Boosting Inductive Transfer for Text Classification using Wikipedia

Somnath Banerjee
Hewlett-Packard Labs, Bangalore, India
somnath.banerjee@hp.com

Abstract

Inductive transfer is applying knowledge learned on one set of tasks to improve the performance of learning a new task. Inductive transfer is being applied in improving the generalization performance on a classification task using the models learned on some related tasks. In this paper, we show a method of making inductive transfer for text classification more effective using Wikipedia. We map the text documents of the different tasks to a feature space created using Wikipedia, thereby providing some background knowledge of the contents of the documents. It has been observed here that when the classifiers are built using the features generated from Wikipedia they become more effective in transferring knowledge. An evaluation on the daily classification task on the Reuters RCV1 corpus shows that our method can significantly improve the performance of inductive transfer. Our method was also able to successfully overcome a major obstacle observed in a recent work on a similar setting.

1. Introduction

In machine learning literature, inductive transfer "refers to the problem of retaining and applying the knowledge learned in one or more tasks to efficiently develop an effective hypothesis for a new task" [1]. A great deal of research on inductive transfer has been done under various names, e.g., learning to learn, life-long learning, transfer learning, multi-task learning, hierarchical Bayes etc. Labeled training data is usually scarce or expensive. Obtaining labeled training data might be cheaper in some related tasks. Sometimes already built models for related tasks are available. The purpose of inductive transfer is to use the knowledge learned on the related tasks to improve the performance on the target task. For example, Wu & Dieterrich [2] showed that the image classification accuracy can be improved when SVMs are trained on a large number of related images but relatively few target images. It has been observed transferring knowledge often helps if the tasks are similar enough. But it can also hinder performance if the tasks are too dissimilar. This later phenomena is known as "negative transfer" [3].

In this paper, we show a method of improving the performance of inductive transfer in the task of text classification using Wikipedia (http://wikipedia.org/). We took a *classifier re-use* [4] model as our base inductive transfer model and then showed how inductive transfer can be made more effective using Wikipedia. In the classifier re-use model, a set of classifiers are built for the related tasks. Knowledge transfer from these classifiers is done by using the outputs of these classifiers on the target task. In our method, the classifiers from which knowledge need to be transferred are trained in a feature space generated from the Wikipedia. The text documents to be classified are mapped to a feature space created from the titles of the Wikipedia articles. For each text document, a set of similar Wikipedia articles are retrieved. The documents are then represented using the terms (words) appearing in the titles of the retrieved Wikipedia articles. We observed that classifiers trained in such feature space are more effective in transferring knowledge than the classifiers trained using the words appearing in the documents.

Wikipedia contains the knowledge about the world. Mapping a document to the similar Wikipedia articles provides some background knowledge about the content of the document. Our hypothesis is background knowledge about the related tasks is helpful in transferring the knowledge learned from one set of tasks to another. Therefore, classifiers trained with the Wikipedia features are more predictive in related but different tasks.

We evaluated our method on the daily classification task (DCT) [5] on the Reuters news corpus. In the news domain, as time progresses new events occur and old events disappear. Therefore, the underlying concepts are not stable but change with time. This problem is known as concept drift. In such setting, classifiers built in the past are not very effective in predicting the class labels of today's (or future's) news articles. As time progresses new classifiers are required to be built by obtaining new training data. Instead of throwing away the previously built classifiers an inductive transfer model can be applied to make use of them and thereby reducing the number of new training data required [5]. Given the popularity of news sites and news feeds, a practical solution to the DCT is of commercial interest.

For the base inductive transfer model we used the same model as proposed by George Forman [5]. This inductive transfer model is a classifier re-use model. Prediction outputs of the previously built classifiers act as additional input features to a new classifier. Using this model it is shown here that the previously built classifiers become more effective in transferring knowledge when they are trained in the feature space generated from Wikipedia. Additionally, a major obstacle observed in [5] was that the ground truth labels of the past *test* cases had to be obtained to make the past classifiers predictive on today's articles. Our method was able to successfully overcome this obstacle.

The remainder of this paper is organized as follows. Section 2 gives an overview of the DCT. Section 3 discusses how inductive transfer is applied for the DCT. In section 4, we discuss the details of the feature generation using Wikipedia. Section 5 outlines the experimental settings and the obtained results. Section 6 presents the related works and Section 7 concludes the paper and give some directions for future research.

