Co-Clustering based Classification Algorithm with Latent Semantic Relationship for Cross-Domain Text Classification through Wikipedia

Peter and Maxwell

Abstract--- *Conventional schemes to document classification need labeled data to build consistent and precise classifiers. On the other hand, labeled data are rarely available, and normally too expensive to obtain. Provided a learning task for which training data are not available, abundant labeled data possibly will exist for a different however related domain. One would like to make use of the related labeled data as auxiliary information to accomplish the classification task in the target domain. In recent times, the paradigm of transfer learning has been introduced to enable efficient learning strategies when auxiliary data obey a different probability distribution. A co-clustering based classification schemes has been proposed earlier to deal with cross-domain text classification. Here, the idea underlying this approach is extended by making the latent semantic relationship between the two domains explicit. This objective is achieved with the use of Wikipedia. Consequently, the pathway that permits propagating labels between the two domains not only captures common words, however also semantic concepts in accordance with the content of documents. Results empirically demonstrates the efficacy of the semantic-based approach to cross-domain classification using a variety of real data.*

I. Introduction

DOCUMENT classification is a key task for many text mining applications. For example, the Internet is a vast repository of disparate information growing at an exponential rate. Efficient and effective document retrieval and classification systems are required to turn the massive amount of data into useful information, and eventually into knowledge. Unfortunately, traditional approaches to classification require labeled data in order to construct reliable and accurate classifiers. Labeled data are seldom available, and often too expensive to obtain. On the other hand, given a learning task for which training data are not available, abundant labeled data may exist for a different but related domain. One would like to use the related labeled data as auxiliary information to accomplish the classification task in the target domain. Traditional machine learning approaches cannot be applied directly, as they assume that training and testing data are drawn from the same underlying distribution. Recently, the paradigm of transfer learning has been introduced to enable effective learning strategies when auxiliary data obey a different probability distribution. A co-clustering based classification algorithm has been proposed to tackle cross-domain text classification [2]. Let Di be the collection of labeled auxiliary documents, called in-domain documents, and do be the set of (out-of-domain) documents to be classified (for which no labels are available). Di and Do may be drawn from different distributions. Nevertheless, since the two domains are related, e.g., baseball vs. hockey, effectively the conditional probability of a class label given a word is similar in the two domains. The method leverages the shared dictionary across the in domain and the out-of-domain documents to propagate the label information from Di to Do. If a word cluster ˆ w usually appears in class c in Di, then if document d $\in$ Do contains the same word clusters ˆ w, it is likely that d belongs to class c as well. The co-clustering approach in [2] (called CoCC) leverages the common words of Di and Do to bridge the gap between the two domains. The method is based on the "Bag of Words" (BOW) representation of documents, where each document is modeled as a vector with a dimension for each term of the dictionary containing all the words that appear in the corpus. In this work, we extend the idea underlying the CoCC algorithm by making the latent semantic relationship between the two domains explicit. This goal is achieved with the use of Wikipedia. By embedding background knowledge constructed from Wikipedia, we generate an enriched representation of documents, which is capable of keeping multi-word concepts unbroken, capturing the semantic closeness of synonyms, and performing word sense disambiguation for polysemous terms. By combining such enriched representation with the Co CC algorithm, we can perform cross-domain classification based on a semantic bridge between the two related domains. That is the resulting pathway that allows propagating labels from Di to Do not only captures common words, but also semantic concepts based on the content of documents. As a consequence, even if the two corpora share few words, our technique is able to bridge the gap by embedding semantic information in the extended representation of documents. As such, improved classification accuracy is expected, as also demonstrated in our experimental results. In our previous work [16], we derived a thesaurus from Wikipedia, which explicitly defines synonymy, hyponymy and associative relations between concepts. Using the thesaurus constructed from Wikipedia, semantic information was embedded within the document representation, and we proved via experimentation that improved classification accuracy can be achieved [15]. In this work, we leverage these techniques to develop a semantic-based cross-domain classification approach.

Peter, University of Hamburg, Germany.
Maxwell, Southern Cross University, Australia.
DOI:10.9756/BIJDM.8330

II. RELATED WORK

Cross-domain classification is related to transfer learning, where the knowledge acquired to accomplish a given task is used to tackle another learning task. In [14], the authors build a term covariance matrix using the auxiliary problem to measure the co-occurrence between terms. The term covariance is then applied to the target learning task. In [6], the authors model the text classification problem with a linear function which takes the document vector representation as input, and provides in output the predicted label. Under this setting, different text classifiers differ only on the parameters of the linear function. A meta-learning method is introduced to learn how to tune the parameters. In [4], Dai et al. modified the Naive Bayes classifier to handle a cross-domain classification task. The technique first estimates the model based on the distribution of the training data. Then, an EM algorithm is designed under the distribution of the test data. KL-divergence is used to measure the distance between the training and test data distributions. An empirical fitting function based on KL-divergence is used to estimate the trade-off parameters of EM. In [3], Dai et al. altered Boosting to address cross domain classification problems. Their basic idea is to select useful instances from auxiliary data, and use them as additional training data for predicting the labels of test data.

