Unsupervised Domain Adaptation based on Text Relatedness

Georgios Petasis
Software and Knowledge Engineering Laboratory,
Institute of Informatics and Telecommunications,
National Centre for Scientific Research "Demokritos",
Athens, Greece
petasis@iit.demokritos.gr

Abstract

In this paper an unsupervised approach to domain adaptation is presented, which exploits external knowledge sources in order to port a classification model into a new thematic domain. Our approach extracts a new feature set from documents of the target domain, and tries to align the new features to the original ones, by exploiting text relatedness from external knowledge sources, such as WordNet. The approach has been evaluated on the task of document classification, involving the classification of newsgroup postings into 20 news groups.

1 Introduction

The portability of natural language processing (NLP) systems to new thematic domains is still a research area that attracts a significant research interest. During the last two decades, the use of machine learning has greatly improved the adaptability to new domains, or even languages. However, the vast majority of machine learning algorithms operate under a basic assumption: both the training and test data should use the same feature space, and follow the same distribution, suggesting that both should originate from the same thematic domain. When the distribution changes, the models must be re-generated from newly collected data. The adaptation can be separated into three large categories, according to the available data from the new domain. In supervised approaches, there is an adequate number of labelled data to train the model from scratch, on the new domain. When a limited number of labelled data are available, usually too few to train a model with satisfactory performance, along with unlabeled ones, the adaptation process is characterised as semi-supervised. Finally, unsupervised approaches must adapt their model to a new domain by learning solely from unlabelled examples.

Transfer learning or *knowledge transfer* is a research area, which tries to extract knowledge from previous experience and apply it on new learning tasks. Based on the idea that prior knowledge (i.e. identifying oranges) can be used on new tasks (i.e. identifying lemons), transfer learning researches three main central problems (Zhang and Shakya, 2009): 1) how to extract the prior knowledge that is related, 2) how to represent the knowledge, and 3) how to apply the knowledge in the new learning task. *Domain adaptation* is a sub-category of transfer learning, where (Pan and Yang, 2010):

1. The source and target domains are different, but related.
2. The source and target tasks are the same (i.e. classification or regression).
3. Labelled examples are available for the source domain.
4. Only unlabeled examples are available for the target domain.

In this paper, we propose a novel approach for the task of domain adaptation. Our method concentrates on the *feature space*, by trying to expand the features of the source domain with features that appear only in the target domain. Features that originate from the two different domains are aligned or linked to each other, through text relatedness. Text relatedness can take many forms, but we have opted for a simple relatedness measure, based on WordNet (Miller, 1995) synonymity.

The rest of the paper is organized as follows: in section 2 related work is presented, where our method is compared to existing approaches. In section 3 our approach to model adaptation based on text relatedness is presented, while section 4 presents evaluation on the 20-newsgroup corpus (Lang, 1995). Finally, section 5 concludes this paper and presents some future directions.

Proceedings of Recent Advances in Natural Language Processing, pages 733–739,
Hissar, Bulgaria, 12-14 September 2011.

2 Related work

The task of transfer learning can be defined as follows: given a source domain D_S, a source task T_S, a target domain $D_T \neq D_S$, and a target task T_T, transfer learning aims in learning a function f_T that accomplishes task T_T, by exploiting knowledge derived from D_S and T_S. A fairly recent overview of the area of transfer learning is given in the survey of (Pan and Yang, 2010), including the definition of transfer learning, its relation to traditional machine learning, a categorisation of transfer learning approaches, and practical applications of transfer learning. More recent approaches that target the task of domain adaptation can be found on the ACL 2010 Workshop on Domain Adaptation for Natural Language Processing (DANLP 2010) (Daumé III et al., 2010).

A lot of approaches exist that perform model adaptation in a fully supervised way (i.e. requiring labelled examples for both the source and target domains). For example, EASYADAPT (Daumé III, 2007) augments the source domain feature space using features extracted from labelled data in target domain. Prior work on semi-supervised approaches to domain adaptation also exists in literature. Recent work in domain adaptation has focused on approaches such as *self-training* and *structural correspondence learning* (SCL). The former approach involves adding self-labelled data from the target domain produced by a model trained in-domain (McClosky, Charniak and Johnson, 2006). The latter approach focuses on ways of generating shared source-target representations based on good pivot features (Blitzer, McDonald and Pereira, 2006); (Ando, 2004); (Daumé III, Kumar and Saha, 2010).