2. Daily classification task (DCT)

In the daily classification task (DCT), a series of news articles are received over a period of time. Every day a limited number of random samples from the many different articles are provided as labeled training data. The task is to classify the remaining articles of that day. The performance is measured by taking the average of the classification accuracy over all days.

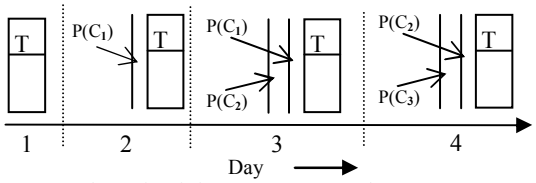
For example, say everyday the system receives N news articles. Out of these N articles, ground truth labels of T random articles are available for training. The job is to classify the remaining (N-T) articles of that day. Performance will be reported by taking the average performance over 365 days (say).

Due to ever changing events in the news domain, feature set defining a particular class keeps changing. Therefore, the predictive accuracy of a classifier built on a particular day decays over time. The strawman approach to solve the DCT problem is to build a classifier everyday using only the T training cases available that day. Then use this classifier to predict the class labels of the remaining (N-T) cases of that day. But we should be able to leverage the models learned in the past. Next we describe an inductive transfer model that can leverage the classifiers built in the previous days to build a classifier for today.

3. Inductive transfer model

In our inductive transfer model, the outputs of the past classifiers are used as additional input features to a new classifier. Like the strawman algorithm, everyday a new classifier is trained using the T available training cases. Inductive transfer is done by adding P additional binary features (positive/negative) to the cases of today. These P features are the predictions of the P previous days' classifiers on the today's cases. Here the value of P determines the number of previous days' classifiers we can use. By varying the value of P in our experiment we will see the impact of P. Note that these P prediction features are used in addition to the bag of words and Wikipedia features of the articles (details in the Section 5). Also, these features are added to the today's training as well as test cases.

An example of our inductive transfer model for P=2 is shown in the Figure 1. At day 1 there is no past classifier. Therefore, the classifier of day 1 is built using the T training cases without any prediction features. At day 2 there is the past classifier of day 1. For each cases of day 2 the prediction of the classifier of day 1 is added as an additional binary feature. Now the classifier of day 2 is built using the T training cases where each training case has an extra binary feature. At day 3, we have 2 previous days' classifiers available. Therefore, two additional prediction features are added to the each cases of day 3. At day 4 also there are only 2 previous day's of classifiers are available as we choose to use P=2 in this example.



T : Number of training cases on any day
C_i : Classifier built on day i using T training cases of that day
$P(C_i)$: Prediction of the classifier C_i on each of the cases

Figure 1 Inductive transfer model

One problem in doing inductive transfer in this fashion is all past classifiers remain always in use for any value of P > 0. For P = 1 today's classifier depends on the yesterday's classifier only. But the yesterday's classifier depends on the one from the day before it, and so on. To break this recurrence while doing inductive transfer from the past classifiers we have to make sure that the past classifiers are independent. This is done by building the past classifiers separately simply using the T training data without adding prediction features to them. That is, each day two different classifiers are built. One is dependant on P previous days' classifiers and this classifier is used in predicting the remaining (N-T) cases of that day. This classifier is referred here as 'today's classifier'. Average performance of the today's classifiers over a period of time (365 days) is used as the performance measure of the system. The other classifier

is trained without any prediction features and therefore independent of any previous classifier. This latter classifier is used while doing inductive transfer in future days. Here the term 'previous days' classifiers' always refer to this kind of classifiers.

At this point readers might think instead of using past classifiers a better method could be training the today's classifier with the past training data itself. That is, instead of using P past classifiers one should simply use P*T past training cases along with the T training cases available today to train the today's classifier. As observed by George Forman [5] this simple method does not perform very well. Apart from that, doing inductive transfer using past classifiers has additional advantages. Firstly, we need to add only P additional features to the cases. Using (P+1)*T training cases can significantly increase the feature space. This is especially true for the bag of words representation of the documents. Secondly; due to concept drift the past training examples can mislead the today's classifier. Whereas if the predictions of past classifiers are used as features, a state of the art classifier should be able ignore those features that are useless. Classifiers, like support vector machine (SVM) are known to be able to ignore large set of redundant words in the text classification task. Therefore, doing inductive transfer through features with a state of the art classifier reduces the risk of "negative transfer".

