Text Categorization with Knowledge Transfer from Heterogeneous Data Sources

Rakesh Gupta
Honda Research Institute USA Inc.
800 California Street, Suite 300
Mountain View, CA, USA
rgupta@hra.com

Lev Ratinov*
Department of Computer Science,
University of Illinois
Urbana, IL, USA
ratinov2@uiuc.edu

Abstract

Multi-category classification of short dialogues is a common task performed by humans. When assigning a question to an expert, a customer service operator tries to classify the customer query into one of N different classes for which experts are available. Similarly, questions on the web (for example questions at Yahoo Answers) can be automatically forwarded to a restricted group of people with a specific expertise. Typical questions are short and assume background world knowledge for correct classification.

With exponentially increasing amount of knowledge available, with distinct properties (labeled vs unlabeled, structured vs unstructured), no single knowledge-transfer algorithm such as transfer learning, multi-task learning or self-taught learning can be applied universally. In this work we show that bag-of-words classifiers performs poorly on noisy short conversational text snippets. We present an algorithm for leveraging heterogeneous data sources and algorithms with significant improvements over any single algorithm, rivaling human performance. Using different algorithms for each knowledge source we use mutual information to aggressively prune features. With heterogeneous data sources including Wikipedia, Open Directory Project (ODP), and Yahoo Answers, we show 89.4% and 96.8% correct classification on Google Answers corpus and Switchboard corpus using only 200 features/class. This reflects a huge improvement over bag of words approaches and 48-65% error reduction over previously published state of art (Gabrilovich et. al. 2006).

Introduction

Multi category classification is a common task performed by humans. For example when assigning a question to an expert, a customer service operator tries to classify the customer query into one of N different classes for which experts are available. Similarly questions on the web (for example Yahoo Answers) can be automatically forwarded to a restricted group of people with a specific expertise. These problems can be formulated as a multi-class classification with one caveat. The information in the customer query or the question is not sufficient and assumes background world knowledge for correct classification. For example, the query *My focus doesn't start* assumes the world knowledge that focus is a car model. Humans tend to seamlessly utilize background knowledge coming from different and diverse sources in their classification. We are interested in duplicating this human capability in our work.

External knowledge can be obtained from the web in many flavors: structured and unstructured, labeled and unlabeled. As an example of rapidly growing external labeled data source, we use the Yahoo Answers online data repository, which contains question/answer pairs cataloged by categories and sub-categories. Similarly, Wikipedia is another rapidly growing dataset, which contains articles organized by titles, and perhaps, loosely organized by categories. This exponential increase of available knowledge has motivated work in fields such as semi-supervised learning, multi-task learning, transfer learning and domain adaptation.

A bag of words approach to topic detection (Hearst 1997; Pevzner & Hearst 2002; Galley *et al.* 2003) does not incorporate knowledge beyond words in the sentences. We believe that enriching the bag of words representation with expressive features that capture semantic and statistical knowledge from large external data sources can inject world knowledge into the classifier. Gabrilovich and Markovitch (2005; 2006; 2007) propose a technique for feature generation using a single external source of knowledge like Open Directory Project (ODP) or Wikipedia. Several approaches for feature generation from web search results and extracting statistical regularities in the auxiliary datasets (Sahami & Heilman 2006; Raina *et al.* 2007) have also been proposed in literature. However none of these techniques leverage multiple heterogeneous data sources.

Since the available auxiliary datasets have different properties, no single approach can be applied universally. In this work we present an algorithm for leveraging heterogeneous data sources and algorithms with significant improvements over any single algorithm. Our approach generates expressive features followed by aggressive feature selection. We note that in contrast to research in semi-supervised learning (Nigam *et al.* 2000; Cozman, Cohen, & Cirelo 2003), we do not assume that the external data and the data for the primary classification task come from the same distribution. Neither do we assume that the labeled external resources share a common labels with the primary dataset, an

*This work was done while the second author was an intern at the Honda Research Institute.

assumption commonly made in domain adaptation (Daume & Marcu 2006; Jiang & Zhai 2007). Furthermore, we do not make the common assumption that the training set for the new task is limited. Instead we show that even when thousands of labeled samples are available, external data resources can significantly improve the performance.

In this paper we concentrate on the following three issues. First is generation of features from different knowledge sources. Some of these datasets may be structured (like Open Directory Project) and some may be less structured (like Wikipedia). Second issue is combining features from these diverse datasets. Third, with very large number of external data available, the amount of generated features can be very large and noisy, leading to over-fitting and low performance. We show that feature pruning has a critical impact on the accuracy of the learner. By pruning features based on mutual information, we can get significantly better results with 200-500 features/class. Once the noisy features are removed, the accuracy increases dramatically by up to 110%. We analyze performance of the classifier with variation in the number of features from 1 to over 100,000/class.

