latent semantic analysis techniques, the AutoTL 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 to get the most relevant web data. The AutoTL then builds one undirected graph for the online web data those nodes represent the tags/labels. With such a graph in hand, the AutoTL further extracts a subgraph covering the whole seed feature set. In addition, an improved Laplacian eigenmaps is adopted to map the high-dimensional feature representation into a low-dimensional one. Finally, the classification has been done through minimizing the mutual information between the instance and feature representation. Our major contributions are as follows:

- We propose the AutoTL, a transfer learning algorithm of effective short text mining. The AutoTL is superior to other algorithms in terms of automatically identifying the related source data from the rich online information with no requirements of the priori probability distribution and it integrates the latent semantic analysis into the short text analysis which facilitates an effective learning.
- We conduct extensive experimental evaluations and the experimental results indicate that our proposed technique is effective and practical.
- We find that the AutoTL may be applied to short text classification, recommender system, and short text clustering.

The reminder of the paper is organized as follows. First, we present the related work. Then, we describe the details of the automatic transfer learning algorithm and present experimental evaluations. Finally, we conclude the paper.

2 Related work

Big data problem is a big challenge in today's world. How can we avoid of learning with only a few example? Machines should learn how to learn in this special era [9–13]. Transfer learning gets the opportunity to prosper. It has been widely used in long text analysis domain. To name a few, Dai et al. [14] 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. [15] 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. [16] proposed TrSVM which requires weak similarity. There are other researchers exploring this field [17–23]. All of these algorithms perform well when the source data and target data are in a very similar domain.

Dai et al. [24] 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. [25] proposed a TPLSA algorithm which tried to bridge the relations between two related domains. Long et al. [26] 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. [27] proposed an algorithm to handle the text analysis when they were in different languages by using the information bottleneck model. However, all these abovementioned 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. [8] 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. The AutoTL differs itself from the algorithm by an automatic source data selection, and no requirements of priori probability distribution.

3 Automatic transfer learning algorithm based on latent semantic analysis

In this section, we present the details of the proposed the AutoTL. We will first define the short text mining problem, then introduce the solution to the feature representation of the target data based on the latent semantic analysis which is followed by the introduction of the classifier generation.

3.1 Problem statement

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 the ith 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 classification. Moreover, no specific source data are given to the learning, to which the traditional data mining and machine learning approaches are unable to be applied. Furthermore, the data priori probability distribution is unknown as well. The problem studied here is given the target domain and limited labels, how to provide an accurate classification over the target domain.

To fit this problem, in this paper, we propose the AutoTL. It automatically transfers the knowledge obtained from other online long text resources, also called source domain (e.g., the web information or social media). The AutoTL adopts the latent semantic analysis to dig the semantics of both the target domain and the source domain. Based on this semantic meaning, it formalizes the important features and links these two different types of data together. It tries to find the best feature representation in order to keep the text semantics for a good classification. Thus, the key techniques of the proposed AutoTL includes keyword extraction, feature weight calculation, new feature space construction, and target domain classification.

3.2 Keyword extraction

As the related source data are not provided, we have to figure out which online resources are the most related to the target data first. In order to do so, a set of keywords are extracted from the target domain and then supplied to a search engine to get the related source data. Therefore, the first step of the AutoTL is to extract the most representative keywords. It is insufficient to simply use the labels as the keywords, as this would lead to the topic distillation. In contrast, we adopt the mutual information to select the source data. The correlation of mutual information between two objects is given by:

$$\begin{aligned} I(P;Q) &= \sum_{x \in P} \sum_{y \in Q} p(x,y) \log \frac{p(x|y)}{p(x)} \\ &= \sum_{x \in P} \sum_{y \in Q} p(x,y) \log \frac{p(x,y)}{p(x)p(y)} \end{aligned} \quad (1)$$

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. $I(x_i, c_j) > \epsilon$(here ϵ is the threshold) indicates that the feature x_i is highly related to c_j. In this case, x_i can be chosen as the keywords.