However, the approach presented in this paper follows an *unsupervised* approach, thus requiring no labelled examples from the target domain. Unsupervised approaches try to exploit knowledge either from external knowledge sources, like our approach and (Gabrilovich and Markovitch, 2005), or from the distribution followed by the target domain (Thrun and Pratt, 1998); (Dai et al., 2007). The work presented in this paper can be categorised as an "unsupervised feature construction" approach, according to (Pan and Yang, 2010). Thus, approaches that try to extend a feature set through the unsupervised extraction of new features share some common ground with our approach. In (Gabrilovich and Markovitch, 2005) an approach that extracts new features by exploiting world knowledge is presented. World knowledge is represented through publically available ontologies, such as the Open Directory Project (ODP), where features from the source domain are mapped to appropriate ontology concepts, and "is-a" relations are exploited in order to acquire new features that augment the original feature set. Finally, the most appropriate features are selected through a feature selection phase. The work presented in (Zhang and Shakya, 2009) is also closely related to our approach: *feature correlation* is used in order to group features into *correlated groups*. For example, words like "orange", "lemon", "apple" and "pear" may often appear together in documents: aggregating them into a new correlated group "fruits", creates a new feature. If enough evidence exists in a document from the target domain (i.e. some of the features of the correlated group appear in the document), the feature that corresponds to the correlated group may help the task T_T in the target domain. In a sense, both approaches exploit information that can be characterised as "text relatedness" (or "feature relatedness"), as both "is-a" relations and correlation can be viewed as a relatedness measure between features. However, our method has also some important differences with these two methods. Our text relatedness measure is based on synonymity, as provided by an electronic dictionary such as WordNet. An electronic dictionary may be an easier resource to find than an ontology or hierarchy, thus our approach may have a small advantage in initial requirements when compared to (Gabrilovich and Markovitch, 2005). On the other hand, the calculation of feature correlation has no initial requirements in resources, but requires a corpus of adequate size, in order to extract the correlated groups. In addition, mining correlated groups may be computationally intensive if the feature set from the source domain is large enough (a problem tackled by limiting the source domain feature set to 2000 features, selected through mutual information, as reported in (Zhang and Shakya, 2009)). Finally, synonymity is a slightly more restricted text relatedness measure, compared to "is-a" relations (that can have many levels in the concept hierarchy) or correlation (which can relate possible unrelated features). Being a slightly more accurate text relatedness metric, it constitutes the need for feature selection, after the expansion of the source feature set, less important. In fact, our approach does not have a feature selection phase at all, in contrary to the two related approaches.

3 Domain adaptation based on text relatedness

The proposed methodology assumes a source domain D_S, a target domain $D_T \neq D_S$, a task T common for both domains, a feature space for the source domain $\mathcal{X}_S$, a label space $\mathcal{L}$ common for both domains, and a set of labelled examples originating from the source domain $L_S = \{X_1, \cdots, X_n\}$, where $X_i = \{x_1, \cdots, x_n, l_i\}$, $x_i \in \mathcal{X}_S$, $l_i \in \mathcal{L}$. In addition, our approach assumes a binary function $r(x_\alpha, x_\beta) \in \{0, 1\}$, $x_\alpha, x_\beta \in \mathcal{X}_S, \mathcal{X}_T$, which decides if two features are related, according to a text relatedness metric. Finally, a function $f_{\mathcal{X}_T}$ is assumed, that can extract a feature space $\mathcal{X}_T$ from the target domain D_T. The function $f_{\mathcal{X}_T}$ can be even a naive one, i.e. a function that returns all words in a corpus from the target domain D_T.