The main contribution of our work is a rigorous analysis of the interaction between different feature generation mechanisms based on diverse datasets, when followed by feature selection. To the best of our knowledge, this is the first work that analyses the utility of feature selection in the context of the interaction between features generated through several algorithms and based on several datasets. In the next section we discuss feature generation techniques, followed by description of primary and auxiliary datasets used in our work. We then discuss feature selection followed by results and conclusions.

Feature Generation

Our problem is formulated as follows: given labeled dataset such as from Switchboard corpus and Google Question/Answers, use unlabeled and differently labeled data such as from Yahoo Question/Answers, Wikipedia, and Open Directory Project to improve classification accuracy. Our approach uses three different techniques to generate expressive features, which are later pruned using mutual information. In this section, we describe different feature generation techniques, depending on the type of external data.

Predictions as Features (PAF)

Given a labeled auxiliary dataset D_{aux}, the simplest way to transfer knowledge to the new task is to learn a classifier C_{aux} on D_{aux} and use the predictions of C_{aux} as additional features for data samples of the new task. This approach has been studied extensively in the domain adaptation literature, see for example (Daume 2007). In the knowledge transfer setting, unlike domain adaptation, the labels between the auxiliary and the primary task need not be identical. However we claim that as long as there is a correlation between the labels, knowledge from the auxiliary task can be injected through this correlation. For example, consider a 3-way classification primary dataset D_{prim}, *Travel, DiningOut, BeautyAndStyle*. Assume an auxiliary dataset D_{aux} is available, with documents organized in three categories: *Restaurants, Bars, CarRentals*.

Categories (20)	Sub categories (822)
Society & Culture	Halloween
	Hanukkah
	Languages
	Mythology & Folklore
	Other Cultures & Groups
	Seniors
Arts & Humanities	...
Beauty & Style	...
Business & Finance	
Cars & Transportation	
Computers & Internet	
Consumer Electronics	

Table 1: Sample Yahoo Categories & Sub-Categories

When a classifier C_{aux} is trained on the auxiliary dataset, when encountered with samples from D_{prim}, C_{aux} will tend to predict the label *CarRental* for samples from the *Travel* category and the labels *Restaurants* or *Bars* to the samples from the *DiningOut* category. Therefore adding the predictions of C_{aux} as features in D_{prim} encodes information that is not available in the primary dataset.

This simple approach has a limitation that a single prediction is made on each sample from the primary domain ignoring the prediction confidence. It is often possible to transform the prediction scores of C_{aux} to probabilities and to add only the predictions with assigned probabilities above some threshold. Instead, we use the following feature generation technique, which is efficient in practice.

Given an auxiliary labeled document collection, D_{aux}, for each document $d \in D_{aux}$, a TF-IDF vector representation of d is constructed from D_{aux}. It is important to note that when constructing the TF-IDF vectors for D_{aux}, in contrast to (Gabrilovich & Markovitch 2007), since the target domain is known up front, we use only the words that appear in D_{prim}. For each category c_i^{aux} in the auxiliary domain, its TF-IDF representation is taken to be the centroid of the TF-IDF representations of the documents assigned to that category. Since this representation can be noisy, we truncate it to top k TF-IDF words. Given a primary dataset, D_{prim}, for each document $d \in D_{prim}$, a TF-IDF vector representation of d is constructed from D_{prim}. For each category c_i^{aux} in the auxiliary domain, c_i^{aux} is added as a feature to d if the cosine similarity between the TF-IDF vector representation of d and the truncated version of TF-IDF vector representation c_i^{aux} is above some threshold.

For example, consider the auxiliary Yahoo Answers dataset, organized into 822 sub-categories (sample subcategories are shown in Table 1). The sub-categories in Yahoo Answers are represented by 20 top TF-IDF words appearing in the documents belonging the the subcategory. For each data sample d in the primary dataset, we add 822 binary features f_{ij} with value 1 if d is similar to subcategory j in Yahoo Answers above some threshold. The similarity is defined by cosine similarity of TF-IDF vectors. Features from Open Directory Project (ODP) are generated similarly.