The inductive transfer model used here is very similar to the temporal inductive transfer (TIX) model described by Forman [5]. But to make the previous days' classifiers useful he had to use hindsight. Hindsight is the ground truth labels of a percentage of the past cases that were not in the training set. In his work, while training the previous days' classifiers in addition to the T training cases a percentage of (N-T) cases were included in the training set with proper ground truth labels. The best performance was observed using full hindsight. Full hindsight means previous days' classifiers are trained using all the N cases of the corresponding day. Note that in such scenario all the N cases have to be properly labeled. The problem here is there is no easy way to obtain the hindsight.

In our setting, no hindsight has been used. Previous days' classifiers are trained using only the T training cases of the corresponding day.

4. Feature generation using Wikipedia

In this section, we describe how features are generated from Wikipedia. Wikipedia is a community edited encyclopedia and it is the largest knowledge repository in the web. With more than 2 million articles, 4+ million registered users, and on average 15+ edits per page the coverage and quality of the Wikipedia articles are incomparable. The value of using such a huge knowledge repository in different information retrieval tasks is started being showing up in recent researches [6][7][8].

We downloaded the English language Wikipedia snapshot of November 26, 2006 from the URL http://download.wikimedia.org. After parsing the Wikipedia xml dump, the articles describing Wikipedia features, template articles, redirects, and articles containing less than 50 non "stop words" were removed. An inverted index of the remaining 1,174,107 articles was created using the open source indexing tool Lucene (http://lucene.apache.org/).

The Wikipedia features of a given article are generated by using the text of the article as a query to the Lucene index. Using the query 100 top matching Wikipedia pages are retrieved from the Lucene index. The terms appearing in the 'titles' of these 100 pages are used as the Wikipedia features of the given article.

Using entire article as a query to retrieve the Wikipedia pages may not be a good idea as it is subjected to the long query problem. But our preliminary experiment by using each sentence of the given article as a different query was yielding much worse results. Another point is, instead of the individual terms the entire title of a retrieved Wikipedia page could have been used as a single feature. But in our experiment, we obtained slightly better results by using each term of the retrieved titles as a separate feature. The terms of the retrieved Wikipedia titles are called here as Wikipedia features. These Wikipedia features can be used in conjunction with the 'bag of words' features of the articles for a classifier. But as shown in our experiment section there is a value in training a classifier only using the Wikipedia features as these features are more stable in terms of concept drift.

5. Experiments and Results

As mentioned earlier, we tested our method on the daily classification task on the Reuters RCV1 [9] corpus. The corpus contains more than 800,000 news articles produced over 365 days (from 1996-08-20 to 1997-08-19). The news articles are manually categorized to many topics.

We sorted the news articles by day and everyday only 400 articles were used just to keep the experiment time manageable. Out of those 400 articles, 100 articles were used as training cases and the job was to classify the remaining 300 articles everyday. The average macro F-measure over 365 days is used as the measure of performance.

Three methods were evaluated in our experiment. All three methods fundamentally differ only in representing the articles in terms of features. The first method represents the articles only using the 'bag of words', the next two use the Wikipedia features also.

Otherwise, all three methods deploy same inductive transfer model described in the Section 3. That is, each day today's classifier is trained using the 100 training articles and this classifier is used to predict the class labels of the remaining 300 articles of that day. Inductive transfer is done by adding the predictions of P pervious days' classifiers as additional features of the articles. The previous days' classifiers are trained without using these prediction features (to break the recurrence as discussed earlier).

Linear Support Vector Machine (SVM) of Weka library [10] (version 3.5) was used as the base classifier. Only binary feature weighting has been used with the complexity constant (C) of SVM equals to 1. Next we describe the different methods in details.

1. Baseline: Each news article is represented only using 'bag of words'. The stop words are removed.

2. BOW+Wiki: The bag of words features of each article is augmented with the Wikipedia features of the article. Therefore, in this representation the features of an article are either the terms appearing in the article or in the Wikipedia titles retrieved using the article as a query. Today's classifiers as well as the previous days' classifiers use this as the base representation of the articles. Note that in addition to these features today's classifiers add P prediction features to each article.