3.1 Text relatedness based on synonymity

Our approach assumes a binary relatedness function $r(x_\alpha, x_\beta)$, that can compare two features (either from the source or from the target feature spaces), and return whether the two features are related or not. Although many relatedness metrics can be devised and used, we have opted for a simple one, based on synonymity. Assuming an electronic dictionary, which contains synonyms, our text relatedness that is based on synonymity can be described with the following algorithm:

- If x_α and x_β are the same, return 1.
- Let S_α be the set of synonyms of x_α, and S_β the set of synonyms of x_β, according to the dictionary.
- If $x_\beta \in S_\alpha$ or $x_\alpha \in S_\beta$, return 1.
- If $S_\alpha \cap S_\beta \neq \emptyset$, return 1.
- Else, return 0.

In simple words, our synonymity relatedness metric returns true, if the two features are synonyms, or when they have at least one common synonym. The electronic dictionary that has been chosen is WordNet (Miller, 1995), as has already been mentioned. It should be noted that all synonyms for all senses are treated equally, without performing any kind of word sense disambiguation (Navigli, 2009), as is performed for example in the approach described in (Gabrilovich and Markovitch, 2005).

3.2 Extracting features from the target domain

Our approach assumes that there is a function $f_{\mathcal{X}_T}$, which can extract features from the target domain D_T. Since no further requirements are assumed about this function, the function can be as naive or complex as the task T requires. We have considered two feature extraction procedures, one naive, and one slightly more complex. The naive feature extraction (the aim of which is to be applied on the target domain D_T) simply extracts all the words that can be found on a corpus from D_T, minus the words that are considered as "stop words", and are filtered by using a stop word list. For the purposes of the experiments that will be presented in subsequent sections, the stop word filtering facilities offered by the Ellogon (Petasis et al., 2002) language engineering platform have been used.

A second feature extraction procedure has been additionally devised, aiming to be applied on the source domain D_S, in case such a need arises. This procedure examines all documents of a corpus, and calculates the TF-IDF score for every word of the document. "Stop words" are also rejected, and the rest of the remaining words are sorted according to their TF-IDF score, in a descending list. Then, an amount of the best scoring words, specified through a parameter θ (interpreted as a percent of the total words in a document), is extracted from each document, and added to the feature space that will be returned as the result.

3.3 Extracting new features

Once we have a method for extracting possible new features from the target domain D_T, through the function $f_{\mathcal{X}_T}$, and a text relatedness metric $r(x_\alpha, x_\beta)$, we can apply these two functions in order to acquire a feature set from the target domain:

- Let $\mathcal{X}_T^{\text{Initial}}$ be the feature space, as extracted from the target domain D_T by the function $f_{\mathcal{X}_T}$.
- Each feature $x_S \in \mathcal{X}_S$ from the source feature set is compared to each feature $x_T \in \mathcal{X}_T^{\text{Initial}}$ in the extracted from the target domain feature set. The function $r(x_\alpha, x_\beta)$ is used for comparing the pair of features.
- Features from the $\mathcal{X}_T^{\text{Initial}}$ that are not related to any feature in $\mathcal{X}_S$, are eliminated

from $\mathcal{X}_{\mathrm{T}}^{\mathrm{Initial}}$, leading to a new feature space $\mathcal{X}_{\mathrm{T}}^{\mathrm{Related}}$.

- As a final step, all features $\mathrm{x_T} \in \mathcal{X}_{\mathrm{T}}^{\mathrm{Related}}$ are examined: every feature $\mathrm{x_T}$ that is related to more than one features in $\mathcal{X}_{\mathrm{S}}$, is removed from $\mathcal{X}_{\mathrm{T}}^{\mathrm{Related}}$, leading to the final feature space that relates to the target domain $\mathcal{X}_{\mathrm{T}}^{\mathrm{Final}}$.

The result of this procedure, the final feature space that should be used for performing task T on the target domain D_T is the union of the two feature spaces: $\mathcal{X} = \mathcal{X}_S \cup \mathcal{X}_T^{Final}$.