Wikipedia clusters	Yahoo Clusters
churches neighbor wedding window cleaning dining feeding customer neighbors nursing restaurant lawn invitations seek cafe onion activities patients charge diner	recipe festival root flavor calories draft dark spices styles ale wine mild beer brand drink owned company cream steam taste
car insurance cars highway grand muscle driver road roads wheel sport streets recall mile traffic speed drive miles gas driving	bike downhill division france local wine city county district saint pop government village rural spa town map ski arms population
department drive texas california driving license florida virginia georgia district insurance arizona required state laws requirements driver education training minor	earth series rose doctor jane ninth adventure lord stories tells finish novels master fiction story television seventh
students student junior science classes activities grade college writing schools semester score act studying university colleges study senior scores sat	bowie albums tonight apple album beatles love song songs solo guitar pepper lyrics imagine tour bass strawberry track yesterday piano

Table 2: Examples of Wikipedia and Yahoo clusters

Explicit Semantic Analysis (ESA[1])

Explicit Semantic Analysis (ESA), proposed in (Gabrilovich & Markovitch 2007), can be seen as an instance of the approach described in section on PAF. The difference is that each document in the auxiliary dataset D_{aux} is treated as belonging to a distinct label. Thus, this approach operates on thousands (or even millions) of labels, with a single training instance for each label. The disadvantage of this approach is that it can produce many irrelevant features. The advantage is that the auxiliary dataset can be unlabeled. In our work, we used the original ESA interpreter provided to us by its designers. This version of ESA, in contrast to other techniques described in the rest of this section, uses only Wikipedia as its auxiliary dataset.

Structure-Based Feature Generation(SBFG)

ESA generates a very large number of features, for example Wikipedia has over 1 million concepts, each of which can be a potential generated feature (Gabrilovich & Markovitch 2006). While (Gabrilovich & Markovitch 2006) report improved results using this approach, we believe that a technique to capture this information with significantly fewer features is sufficient. We propose clustering of unlabeled data to reduce the number of generated features. This approach enables real time performance, requires much less memory since the clustering is done off-line, and at feature-generation stage considers only a small number of clusters, summarized with top-20 TF-IDF words. For example, if we cluster 10^6 Wikipedia articles into 500 clusters, summarizing each cluster with 20 words, we only need to store 10^4 words in the memory, a reduction by a factor of $2 \cdot 10^3$ compared to the data used by the ESA approach. The increase in computational efficiency is of the same factor.

To reduce the number of features, the documents in the unlabeled auxiliary dataset D_{aux} are clustered without supervision with agglomerative clustering to a prescribed number of clusters[2]. Each cluster is summarized with its top-20 TF-IDF words. Sample clusters are shown in Table 2. Each cluster can be considered as high-level feature f which is is added as an additional feature to the samples in the primary dataset with considerable TF-IDF cosine similarity with f. We note that other techniques, such as principal component analysis (Berry, Dumais, & O'Brien 1995) and sparse coding (Raina *et al.* 2007) can be used for high-level feature extraction.

Auxiliary Datasets

In our work we use three auxiliary datasets: Wikipedia, Open Directory Project and Yahoo Answers. Wikipedia is an online encyclopedia containing 2million concepts (12 GB). To reduce to a manageable size we pruned all but 200k concepts (1 GB) using OpenMind Indoor Common Sense project (Gupta & Kochenderfer 2004). We look at the titles and keep concepts that have a word from OpenMind Indoor Common Sense database. This prunes advanced scientific titles such as *Maxwell-Boltzmann distribution* which are unlikely to occur in everyday conversations. We also prune concepts related to people names, place names, concepts with very long titles (more than 40 words in title), images and redirect pages from Wikipedia.

Yahoo Answers[3] and Open Directory Project[4] are online data directories. Yahoo Answers is a collection of user questions organized into categories and sub-categories. We downloaded around 1K question/answer pairs for each category, accounting for roughly 20,000 question/answer pairs and 300M bytes of data. Open Directory Project is a directory of web sites, again organized into categories and 494 sub-categories. A short textual description next to each link describing the content of the web site is available. We extracted the descriptions of the web sites together with their labels. This resulted in 424MB of data and 4M distinct descriptions. We used 300 Clusters for Yahoo Answers and 500 for Open Directory Project.

[1]We thank Evgeniy Gabrilovich and Shaul Markovitch for providing us the code for the ESA semantic interpreter.

[2]We use the CLUTO clustering software available at http://glaros.dtc.umn.edu/gkhome/views/cluto. Experimentally, good results are achieved with KN clusters, where N is the number of labels in the primary dataset and K is set to 50.