However, to identify the most helpful additional training instances, the approach relies on the existence of some labeled testing data, which in practice may not be available. In [9, 8], Gabrilovich et al. proposed a method to integrate text classification with Wikipedia. They first build an auxiliary text classifier that can match documents with the most relevant articles of Wikipedia, and then augment the bag-of-word representation with new features corresponding to the concepts (mainly the titles) represented by the relevant Wikipedia articles. They perform feature generation using a multi-resolution approach: features are generated for each document at the level of individual words, sentences, paragraphs, and finally the entire document. This method only leverages text similarity between text fragments and Wikipedia articles, ignoring the abundant structural information within Wikipedia, e.g. internal links. The processing effort of this method is very high, since each document needs to be scanned many times. Furthermore, the feature generation procedure inevitably brings a lot of noise, because a specific text fragment contained in an article may not be relevant for its discrimination. In [1], Banerjee et al. tackled the daily classification task (DCT) [7] by importing Wikipedia knowledge into documents. Using Lucerne to index all Wikipedia articles, each document is used as a query to retrieve the top 100 matching Wikipedia articles. The corresponding titles become new features. This technique is prone to bring a lot noise into documents. Similarly to [7], documents are further enriched by combining the results of the previous n daily classifiers with new testing data. By doing so, the authors claim that the combined classifier is at least no worse than the previous n classifiers. However, this method is based on the assumption that a category may be comprised of a union of (potentially undiscovered) subclasses or themes, and the class distribution of these subclasses may shift over time.

III. THE CO CC ALGORITHM

The authors in [2] use co-clustering to perform crossdomain text classification. We summarize here the Co CC algorithm [2]. Let Di and Do be the set of in-domain and out-ofdomain data, respectively. Data in Di are labeled, and C represents the set of class labels. The labels of Do (unknown) are also drawn from C. Let W be the dictionary of all the words in Di and Do. The goal of coclustering Do is to simultaneously cluster the documents Do into |C| clusters, and the words W into k clusters. Let ˆD o = { ˆ d1, ˆ d2, ..., ˆ d|C|} be the |C| clusters of Do, and ˆW = { ˆ w1, ˆ w2, ..., ˆ wk} the k clusters of W. Following the notation in [5], the objective of co-clustering Do is to find two mappings CDo : {d1, ..., dm} → {ˆ d1, ˆ d2, ..., ˆ d|C|} and CW : {w1, ...,wn} → {ˆ w1, ˆ w2, ..., ˆ wk}, where |Do| = m and |W| = n. The tuple (CDo, CW), or (ˆD o, ˆW), represents a co-clustering of Do. To compute (ˆD o, ˆW), a two step procedure is introduced in [2], as illustrated in Figure 1 (the initialization step is discussed later). Step 1 clusters the out-of-domain documents into |C| document clusters according to the word clusters ˆW . Step 2 groups the words into k clusters, according to class labels and out-of-domain document clusters simultaneously. The second step allows the propagation of class information from Di to do, by leveraging word clusters. Word clusters, in fact, carry class information, namely the probability of a class given a word cluster. This process achieves the classification of out-of-domain documents. As in [5], the quality of the co-clustering (ˆD o, ˆW) is measured by the loss in mutual information

$$I\,(Do; W) - I\,(\hat{D}o; \hat{W}) \quad (1)$$

Thus, co-clustering aims at minimizing the loss in mutual information between documents and words, before and after the clustering process. Similarly, the quality of word clustering is measured by

$$I(C; W) - I(C; \hat{W}) \quad (2)$$

Where the goal is to minimize the loss in mutual information between class labels C and words W, before and after the clustering process by combining (1) and (2), the objective of co-clustering based classification becomes:

$$\min_{\hat{D}o,\hat{W}} \{I(Do;W) - I(\hat{D}o; \hat{W}) + \lambda(I(C;W) - I(C; \hat{W}))\}$$

Where λ is a trade-off parameter that balances the effect of the two clustering procedures. The above objective function enables the classification of out-of-domain documents via co- lustering, where word clusters provide a walkway for labels to migrate from the in-domain to the out-of domain documents. The Co CC algorithm computes a co clustering (CD o, CW) that corresponds to a local minimum of the above equation. For details, see [2]. The Co CC algorithm requires an initial co-clustering (C (0) Do, C (0) W) in input. As depicted in Figure 1, in [2] a Naïve Bayes classifier is used to initialize the out-of-domain documents into clusters. The initial word clusters are generated using the CLUTO software [10] with default parameters.