3.3 Feature weight calculation

After selecting the source data, the next step of the AutoTL is to identify the useful labels/features from the source data, which can be used to strengthen the target data classification. A naive approach is adopted to calculate the similarity between different sets of features from the target domain and the source domain respectively. According to the similarity among the words, the useful features can be selected. However, such an approach treats each word individually, ignoring the relations between the text and the semantics that are hidden in the context keywords. Hence, we utilize the latent semantic analysis approach instead [28, 29]. Semantic analysis shows its superiority on such a task as it organizes the text into a space semantic structure that keeps the relationships between the text and the words.

Text matrix is used in the latent semantic analysis. It not only captures the word frequency in the text but distinguishes the texts. Typically, in latent semantic analysis, the feature weights are calculated as the multiplication of 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 the whole texts). Particularly, the feature weight $W(i,j)$ is given by:

$$\begin{aligned} W(i,j) &= \mathrm{LW}(i,j) * \mathrm{GW}(i) \\ &= \log(tf(i,j)+1) * (1 - \sum_j \frac{p_{ij} \log(P_{ij})}{\log N}) \end{aligned} \quad (2)$$

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

This traditional method works well in the context where the target and the source domains share the same data and distribution. Unfortunately, it cannot be directly applied to our context where the target and source data are completely different in terms of the data type as well as the data distribution. The reason is that traditional methods do not consider the difference between the source and the target domains resulting in poor classification. 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.

3.3.1 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 the feature plays an important role in the text. Meanwhile, if this feature has high frequency in other texts as well, its weight should be degraded due to the less separative capacity. 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:

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

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 the 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.

3.3.2 Entropy weight

In this paper, we use the entropy to represent the weight of the classification labels which is defined as $\mathrm{CW}(c|i)$. The entropy weight represents the 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:

$$H(X) = -\sum p(x_i)\log p(x_i) \quad (4)$$

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

$$\begin{aligned}H(X|Y) &= -\sum p(x_i|Y)\log p(x_i|Y) \\ &= -\sum p(x_i,Y)\log(x_i,Y)\end{aligned} \quad (5)$$

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

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

Normally, $H(C_i)$ is hard to calculate and should satisfy 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:

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

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

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

Different from the traditional latent semantic analysis that builds the feature-document weight matrix, the AutoTL builds the feature-classification labels weight matrix. In the matrix, the weight w_{ij} 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.

3.4 New feature space construction

Considering that the features may contain many relations in real life, we try to capture the relations among these features to improve the classification quality. The approach we proposed is to construct the source domain labels as an undirected graph, whose nodes denote the labels and its edges are the relations. To build the relation from the feature seed sets, we extract a subgraph that contains all feature seed sets from the undirected graph. This eventually build the connections between the labels in the source domain and the target domain.

Since the label graph is normally high-dimensional, we adopt the the Laplacian eigenmaps algorithm [30] to map all nodes in the subgraph into a low-dimensional space. This effectively alleviates the problems such as data over fitting and low efficiency, caused by the high dimension. The Laplacian eigenmaps 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. The algorithm does not consider the category information of the samples when calculating the neighbor distance. Thus, no matter the point inside or outside the category, it gives the points with the same distance the same weight. 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 by using different methods to calculate the weight of the labeled data and unlabeled data. Intuitively, we make point distance inside the category be less with distance than those points 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 for 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, respectively, and $d(x_i, x_j)$ is the Euclidean distance between x_i and x_j. Parameter β can prevent $D(x_i, x_j)$ from becoming 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:

$$\min \sum_{i,j} \|Y_i - Y_j\| w_{ij} \qquad \text{s.t.} \quad Y^T DY = I \tag{10}$$

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.

3.5 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 3.2. This can be done based on the existing classifier, such as the SVM classifier. To better appreciate the framework, Fig. 1 provides the main steps of the entire AutoTL framework. The detailed Algorithm AutoTL is as follows:

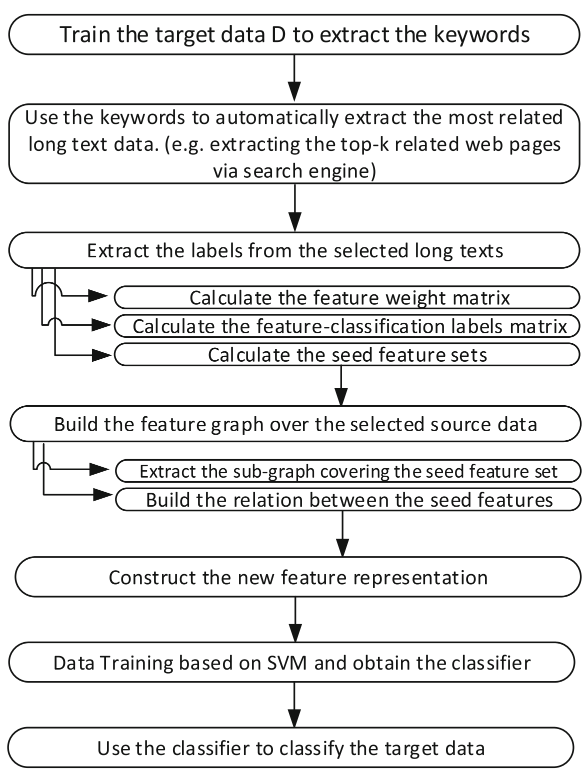

Fig. 1 AutoTL framework

Algorithm AutoTL

Input: $T = T^l \cup T^u$
(T^l are labeled data and T^u are unlabeled data).
Output: the classification results of the target data according to I.

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

1.**Initialize** $k, \lambda, \beta, \varepsilon$;
(neighbor value k , feature seed threshold λ , parameter β, feature threshold ε)
2.For $C = 1, \ldots, N$ (C is the labels and N is the number of the labels id)
3. Input the target labels to a search engine.
4. Extract the first n pages of data as the data that the most associated with the target areas.
5. Get the feature seed sets according to $\tilde{M}$, k and λ.

$$\tilde{M} = U\tilde{\Sigma}V^T \approx U\Sigma V^T = M$$

6. Build the undirected graph of the social media.
7. Extract a subgraph that contains all these seed feature sets.
8. Filter the features of the target representation according to $\mathrm{tr}(Y^T LY)$, $\beta and \varepsilon$

$$\begin{cases} \arg\min \mathrm{tr}(Y^T LY) \\ s.t. \quad Y^T DY = I \end{cases}$$

---

4 Experimental results and analysis

This section provides the experimental evaluation. All experiments are conducted on a machine with Dual Core E5300, 1.86-GHz CPU, and 16-GB memory running in Windows 7. In order to evaluate the effectiveness of the AutoTL, we use 20Newsgroups, SRAA (Simulated Real Auto Aviation), and Reuter-21578 as three main document classification tasks in the experiments. The 20Newsgroups includes 18,774 news reports, which consists of 7 big categories, 20 small categories, and 61,188 vocabularies. SRAA includes more than 70,000 UseNet articles, which consists of 2 big categories and 4 small categories. Reuter-21578 includes 22 files, which consists of 5 categories. From the above 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 [14], TrSVM [16], and DRTAT [31].

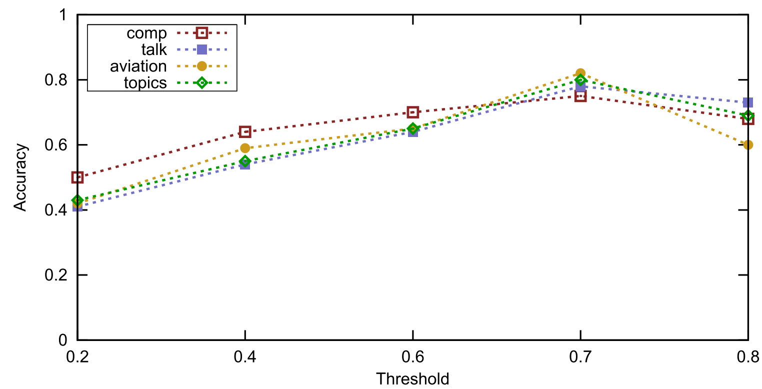

Fig. 2 Impact of the mutual information threshold

4.1 Analysis of experimental results

There are two important factors that would impact the performance of the AutoTL: 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.

4.1.1 Impact of mutual information threshold

The experiments are conducted using four different datasets: comp, talk, aviation, and topics. Figure 2 presents the results of the AutoTL with the threshold from 0.2 to 0.8.