[3]available at: http://answers.yahoo.com/

[4]available at and http://www.dmoz.org

Experimental Design

We evaluated our approach on two datasets: the Switchboard corpus (Godfrey, Holliman, & McDaniel 1992), a benchmark dataset for topic spotting and speech recognition, and the Google Answers dataset[5], a collection of short questions cataloged by topics, which we extracted from the web.

The Google Answers dataset is a collection of 8,850 questions pertaining to 9 top-level categories extracted from the web (around 1000 questions per category). The list of the categories is given in Table 3. Before Google Answers was discontinued, Google required a payment for answering a question which was answered by an expert in Google Answers. Therefore, the questions in this dataset are typically directed to experts, and require a lot of prior knowledge to be correctly categorized. Below are some examples of the questions in the Arts & Entertainment category:

1. *In 1998, Henry Rollins did a spoken word engagement gig in/near Venice beach... i'd like to know the date.*
2. *Please provide general information including best photos of beautiful "antigua town" of Guatemala country.*
3. *Looking for Boy Goergoe manager's phone number.*

The Switchboard corpus is a multi-speaker corpus of conversational speech with about 2500 conversations by 500 speakers from around the US. These conversations are transcribed by speaker turns and span 70 topics (like camping, taxes and recycling). A sample of the transcribed data is given below:

A.5: Uh, do you ha-, are you a musician yourself?
B.6: Uh, well, I sing.
A.7: Uh-huh.
B.8: I dont play an instrument.
A.9: Uh-huh.
Where, do you sing in, in a choir or a choral group?
B.10: Oh, not right now.

The Switchboard dataset was previously used for classifying transcripts in (Myers *et al*. 2000). Following (Myers *et al*. 2000), we manually map selected topics to ten categories as shown in Table 3. We consider the task of classifying each individual speaker turn to one of 10 categories. Our 10 category corpus has 46,000 utterances, with an average of 71.7 speaker turns per conversation. Since many of speaker turns do not carry meaningful information, We filter out stop words and sentences that contain less than 10 words to get 6840 sentences.

We use SVM with a linear kernel as our classification technique [6]. We report averaged results over five fold experiments using 20% of data for testing each time.

Feature Selection

On the Google Answers dataset, the ESA feature expansion algorithm on the Wikipedia dataset generates 98006 features. Yahoo PAF generates 822 features, Open Directory Project generates 494 features, and clustering method generate as many features as the number of clusters. With 11430 BOW features this gives a total of 112056 features. Clearly, to make the classifier more efficient and to avoid over-fitting, feature selection must be performed (Yang & Pedersen 1997). The basic feature classification algorithm is as follows. For each class c we compute the expected mutual information of the feature f and the class c. Mutual Information (MI) is zero if the feature distribution in the collection is same as in class and is maximum when the feature is a perfect indicator of the class. In the information theoretic sense, MI measures how much information the presence/absence of the feature f contributes to making the correct classification decision on c. Formally :

$$MI(f;c) = \sum P(F,C) log \frac{P(F,C)}{P(F)P(C)}$$

where F is a binary random variable that takes the value 1 if sample x contains the feature f and C is a binary random variable that takes the value 1 if x belongs to category c. We use maximum likelihood to estimate these probabilities (Manning, Raghavan, & Schtze 2008).

Table 4 shows top 10 features with highest MI scores for the selected classes in the Google Answers dataset. Surprisingly, using only these 10 features per class, it is possible to achieve 63.39% accuracy for the Google dataset, which is better than using bag-of-words (BOW) approach with best performance at 46.55% with 200 features/class. ESA features shown in green are quite good and intuitive. PAF features generated from Yahoo dataset shown in red and the PAF features generated from ODP features shown in blue. Most interestingly, some of the features do not have an obvious correlation to the category. For example, the Yahoo subcategory *'Entertainment&Music- Jokes'* is not obviously highly correlated to *'Sports and Recreation'* category in Google Answers. However, its top-20 TF-IDF words summarization empirically explains the target category well. We note that MI is a greedy method because in the Health category *cancer* is selected as a feature even though it is highly correlated with *prostate cancer* which is previously selected as a feature and is therefore redundant.

Switchboard categories	**Google categories**
Books (books and literature; magazines)	Sports & Recreation
Fitness (exercise and fitness)	Arts & Entertainment
Movies (movies)	Family & Home
Pets (pets)	Relationships & Society
Sports (football; baseball; basketball)	Business & Money
Family (family life; family reunions; care of the elderly)	Reference & Education & News
Food (recipes; food and cooking)	Health
Music (music)	Computers
Restaurants (restaurants)	Science
Weather (weather; climate)	

Table 3: Switchboard and Google categories

[5] Google Answers *http://answers.google.com/answers/*.