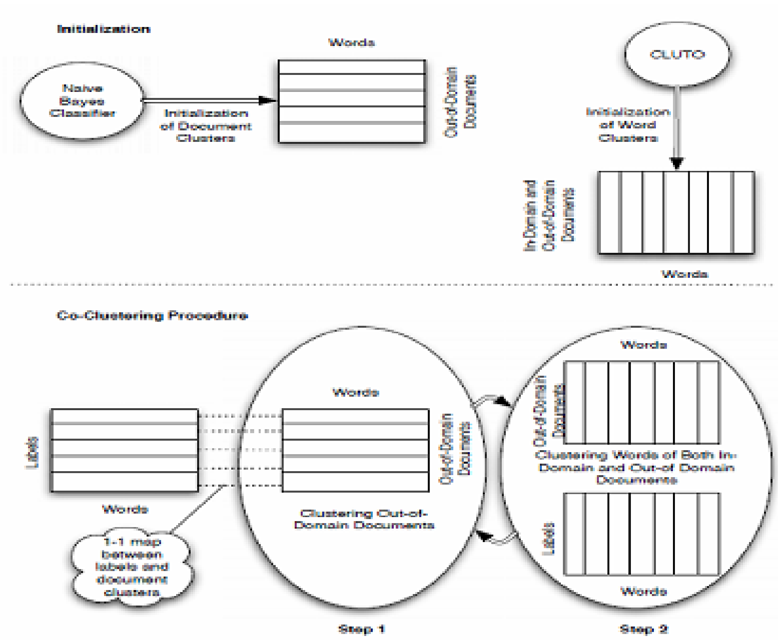

Figure 1: Co-Clustering for Cross-Domain Text Classification

IV. Semantic-Based Cross-domain Classification

We now present the methodology based on Wikipedia to embed semantics into document representation, and our overall approach to cross-domain classification. The thesaurus derived from Wikipedia provides a list of concepts. It leverages the hyperlink structure of Wikipedia to capture semantic relations between concepts, namely equivalence (synonymy), hierarchical (hyponymy), and associative. In particular, since associative hyperlinks capture different degrees of relatedness, three measures have been introduced to properly rank associative links between articles (or concepts) [16]: Content-based, Out-link category based, and Distance-based. The content-based measure (denoted as SBOW) is based on the bag-of-words representation of Wikipedia articles. Each article is modeled as a tf-idf vector: the value associated to a given term reflects its frequency of occurrence within the corresponding document (Term Frequency, or tf), and within the entire corpus (Inverse Document Frequency, or idf). The associative relation between two articles is then measured by computing the cosine similarity between the corresponding vectors. The out-link category-based measure compares the outlink categories of two associative articles. The out-link categories of a given article are the categories to which out-link articles from the original one belong. The larger the number of shared categories, the stronger the associative relation between the articles. To capture this notion of similarity, articles are represented as vectors of out-link categories, where each component corresponds to a category, and the value of the i-th component is the number of out-link articles which belong to the i-th category. Cosine similarity is then computed between the resulting vectors, and denoted SOLC. The third measure is a distance measure. The distance between two articles is measured as the length of the shortest path connecting the two categories they belong to, in the acyclic graph of the category taxonomy of Wikipedia. The distance measure is normalized by taking into account the depth of the taxonomy. It is denoted Dcat. A linear combination of the three measures quantifies the overall strength of an associative relation between concepts [16]:

$$S = \lambda_1 S_{BOW} + \lambda_2 S_{OLC} + (1 - \lambda_1 - \lambda_2)(1 - D_{cat}) \quad (3)$$