3. WikiOnly: Our conjecture is the Wikipedia features are more stable in terms of concept drift as it captures the background knowledge of contents of the articles. To make full use of it we choose to train the previous days' classifiers only using Wikipedia features of the articles. That is, everyday we train the classifier that will be used in future for inductive transfer with only the Wikipedia features of the training articles. The other classifier, i.e., today's classifier, is trained using bag of words (and P prediction features) similar to the method 1. Also, while generating the prediction features of today's articles the previous days' classifiers use only the Wikipedia features of the today's articles.

Each method is tested on the DCT of binary classification for several individual classes in the Reuters corpus. The results are shown here by varying the value of P. Figure 2 shows the average F-measure achieved by different methods on four major classes in the Reuters corpus; ECAT (economics), M13 (money markets), GCAT (government/social), GSPO (sports). Results for 30 most common classes in Reuters are also shown in Figure 3. The classes here are chosen following George Forman's work [5] to make the results comparable.

5.1. Discussion

Figure 2 shows that the performance of the baseline method remains flat with the increase of P. Only in the case of M13 we can see some marginal improvement as P increases. Note that the F-measure at P=0 gives the performance of strawman algorithm (no inductive transfer). Therefore, the flat curves of this method indicate that the inductive transfer from previous days' classifiers has almost no effect.

Next we observe that in the most cases BOW+Wiki method yields much higher average F-measure than the baseline for all different values of P. Here also P=0 means no inductive transfer but the articles are represented using bag of words as well as with the Wikipedia features. That led a significant increase in F-measure in most cases even for P=0 (almost 10 points for ECAT, GCAT, and GSPO). But in this case also the curves remain almost flat with the increase of P. This implies that in this method also there is not much value in using the predictions of the previous days' classifiers.

In WikiOnly method, today's classifiers use only the bag of words representation of the articles. Therefore, the performance at P=0 is the same as the baseline method. But as P increases the average F-measure starts improving. At P=128 this method either surpass the performance of BOW+Wiki method (for ECAT and M13) or at least yields same average F-measure. This huge increase in performance is just because of the inductive transfer from the previous days' classifiers. Here the previous days' classifiers are built only using the Wikipedia features of the articles. This supports our hypothesis that the classifiers trained with some background knowledge are more effective in doing inductive transfer.

Since the WikiOnly method was performing best in our experiment, in Figure 3 we show the performance impact of this method on 30 major classes in the Reuters corpus. In this figure, the length of an arrow indicates the impact on performance for P=0 to P=128. It can be seen that for the majority of the classes there are significant improvements in the average F-measure. Also, there is no negative impact on the performance (no downward arrow).