3.4 Representing the extracted knowledge

The augmented feature space $\mathcal{X}$ that has been extracted as described in the previous subsection, contains all features of the source domain D_S, and new features from the target domain, each of which is unambiguously related to a single feature from D_S. The only unsolved issue is how this augmented feature space is going to be represented as vectors, which can be used with a machine learning algorithm. Although this decision may rely on the particular machine learning algorithm that will be used, empirical evaluation suggested that the best alternative is to form "groups of features", where each old feature is replaced by two features: the original one, plus the related one from the target feature space, if one exists. This representation has been proved beneficial, at least for the task we have chosen to evaluate our approach (document classification), the chosen representation (bag-of-words) and the chosen classifier (kNN with $k = 1$ and cosine similarity as the distance metric).

4 Empirical evaluation

This section will present an empirical evaluation of the proposed approach for domain adaptation based on text relatedness, with the help of the 20-newsgroup dataset (Lang, 1995): the 20-newsgroup dataset is a collection of approximately 20000 newsgroup documents, partitioned (nearly) evenly across 20 different newsgroups, and is a standard evaluation corpus in many works related to domain adaptation or transfer learning. The task chosen for the empirical evaluation is document classification.

4.1 The 20-newsgroup corpus

The 20-newsgroup corpus is preconfigured in training and testing material. Despite the fact that it is a popular evaluation corpus for domain adaptation approaches, it is unclear to us if all works that report results on the corpus use the same train/test partitioning, as different results are reported even for the base cases, as in (Pan and Yang, 2010) for example. In order to ease comparison with other approaches we opted in using the predefined train/test segmentation of the corpus, as it is distributed. Regarding the task, we will limit evaluation to the three more popular evaluation pairs: "rec vs talk", "rec vs sci", "sci vs talk".

The main idea behind the separation of these pairs, is that newsgroup posts from relevant but different newsgroups are put in the source/target domains. The "rec vs talk" class for example, may contain posts from the newsgroups "talk.politics.misc", "talk.politics.guns", "rec. motorcycles", and "rec.sport.hockey" as training material representing the source domain, while the test data (representing the target domain) may comprise from posts of the following newsgroups: "talk.politics.mideast", "talk.religion. misc", "rec.autos" and "rec.sport.baseball".

All posts in the three pairs of interest were pre-processed, in order for words to be recognised. A feature space from the posts constituting the training material was extracted, using the second method described in subsection 3.2, the one that extracts the top scoring words according to their TF-IDF weights, the number of which is controlled through a percentage of the total words of each post. This parameter was set to 0.003%, as it was found to roughly correspond to about one word from each post, leading for example to 4564 features for "rec vs sci", whose training material contains 4762 newsgroup posts. The reason behind this choice was to avoid possible over-fitting in the presence of too many features, and to provide our domain adaptation approach a chance to discover a large number of features from the target domain. As a measure of comparison, in (Zhang and Shakya, 2009) an initial feature space of 2000 features was selected.

Another point of interest is the choice of the machine learning algorithm, which will be used in order to learn from vectors. Support Vector Machines (SVMs) are quite popular as a base case in model adaptation problems, since prior studies found SVMs to offer the best performance, at least for document classification using a bag-of-words representation (Dumais et al., 1998); (Yang and Liu, 1999). However, since our approach expands the feature space, we wanted to evaluate the effect of the augmented feature

Pair	Source Domain Posts	Target Domain Posts	kNN ($k = 1$, cosine similarity)			SVM (LIBLINEAR)		
			Precision	Recall	F_1	Precision	Recall	F_1
rec vs sci	4762	3169	83.00%	41.90%	55.69%	83.62%	42.22%	56.11%
rec vs talk	4341	2891	83.35%	51.61%	55.51%	87.04%	53.89%	66.57%
sci vs talk	4325	2880	78.98%	41.63%	54.52%	82.67%	43.58%	57.07%

Table 1: Corpus characteristics and base case evaluation for the 20-newsgroup corpus.