[6] We use the SVM Multi-Class implementation of (Tsochantaridis *et al.* 2004; Crammer & Singer 2001)

Science	health	Sports & Recreation	Business & Money
explosive material	Top Games Card-Games	Entertainment & Music-Horoscopes	Business & Finance-Australia & New Zealand
spacecraft propulsion	Dining Out-Halifax	Top Shopping Tools	Business & Finance-Marketing & Sales
gravity	medicine	Top Shopping Tobacco	Top Business Healthcare
hydrogen	Top Games Dice	Top Reference Maps	Top Business Mining-and-Drilling
water (molecule)	Dining Out-Gualeguaychu	Education&Reference-Primary&Secondary Education	bank
black hole	prostate cancer	Top Shopping Health	citigroup
sewage treatment	hormone replacement therapy (trans)	sports timeline	Top Arts Video
temperature	cancer	DiningOut-Zurich	investment bank
sun	non-hodgkin's lymphoma	Entertainment & Music-Blues	mergers and acquisitions
titanium	Dining Out-Hamburg	Entertainment & Music-Jokes	mutual fund

Table 4: Top 10 features using Mutual Information for Google Answers dataset. Best viewed in color. ESA features are shown green, PAF(Yahoo) features are shown red. PAF(ODP) features are shown blue. No bag-of-words features were selected in this case

Results

As discussed previously, we apply our approach to two datasets: the Switchboard corpus and the Google Answers. Our features come from:

1. **BOW**: bag of words after stop-word removal.
2. **ESA**: Features extracted from Wikipedia using ESA.
3. **PAF(Yahoo)**: Yahoo Answers, organized into 822 sub-categories.
4. **PAF(ODP)**: Open Directory Project, organized into 494 sub-categories.
5. **SBFG(Wiki)**: Pruned Wikipedia articles automatically clustered into 500 clusters.
6. **SBFG(Yahoo)**: Yahoo Answers, automatically clustered into 300 clusters.
7. **SBFG(ODP)**: Open Directory Project, automatically clustered into 500 clusters.

The results from using different feature combinations on the two primary classification datasets are summarized in this section. Figure 1 shows variation of accuracy with number of features/class for the Google Answers dataset. Note that the x axis is on a logarithmic scale. We note that Google dataset has 112052 total features from all techniques with 11430 BOW features. The best results are obtained using top 200-500 features. Using features from all six auxiliary data technique pairs with bag of words (BOW) shows substantial improvements over the baseline technique using only BOW and BOW+ESA techniques. Switchboard dataset with a total of 85703 features from all techniques has similar results.

The performance of different feature generation techniques for top 200 features/class are shown in Table 5. BOW+ESA+PAF refers to all techniques using words and category data from all auxiliary datasets, and BOW+SBFG refers to all techniques using words and clusters based on all auxiliary datasets. Using all six auxiliary data technique pairs with BOW features gives a 64% error reduction in Switchboard dataset and 48% error reduction in the Google Answers dataset over the BOW+ESA(Wiki) baseline technique.

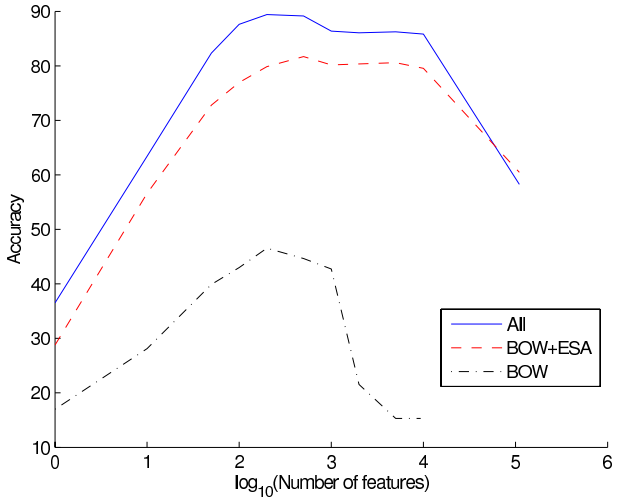

Figure 1: Variation of accuracy with number of features/class. Note that X axis is on log scale

In many applications, such as call placement, the cost of error is high (annoyed customers who reach the wrong extension), therefore, it is common to refuse making an automated decision, and redirect the call to human operator. It is important to increase the precision of the classifier by paying the price of reduced recall (refusal to make automated decision). Figure 2 compares the precision/recall curves for the baseline classifier and our approach for Google Answers dataset. The results indicate that the proposed approach allows a significantly better increase in precision while paying a smaller recall penalty relative to the baseline classifier. Note that the mean F1 score for the classifier trained with the extended features is higher than the baseline classifier.