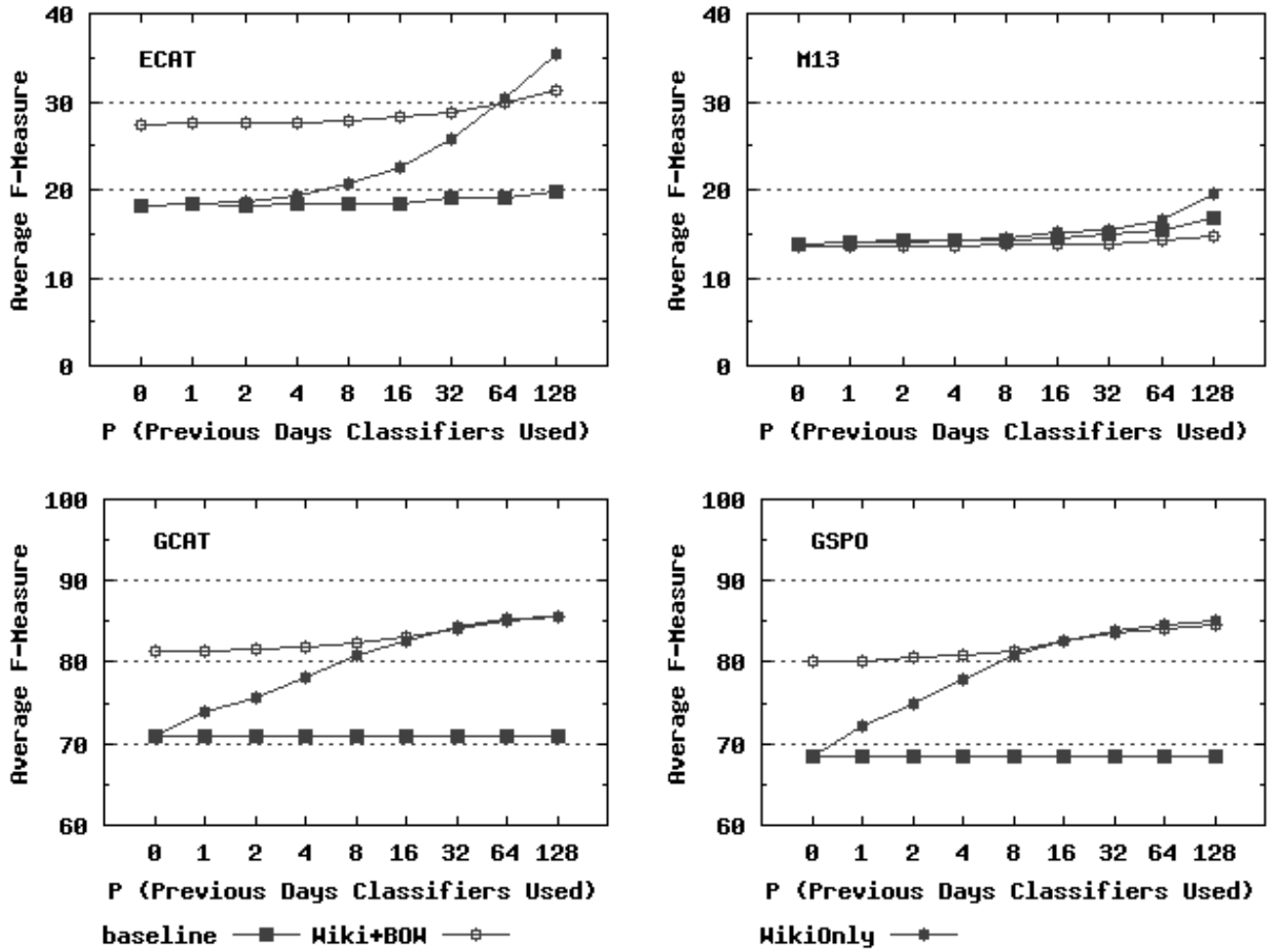


Figure 2 Results for 4 major classes in Reuters: ECAT and M13 (top), GCAT and GSPO (bottom)

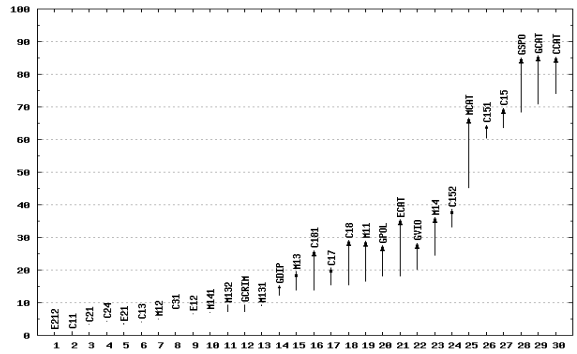


Figure 3 Results for 30 major classes in Reuters

6. Related Works

In terms of the inductive transfer model and experimental setting our work is most closely related to the work of George Forman on concept drift [5]. The major concern sited by him was that the inductive transfer from the past classifiers in DCT becomes really useful only when hindsight is used. We also observed that when the articles are represented using plain bag of words and only few (100) training cases are available each day then the previous days' classifiers are not very useful (the baseline method in Figure 2). To overcome this, in his method the ground truth labels from the test examples (the remaining 300 cases in our experimental setting) were used to train the previous days' classifiers. Obtaining the ground truth label of the test examples is expensive and that is the purpose of building a classifier. In most cases, our method was able to successfully overcome this major obstacle. It has also achieved a comparable performance to that of what observed by Forman with full (100%) hindsight.