Domain adaptation based on text relatedness						
Pair	kNN ($k = 1$, cosine similarity)			SVM (LIBLINEAR)		
	Precision	Recall	F_1	Precision	Recall	F_1
rec vs sci	64.88%	50.02% (+8.12)	56.49%	65.75%	50.69% (+8.47)	57.25%
rec vs talk	61.17%	55.44% (+3.83)	58.17%	65.11%	59.01% (+5.12)	61.91%
sci vs talk	59.60%	49.70% (+8.07)	54.20%	63.18%	52.69% (+9.11)	57.46%
(Shi, Fan and Ren, 2008)						
Pair	Recall (base/SVM)	Recall (TrAdaBoost)	Recall (AcTraK)			
rec vs sci	59.1%	67.4% (+8.3)	70.6% (+11.5)			
rec vs talk	60.2%	72.3% (+12.1)	75.4% (+15.2)			
sci vs talk	57.6%	71.3% (+13.7)	75.1% (+17.5)			

Table 2: Evaluation results on domain adaptation for the 20-newsgroup corpus. Results from (Shi, Fan and Ren, 2008) are also shown for comparison purposes (evaluated on different data partitioning).

space with the least possible intervention from the chosen machine learning algorithm. Thus, we selected one of the simplest machine learning algorithms available, the k-nearest neighbour algorithm (kNN). kNN does not have a training phase, it just classifies test instances using a similarity metric to measure distances from the training instances. In all experiments reported in this work, a kNN implementation was used with k=1, and cosine similarity as the distance metric.

The bag-of-words representation was used for all experiments in this paper. Under this representation, each document (newsgroup post) is represented with a single vector, which has the same dimension as the feature namespace in use. The value for each feature is binary: 1 represents that this feature exists in the document, 0 represents that this feature does not exist in the document. The characteristics of the 20-newsgroup corpus, as well as evaluation results for the base classifier are shown in Table 1. Despite the fact that kNN is the chosen classifier due to reasons already discussed, we have also applied an SVM algorithm with linear kernel, as implemented by the LIBLINEAR library (Fan et al., 2008). LIBLINEAR has been applied in order to ease comparisons with other approaches employing SVMs for classification.

The evaluation results of our approach are shown in Table 2. The upper part of Table 2 contains the evaluation results of our approach. The rows correspond to the examined pairs of newsgroups, while columns include information about the performance of both the kNN and LIBLINEAR classifiers, in terms of precision, recall and F-measure (F_1). In table columns concerning recall, the improvement from the base case is also displayed, as difference between percentages. The lower part of Table 2 contains evaluation results from (Shi, Fan and Ren, 2008), where two model adaptation approaches were evaluated and compared with SVMs, used as a base case. While experiments in (Shi, Fan and Ren, 2008) use a different partitioning of the corpus as training and testing data, suggesting that the performance of these approaches are not directly comparable to our approach, the improvement in performance provides a good indication of the contribution of the approaches, and can be compared to the improvement achieved by our approach.

As we can see from Table 2, the kNN classifier is able to provide answers for a much larger number of documents after the feature space has been augmented with features from the target domain. This is evident by the increase in recall. However, another aspect of feature space expansion should be noted: the classifier is able to provide an answer for a much larger number of newsgroup posts, even if the answer is not correct. For example, only 1600 (out of 3169) posts of the target domain contained features from the feature space of the source domain, in the case of the "rec vs sci" pair. However, after our approach expands the feature space with features from the target domain, 2289 posts of the target domain contained at least one feature from the

augmented feature space, offering the possibility for classifying a larger number of posts.

The increase in performance achieved by our approach ranges from 4% (for "rec vs talk") to 8% (for "rec vs sci"). In comparison, the algorithm TrAdaBoost (Dai et al., 2007) achieved an increase ranging from 8% to 14%. The algorithm TrAdaBoost employs boosting in a semi-supervised approach, which exploits a small set of labelled data from the target domain, in addition to a large labelled data set from the source domain., in order to minimise the importance of labelled data from source domain (through weighting) whose distribution does not match the one of the target domain. Considering the fact that our approach employs a simple classification algorithm (kNN, $k = 1$, binary features), along with a fairly simple text relatedness similarity (synonymity), our approach performed surprisingly well. AcTraK (Shi, Fan and Ren, 2008) achieves an additional improved of about 4% compared to TrAdaBoost, with the help of active learning in a semi-supervised approach, where labelled data may be asked when necessary.