From these results, we obtain two insights. First, selecting different mutual information threshold to determine whether two features are correlated impact the performance. Second, AutoTL achieves a better performance when the threshold is set around 0.7, while 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 to a worse learning result.

4.1.2 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. These results indicate 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, which is in accordance to the fact that 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.

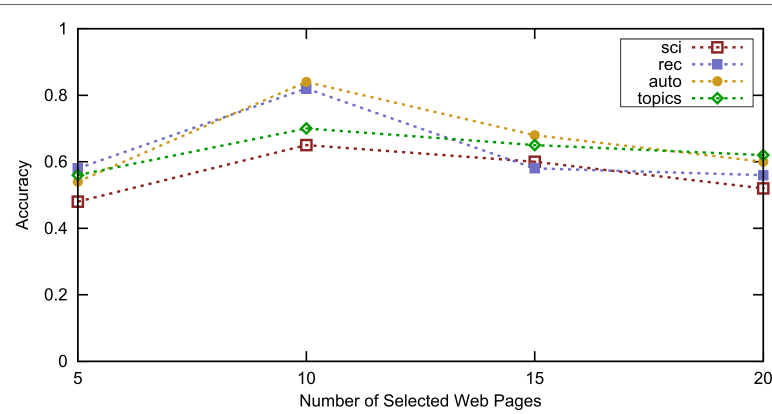

Fig. 3 Impact of the number of web pages

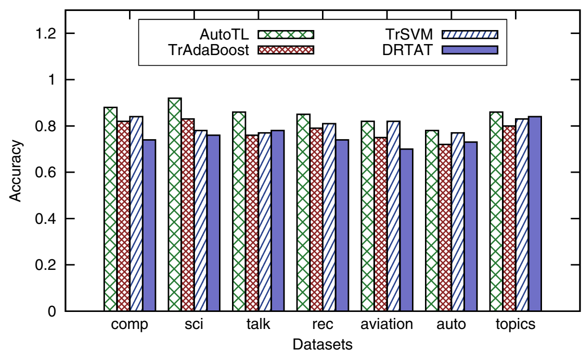

Fig. 4 Performance comparison over different datasets

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

4.1.3 Performance comparison on different datasets

Furthermore, we compare the performance among four different algorithms: AutoTL, TrAdaBoost, DRTAT 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 seven datasets. From the result, we can see that four algorithms perform different over the seven datasets. AutoTL outperforms other algorithms among the different datasets. This validates the effectiveness of AutoTL.

4.1.4 Performance comparison under different amount of source data

Finally, as a complete study, we also compare the performances among the four algorithms when we choose different number of web pages as the source data. Figure 5a, 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 the other algorithms. For example, in Fig. 5a, 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.

5 Conclusions

Transfer learning is a technique that leverages useful knowledge and skills in the previous tasks and applies them to new tasks or domains. In this paper, we proposed the AutoTL, an automatic transfer learning framework to analyze the short text data by utilizing the long text knowledge such as web data. The AutoTL shows its superiority over other algorithms introducing the latent semantic analysis. It does not enforce users to provide a specific source data for training but conducts an automatic source data selection. And at the same time, no priori probability distribution is required. And the AutoTL integrates rich online information and latent semantic analysis in short text learning tasks, which highly increases the learning accuracy. Extensive experimental evaluations indicate that the AutoTL is practical and effective. It may be applied to the short text classification, recommender system, and short text clustering. Furthermore, short text generally has a strong relationship with the context, accessory link, picture, video, and so on. We may address these to enhance their works in the future.