Where $\lambda_1, \lambda_2 \in (0, 1)$ are parameters to weigh the individual measures. The Wikipedia-based thesaurus is used to derive an extended vector space model for documents [15]. The BOW model of a document d is defined as follows: $\varphi : d \mapsto \varphi(d) = (\text{tf-idf}(t_1, d), \ldots, \text{tf-idf}(t_D, d)) \in R^D$, where $\text{tf-idf}(t_i, d)$ is the tf -idf value of term t_i in document d, and D is the size of the dictionary. Following the procedure introduced in [15], for each document in a given corpus, Wikipedia concepts mentioned therein are identified. An exact matching strategy is used; that is, only the concepts that explicitly appear in the document become the candidate concepts. Once the candidate concepts have been identified, the Wikipedia thesaurus is used to select synonyms, hyponyms, and associative concepts of the candidate ones. The vector associated to a document d is then enriched to include such related concepts: φ(d) = (<terms>,<candidate concepts>,<related concepts>)1. The value of each component corresponds to a tf-idf value. The feature value associated to a related concept is the tf-idf value of the corresponding candidate concept. Semantic information is embedded in φ(d) by means of a proximity matrix P defined for each pair of concepts. P is a symmetric matrix whose elements are defined as follows. For any two terms t_i and t_j, $P_{ij} = 0$ if $i \neq j$; $P_{ij} = 1$ if $i = j$. For any term t_i and any concept c_j, $P_{ij} = 0$. For any two concepts c_i and c_j :

$$P_{ij} = \begin{cases} 1 & \text{if } c_i \text{ and } c_j \text{ are synonyms;} \\ \mu^{-depth} & \text{if } c_i \text{ and } c_j \text{ are hyponyms;} \\ S & \text{if } c_i \text{ and } c_j \text{ are associative concepts;} \\ 0 & \text{otherwise.} \end{cases}$$

S is computed according to equation (3). depth is the distance between the corresponding categories of two hyponym concepts in the category structure of Wikipedia. M is a back-off factor, which regulates how fast the proximity between two concepts decreases as their category distance increases. Following [15], we set μ = 2in our experiments. By composing the vector φ(d) with P, we obtain the desired extended vector space model for document d: ˜φ(d) = φ(d)P. ˜φ(d) is a less sparse vector with non-zero entries not only for concepts mentioned in d, but also for all concepts that are semantically similar to those present in d. We apply this procedure to all documents in Di and in Do. As a result, the representation of two related documents d1 ∈ Di and d2 ∈ Do corresponds to two close vectors ˜φ(d1) and ˜φ(d2) in the extended vector space model. In other words, the extended vector space model applied to Di and Do has the effect of enriching the shared dictionary with concepts that encapsulate the content of documents. As such, related domains will have a shared pool of terms/concepts of increased size that has the effect of making explicit their semantic relationships. We thus perform co-clustering based cross-domain classification by providing the CoCC algorithm the extended vector space model of in-domain and out-of-domain documents. The setW now comprises the new dictionary, which includes terms and concepts, both candidate and related. We note that concepts form individual features, without undergoing stemming, or splitting of multi-word expressions.

V. EMPIRICAL EVALUATION

We evaluated our approach using the 20 Newsgroups [11], and the SRAA [12] data sets. We split the original data in two corpora, corresponding to in-domain and outof-domain documents. Different but related categories are selected for the two domains. Data sets across different classes are balanced. Table 1 shows how categories were distributed for each data set generated from the 20 Newsgroups corpus. We generated ten different data sets comprised of different combinations of categories. Each set contains several top categories, which also define the class labels. Data are split into two domains based on their subcategories. The SRAA [12] data set contains 73,218 articles from four discussion groups on simulated auto racing, simulated aviation, real autos, and real aviation. It is often used for binary classification, where the task can be defined as the separation of documents on “real” versus “simulated” topics, or on “auto” vs. “aviation”. We generated two binary classification problems accordingly, as specified in Table 3. In our experiments, we compare the classification results of the Co CC approach based on the BOW representation of documents (denoted Co CC without enrichment), and based on the extended vector space model (denoted CoCC with enrichment). The CoCC algorithm uses a Naive Bayes classifier to initialize the out-of-domain documents into clusters.

Thus, we also report the results of Naive Bayes, with and without enrichment, respectively. Standard pre-processing was performed on the raw data. All letters in the text were converted to lower case, stop words were eliminated, and stemming was performed using the Porter algorithm [13]. Words that appeared in less than three documents were eliminated from consideration. Term Frequency was used for feature weighting when training the Naive Bayes classifier, and for the CoCC algorithm. To compute the enriched representation of documents,