4.2 Representing the augmented feature space

Given the specific choices we have done regarding the task of document classification for the representation and the machine learning algorithm in use, we have performed an empirical evaluation in order to examine the effect of different ways in representing the acquired knowledge. We have examined three cases, concerning the incorporation of the augmented features in $\mathcal{X}_{\mathrm{T}}^{\mathrm{Final}}$ to the vectorial representation:

Expanding training vectors: under this scenario, the new features are also represented in the vectors, increasing the dimensionality of the vectors. A new dimension is created for each feature in the $\mathcal{X}_{\mathrm{T}}^{\mathrm{Final}}$ feature space. The value for each new feature is the value of its related, original feature in this vector.

Expanding and duplicating vectors: this case is very similar to the previous one regarding dimensionality: the dimensionality also increases, identical to the previous case. However, there is a difference in how the values of new features are set: instead of placing the value 1 to the original training vector, if the linked original feature is also 1, the original vector is duplicated, and the value 1 is set in the copy, for the new feature. As a result, each original vector is duplicated as many times as there are augmented features whose value should be 1 for this vector. Each copy differs from the original one only at the value of one feature.

Grouping features: under this scenario, the dimensionality of the vectors is not increased. Instead some of the features become "grouped features": they occupy a single dimension in vectors, but they represent different words, when matched in documents. This case was used in the evaluation presented in the previous subsection.

The evaluation has been performed only for the "rec vs sci" pair of newsgroups, using the same classifier as in subsection 4.1. The results are shown in Table 3. Our approach managed to achieve an improvement in accuracy (recall) in all three cases. However, the improvement was significantly better for case 3, while case 2 performed worse than the other two methods. The reason for the worst improvement can be attributed to the fact that the number of vectors that were added was not enough to cover all possible permutations. Assuming N augmented features whose value must be 1 (as there are also N original features whose value is 1), $2N^2 - 1$ vectors must be inserted, in order to cover all possible permutations. However, adding so many vectors can quickly lead to an intractable problem. Instead our approach followed a more conservative path, adding only $N - 1$ vectors to the original training set, covering unfortunately only a part of possible cases, and not fully exploiting the potential of the augmented features.

	Precision	Recall	F_1
Case 1	79.26%	45.69%	57.96%
Case 2	76.79%	44.27%	56.16%
Case 3	64.88%	50.02%	56.94%
Base case	83.00%	41.90%	55.69%

Table 3: Evaluation results for various representations of the augmented feature space.

Case 1 was not too far from case 2. The reason for this behaviour can be attributed to the classification algorithm we have used. Cosine similarity depends on the number of common features with value 1 between the two vectors, divided by the magnitude of the two vectors. We can easily image a case where in a post, some of the original features without augmented ones exist in the post, but from the related features, only some of the augmented features exists, and none of the original related features exists. Trying to match such a test vector to a training one that has the augmented, but also their original related features set to 1, may be misclassified in favour of a vector with less magnitude, and possibly with no related features (both original and augmented)

set to one. Thus, also this case is unable to fully exploit the augmented features, as it may favour classifying test vectors with augmented features into training vectors without augmented features.

5 Conclusions and future work

In this paper, a domain adaptation approach was presented, that exploits text relatedness in the form of WordNet synonymity, in order to augment an initial feature space, derived from the source domain, with new features from the target domain. The proposed approached was empirically evaluated with the help of a manually annotated corpus. Evaluation results suggest that our approach can achieve an improvement comparable to other approaches that can be found in the bibliography, despite the fact that it employs kNN as its classifier to the task of document classification.

Since our current implementation of text relatedness is quite simple, based on WordNet synonymity, trying out more complex relatedness functions would be an interesting future direction to explore. A particularly interesting text relatedness function is Omiotis (Tsatsaronis, Varlamis and Vazirgiannis, 2010), which exploits many knowledge sources in order to estimate the relatedness between two words.

Acknowledgments

The author would like to acknowledge partial support of this work from the European Community Seventh Framework Programme, as part of the FP7 – 231854 SYNC3 project.