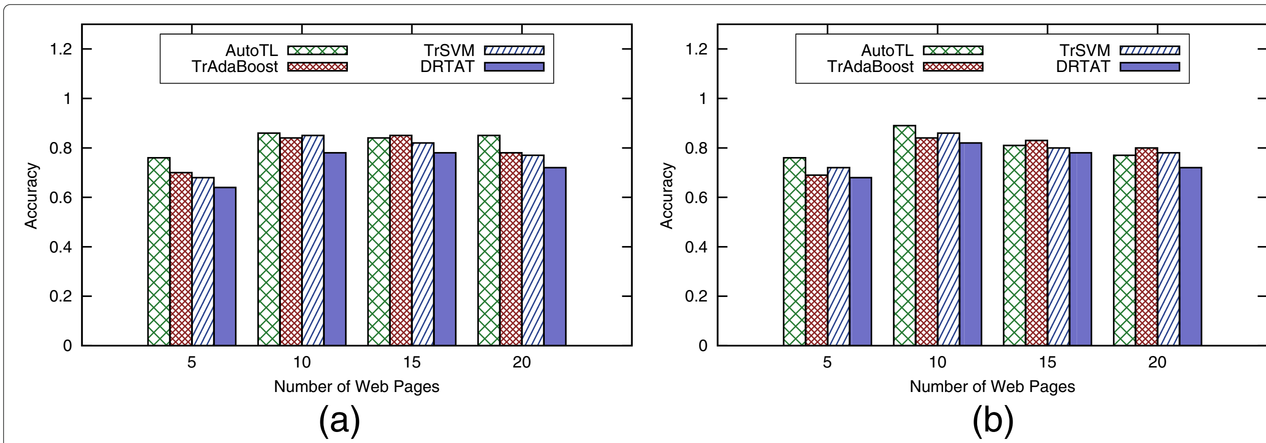

Fig. 5 Performance comparison over comp and sci datasets. **a** comp dataset. **b** sci dataset

Acknowledgements
The work is funded by the National Natural Science Funds of China (Grant No. 61503073), Heilongjiang Scientific Research Foundation for Returned Scholars (Grant No. LC2015025), Fundamental Research Funds for Central Universities (Grant No. HEUCFD1508 and HEUCF160602), a special study of technological innovation fund of Harbin (Grant No. 2013RFQXJ113 and 2013RFQXJ117), and Postdoctoral Scientific Research Foundation of Heilongjiang Province.

Authors' contributions
JZ proposes the AutoTL framework and LY extends the work further by proposing the AutoTL Algorithm and find the AutoTL may be applied to short text classification, recommender system and short text clustering. Both authors read and approved the final manuscript.

Competing interests
The authors declare that they have no competing interests.

Received: 6 January 2016 Accepted: 30 January 2017
Published online: 01 March 2017