Another body of recent research that is related to our work is using Wikipedia to perform different information retrieval tasks. Recently, Gabrilovich et al have shown how Wikipedia can be used in doing better text classification [6] and computing the semantic relatedness [7]. Earlier we also observed that Wikipedia

can be used in improving the accuracy of clustering short texts [8]. In the text classification work of Gabrilovich [6], the bag of words representation of a text document was appended with the Wikipedia titles retrieved using several queries constructed from the document. The queries were the words, sentences, paragraphs, and the entire document. A significant improvement in text classification accuracy was observed. Our primary interest here is to make the text classifiers more predictive in related but different tasks. As shown here when the classifiers are built only using the Wikipedia features they remain predictive as time progresses. Also, in our preliminary experiment, we observed that retrieving Wikipedia features using multiple queries (words, sentences and paragraphs) is not a good idea in this setting. In that case many Wikipedia features correspond to the individual terms of the given document and therefore not very stable over time.

In general, major prior work in inductive transfer is done on how to combine the outputs of different classifiers [4][11]. Here we did not do any experiment to intelligently combine the outputs of the different classifiers. Our main focus was to train the text classifiers with more stable features to make them more effective in transferring knowledge.

7. Conclusions and Future Work

We have shown a method of making inductive transfer more effective in the text domain. Our hypothesis is that the Wikipedia provides background knowledge of the contents of the articles. Therefore, the Wikipedia features are more stable and knowledge transfer done from the classifiers built with the Wikipedia features is more effective. When applied to the daily classification task on the Reuters corpus our method showed significant improvement in performance. It was also able to overcome the major obstacle faced in a recent work on the same task. We believe the observation here that the Wikipedia features can help in making inductive transfer more effective is useful for other inductive transfer model in the text domain.

Future works include testing our method in other inductive transfer models, more intelligent methods of retrieving and using Wikipedia features. We would also like to quantify the amount of change in the underlying concepts the Wikipedia features can sustain. This will provide the understanding of how much the related tasks can be dissimilar for inductive transfer with Wikipedia features to remain effective.

8. References

[1] D. Silver, G. Bakir, K. Bennett, R. Caruana, M. Pontil, S. Russell, and P. Tadepalli, organizers. *Workshop on Inductive Transfer: 10Years Later. 19th Conf. on Neural Information Processing Systems (NIPS)*, 2005

[2] P. Wu, and T. G. Dietterich, "Improving SVM accuracy by training on auxiliary data sources", *Proceedings of the twenty-first International Conference on Machine learning*, 2004

[3] M. T. Rosenstein, Z. Marx, and L. P. Kaelbling, "To Transfer or Not To Transfer", *Workshop on Inductive Transfer: 10Years Later. 19th Conf. on Neural Information Processing Systems (NIPS)*, 2005

[4] K. D. Bollacker, and J. Ghosh, "A Supra-Classifier Architecture for Scalable Knowledge Reuse", *Proceedings of the fifteenth International Conference on Machine Learning*, 1998

[5] G. Forman, "Tackling Concept Drift by Temporal Inductive Transfer", *Proceedings of the 29th Annual International ACM SIGIR Conference*, 2006

[6] E. Gabrilovich, and S. Markovitch, "Overcoming the Brittleness Bottleneck using Wikipedia: Enhancing Text Categorization with Encyclopedic Knowledge" *Proceedings of The 21st National Conference on Artificial Intelligence (AAAI)*, 2006

[7] E. Gabrilovich, and S. Markovitch, "Computing Semantic Relatedness using Wikipedia-based Explicit Semantic Analysis" *Proceedings of The 20th International Joint Conference on Artificial Intelligence (IJCAI)*, 2007

[8] S. Banerjee, K. Ramanathan, and A. Gupta, "Clustering Short Texts using Wikipedia", *Proceedings of the 30th Annual International ACM SIGIR Conference, 2007*

[9] D. Lewis, Y. Yang, T. Rose, and F. Li, RCV1: "A New Benchmark Collection for Text Categorization Research". *Journal of Machine Learning Research*, 5:361-397, 2004

[10] I. H. Witten, and E. Frank, "Data Mining: Practical machine learning tools and techniques", 2nd Edition, Morgan Kaufmann, San Francisco, 2005.

[11] P. N. Bennett, S. T. Dumais, and E. Horvitz, "Inductive Transfer for Text Classification using Generalized Reliability Indicators", *Proceedings of ICML Workshop on The Continuum from Labeled to Unlabeled Data*, 2003.