References

Ando, R.K. (2004) 'Exploiting unannotated corpora for tagging and chunking', Proceedings of the ACL 2004 on Interactive poster and demonstration sessions (ACLdemo '04), Stroudsburg, PA, USA.

Blitzer, J., McDonald, R. and Pereira, F. (2006) 'Domain Adaptation with Structural Correspondence Learning', Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP '06), Sydney, Australia, 120--128.

Dai, W., Yang, Q., Xue, G.-R. and Yu, Y. (2007) 'Boosting for transfer learning', Proceedings of the Twenty-Fourth International Conference (ICML 2007), Corvalis, Oregon, USA, 193-200.

Daumé III, H. (2007) 'Frustratingly Easy Domain Adaptation', Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, Prague, Czech Republic, 256--263.

Daumé III, H., Deoskar, T., McClosky, D., Plank, B. and Tiedemann, J. (2010) Proceedings of the 2010 Workshop on Domain Adaptation for Natural Language Processing, Uppsala, Sweden: Association for Computational Linguistics.

Daumé III, H., Kumar, A. and Saha, A. (2010) 'Frustratingly Easy Semi-Supervised Domain Adaptation', Proceedings of the 2010 Workshop on Domain Adaptation for Natural Language Processing (DANLP 2010), Uppsala, Sweden, 53--59.

Dumais, S., Platt, J., Sahami, M. and Heckerman, D. (1998) 'Inductive Learning Algorithms and Representations for Text Categorization', CIKM'98, 148--155.

Fan, R.-E., Chang, K.-W., Hsieh, C.-J., Wang, X.-R. and Lin, C.-J. (2008) 'LIBLINEAR: A Library for Large Linear Classification', Journal of Machine Learning Research, vol. 9, Aug, pp. 1871--1874.

Gabrilovich, E. and Markovitch, S. (2005) 'Feature Generation for Text Categorization Using World Knowledge', Proceedings of The Nineteenth International Joint Conference for Artificial Intelligence, Edinburgh, Scotland, 1048--1053.

Lang, K. (1995) 'NewsWeeder: Learning to filter netnews', Proceedings of the Twelfth International Conference on Machine Learning, 331-339.

McClosky, D., Charniak, E. and Johnson, M. (2006) 'Reranking and self-training for parser adaptation', Proceedings of ACL-COLING, Sydney, Australia, 337--344.

Miller, G.A. (1995) 'WordNet: a lexical database for English', Commun. ACM, vol. 38, no. 11, November, pp. 39--41, Available: 0001-0782.

Navigli, R. (2009) 'Word sense disambiguation: A survey', ACM Comput. Surv., vol. 41, no. 2, February, pp. 10:1--10:69, Available: 0360-0300.

Pan, S.J. and Yang, Q. (2010) 'A Survey on Transfer Learning ', IEEE Transactions on Knowledge and Data Engineering, vol. 22, no. 10, October, pp. 1345 - 1359.

Petasis, G., Karkaletsis, V., Paliouras, G., Androutsopoulos, I. and Spyropoulos, C.D. (2002) 'Ellogon: A New Text Engineering Platform', LREC 2002, Canary Islands, 72--78.

Shi, X., Fan, W. and Ren, J. (2008) 'Actively Transfer Domain Knowledge', Proceedings of the European conference on Machine Learning and Knowledge Discovery in Databases (ECML PKDD '08) - Part II, Berlin, Heidelberg, 342--357.

Thrun, S. and Pratt, L. (1998) Learning To Learn, Kluwer Academic Publishers.

Tsatsaronis, G., Varlamis, I. and Vazirgiannis, M. (2010) 'Text Relatedness Based on a Word Thesaurus', Journal of Artificial Intelligence Research, vol. 37, pp. 1--39.

Yang, Y. and Liu, X. (1999) 'A Re-Examination of Text Categorization Methods', SIGIR'99, 42--49.

Zhang, J. and Shakya, S.S. (2009) 'Knowledge Transfer for Feature Generation in Document Classification', Proceedings of the 2009 International Conference on Machine Learning and Applications (ICMLA '09), Washington, DC, USA, 255--260.