We need to set the parameters λ1 and λ2 in Equation (3). These parameters were tuned according to the methodology suggested in [16]. As a result, the values λ1 = 0.4 and λ2 = 0.5 were used in our experiments. The CoCC algorithm requires the initialization of document clusters and word clusters. We follow the methodology adopted in [2], and compute the initial document clusters using a Naive Bayes classifier, and the initial word clusters using the LUTO software [10] with default parameters. The Naive Bayes classifier is trained using Di. The trained classifier is then used to predict the labels of documents in Do. In our implementation, we keep track of class labels associated to clusters by the Naive Bayes classifier, to compute the final labels of documents in Do. Table 2 presents the precision rates obtained with Naïve Bayes and the CoCC algorithm, both with and without enrichment. The results of the CoCC algorithm corresponds to λ = 0.25, and 128 word clusters. The precision values are those obtained after the fifth iteration. Table shows that the CoCC algorithm with enrichment provides the best precision values for all data sets. For each data set, the improvement offered by CoCC with enrichment with respect to the Naive Bayes classifier (with enrichment), and with respect to CoCC without enrichment is significant. As shown in Table 2, themost difficult problem is the one with four categories: rec vs talk vs sci vs comp. A closer look to the precision rates reveals that almost all “recreation” and “talk” documents in Do are correctly classified. The misclassification error is mostly due to the fact that the top categories “science” and “computers” are closely related (in particular, the sub-category “electronics” of “science” may share many words with the category “computers”). As a consequence, several “science” documents are classified as “computers” documents. Nevertheless, Co CC with enrichment achieves 71.3% accuracy, offering a 8.9% improvement with respect to Co CC without enrichment, and a 17.5% improvement with respect to Naive Bayes. It is interesting to observe that in all cases the Naive Bayes classifier itself largely benefits from the enrichment. We also show the precision achieved by Co CC with enrichment as a function of the number of iterations for the four multicategory problems considered in our experiments. In each case, the algorithm reached convergence after a reasonable number of iterations (at most 27 iterations). The improvements in precision ith respect to the initial clustering solution are confined within the first little iteration. We obtained a consistent result across all data sets. For this reason, we provide the precision results obtained after the fifth iteration. We also tested the sensitivity of Co CC with enrichment with respect to the λ parameter, and with respect to the number of clusters. We report the results obtained on the three category problem derived from the 20 Newsgroups data set: sci vs talk vs comp. Following the settings in [2], we used λ values in the range (0.03125, 8), with three different numbers of word clusters: 16, 64 and 128. Figure 3 shows the results. Overall, the precision values are stable. A reasonable range of values for λ is [0.25, 0.5]. The precision values as a function of different number of clusters are given in Figure 4. We tested different numbers of clusters between 2 and 512 for three different values of λ: 0.125, 0.25, and 1.0. The same trend was obtained for the three λ values. Precision increases significantly until a reasonable number of word clusters is achieved. A value of 128 provided good results for all problems considered (this finding is consistent with the analysis conducted in [2]).

Cross-Domain Classification Precision Rates

Data Set	w/o enrichment		w/ enrichment	
	NB	CoCC	NB	CoCC
rec vs talk	0.824	0.921	0.853	0.998
rec vs sci	0.809	0.954	0.828	0.984
comp vs talk	0.927	0.978	0.934	0.995
comp vs sci	0.552	0.898	0.673	0.987
comp vs rec	0.817	0.915	0.825	0.993
sci vs talk	0.804	0.947	0.877	0.988
rec vs sci vs comp	0.584	0.822	0.635	0.904
rec vs talk vs sci	0.687	0.881	0.739	0.979
sci vs talk vs comp	0.695	0.836	0.775	0.912
rec vs talk vs sci vs comp	0.487	0.624	0.538	0.713
real vs simulation	0.753	0.851	0.826	0.977
auto vs aviation	0.824	0.959	0.933	0.992

REFERENCES

[1] S. Banerjee, "Boosting inductive transfer for text classification using Wikipedia", International Conference on Machine Learning and Applications, 2007.

[2] W. Dai, G.R. Xue, Q. Yang and Y. Yu, "Co-clustering based classification for out-of-domain documents", International Conference on Knowledge Discovery and Data Mining, 2007.

[3] W. Dai, Q. Yang, G.R. Xue and Y. Yu, "Boosting for transfer learning", International Conference on Machine Learning, 2007.

[4] W. Dai, Q. Yang, G.R. Xue and Y. Yu, "Transferring naïve bayes classifiers for text classification", AAAI Conference on Artificial Intelligence, 2007.

[5] I.S. Dhillon, S. Mallela and D.S. Modha, "Information theoretic co-clustering", International Conference on Knowledge Discovery and Data Mining, 2003.

[6] C. Do and A.Y. Ng, "Transfer learning for text classification", Annual Conference on Neural Information Processing Systems, 2005.