References

1. SJ Pan, Q Yang, A survey on transfer learning. IEEE Trans. Knowl. Data Eng. **22**(10), 1345–1359 (2010)
2. B Tan, Y Song, E Zhong, Q Yang, in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. Transitive transfer learning (ACM, New York, 2015), pp. 1155–1164
3. Y Zheng, B Jeon, D Xu, Q Wu, H Zhang, Image segmentation by generalized hierarchical fuzzy c-means algorithm. J. Intell. Fuzzy Syst. **28**(2), 961–973 (2015)
4. Y Sun, C Lu, R Bie, J Zhang, Semantic relation computing theory and its application. J. Netw. Comput. Appl. **59**, 219–229 (2016)
5. L Zheng, Y Chu, in *23rd International Conference on Pattern Recognition, ICPR*. Texture classification with discrete fractional fourier transform (IEEE, Cancun, 2016)
6. W Kang, AKH Tung, F Zhao, X Li, in *ICDE*. Interactive hierarchical tag clouds for summarizing spatiotemporal social contents (IEEE, IL, 2014), pp. 868–879
7. Z Lu, Y Zhu, SJ Pan, EW Xiang, Y Wang, Q Yang, in *AAAI*. Source free transfer learning for text classification (AAAI, Québec, 2014), pp. 122–128
8. O Jin, NN Liu, K Zhao, Y Yu, Q Yang, in *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*. Transferring topical knowledge from auxiliary long texts for short text clustering (ACM, Glasgow, 2011), pp. 775–784
9. S Cheng, Z Cai, J Li, X Fang, in *2015 IEEE Conference on Computer Communications*. Drawing dominant dataset from big sensory data in wireless sensor networks (IEEE, Glasgow, 2015), pp. 531–539
10. Y Sun, H Yan, C Lu, R Bie, Z Zhou, Constructing the web of events from raw data in the web of things. Mob. Inf. Syst. **10**(1), 105–125 (2014)
11. Y Sun, AJ Jara, An extensible and active semantic model of information organizing for the internet of things. Pers. Ubiquit. Comput. **18**(8), 1821–1833 (2014)
12. X Wen, L Shao, Y Xue, W Fang, A rapid learning algorithm for vehicle classification. Inf. Sci. **295**, 395–406 (2015)
13. X Wu, Z Cai, X-F Wan, T Hoang, R Goebel, G Lin, Nucleotide composition string selection in hiv-1 subtyping using whole genomes. Bioinformatics. **23**(14), 1744–1752 (2007)
14. W Dai, Q Yang, G-R Xue, Y Yu, in *Proceedings of the 24th International Conference on Machine Learning*. Boosting for transfer learning (ACM, Corvallis, 2007), pp. 193–200
15. M Canhua, Z Yuhong, H Xuegang, L Peipei, Transfer learning algorithms based on maximum entropy model. Comput. Res. Dev. **48**(9), 1722–1728 (2011)
16. H Jia-Ming, Y Jian, H Yun, L Yu-Bao, W Jia-Hai, TrSVM: a transfer learning algorithm using domain similarity. J. Comput. Res. Dev. **48**(10), 1823–1830 (2011)
17. B Gu, VS Sheng, Z Wang, D Ho, S Osman, S Li, Incremental learning for ν-support vector regression. Neural Netw. **67**, 140–150 (2015)
18. B Gu, VS Sheng, KY Tay, W Romano, S Li, Incremental support vector learning for ordinal regression. IEEE Trans. Neural Netw. Learn. Syst. **26**(7), 1403–1416 (2015)
19. T Ma, J Zhou, M Tang, Y Tian, A AL-DHELAAN, M AL-RODHAAN, S Lee, Social network and tag sources based augmenting collaborative recommender system. IEICE Trans. Inf. Syst. **98**(4), 902–910 (2015)
20. Z Cai, MF Ducatez, J Yang, T Zhang, L-P Long, AC Boon, RJ Webby, X-F Wan, Identifying antigenicity-associated sites in highly pathogenic H5N1 influenza virus hemagglutinin by using sparse learning. J. Mol. Biol. **422**(1), 145–155 (2012)
21. Z Cai, M Heydari, G Lin, Clustering binary oligonucleotide fingerprint vectors for DNA clone classification analysis. J. Comb. Optim. **9**(2), 199–211 (2005)
22. Y Shi, M Hasan, Z Cai, G Lin, D Schuurmans, Linear coherent bi-clustering via beam searching and sample set clustering. Discret. Math. Algorithms Appl. **4**(02), 1250023 (2012)
23. Z Cai, R Goebel, MR Salavatipour, G Lin, Selecting dissimilar genes for multi-class classification, an application in cancer subtyping. BMC Bioinforma. **8**(1), 1 (2007)
24. W Dai, G-R Xue, Q Yang, Y Yu, in *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. Co-clustering based classification for out-of-domain documents (ACM, San Jose, 2007), pp. 210–219
25. G-R Xue, W Dai, Q Yang, Y Yu, in *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. Topic-bridged plsa for cross-domain text classification (ACM, Singapore, 2008), pp. 627–634
26. M Long, J Wang, G Ding, D Shen, Q Yang, Transfer learning with graph co-regularization. IEEE Trans. Knowl. Data Eng. **26**(7), 1805–1818 (2014)
27. X Ling, G-R Xue, W Dai, Y Jiang, Q Yang, Y Yu, in *Proceedings of the 17th International Conference on World Wide Web*. Can chinese web pages be classified with english data source? (ACM, Beijing, 2008), pp. 969–978
28. S Dumais, G Furnas, T Landauer, S Deerwester, in *Proceedings of the ACM Conference on Human Factors in Computing Systems*. Using latent semantic analysis to improve information retrieval (ACM, Washington DC, 1988), pp. 281–285
29. Y Sun, R Bie, J Zhang, Measuring semantic-based structural similarity in multi-relational networks. Int. J. Data Warehous. Min. **12**(1), 20–33 (2016)
30. M Belkin, P Niyogi, Laplacian eigenmaps for dimensionality reduction and data representation. Neural Comput. **15**(6), 1373–1396 (2003)
31. L Wei, Z Huaxiang, Ensemble transfer learning algorithm based on dynamic dataset regroup. Comput. Eng. Appl. **46**(12), 126–128 (2010)