Conclusions

In this paper we propose a feature-generation approach to knowledge transfer. We combine various sources of unlabeled and labeled data to make human-like decisions that

Algorithm	Accuracy on Switchboard with 200 features/class	Accuracy on Google Answers with 200 features/class
Baseline(BOW)	**68.16%**	**46.55%**
BOW+ESA(Wiki)	**91.04%**	**79.87%**
BOW+SBFG(Wiki)	82.98%	74.71%
BOW+PAF(Yahoo)	79.07%	71.33%
BOW+SBFG(Yahoo)	81.26%	69.98%
BOW+PAF(ODP)	79.79%	73.55%
BOW+SBFG(ODP)	83.70%	78.38%
BOW+ESA+PAF	94.74%	87.38%
BOW+SBFG	91.35%	80.75%
All	**96.88%**	**89.41%**

Table 5: Variation of accuracy with different combination of techniques for 200 features/class

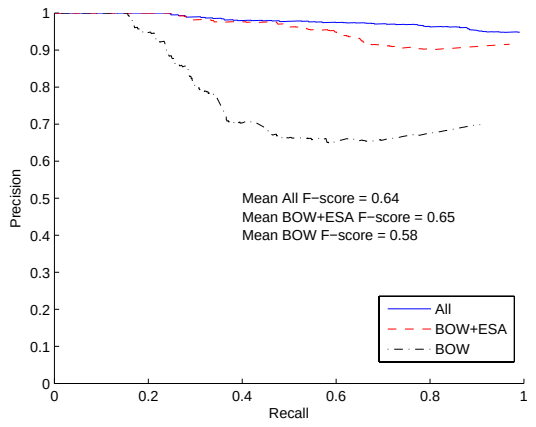

Figure 2: Comparison of the precision/recall tradeoff on the Google Answers dataset for the BOW classifier, BOW+ESA classifier and classifier with all features for 200 features/class.

incorporate world knowledge. We make large scale use of multiple external data sources in a single system. While each individual approach leads to some performance improvement, when large number of features are generated from different external data sources, the effect of features on accuracy is significant.

In addition to combining features from different external data sources, we have analyzed the performance of Mutual Information (MI) as a technique to prune features by two orders of magnitude to improve accuracy as well as the speed of the classifier. We have shown a 48-64% error reduction using heterogenous data sources compared to previous state of art.

References

Berry, M. W.; Dumais, S. T.; and O'Brien, G. W. 1995. Using linear algebra for intelligent information retrieval. *SIAM Review* 37(4):573–595.

Cozman, F. G.; Cohen, I.; and Cirelo, M. C. 2003. Semi-supervised learning of mixture models and bayesian networks. In *ICML-03*.

Crammer, K., and Singer, Y. 2001. On the algorithmic implementation of multiclass kernel-based vector machines. *J. of Machine Learning Research* 2:265–292.

Daume, H., and Marcu, D. 2006. Domain adaptation for statistical classifiers. In *J. Artificial Intelligence, 26:101-126*.

Daume, H. 2007. Frustratingly easy domain adaptation. In *ACL-07*.

Gabrilovich, E., and Markovitch, S. 2005. Feature generation for text categorization using world knowledge. In *IJCAI-05*.

Gabrilovich, E., and Markovitch, S. 2006. Overcoming the brittleness bottleneck using wikipedia: Enhancing text categorization with. In *AAAI-06*.

Gabrilovich, E., and Markovitch, S. 2007. Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *IJCAI-07*.

Galley, M.; McKeown, K.; Fosler-Lussier, E.; and Jing, H. 2003. Discourse segmentation of multi-party conversation. In *ACL-03*.

Godfrey, J. J.; Holliman, E. C.; and McDaniel, J. 1992. Switchboard: telephone speech corpus for research and development. In *ICASSP-92*, volume 1, 517–520.

Gupta, R., and Kochenderfer, M. 2004. Common sense data acquisition for indoor mobile robots. In *AAAI-04*.

Hearst, M. A. 1997. Texttiling: Segmenting text into multi-paragraph subtopic passages. *Computational Linguistics* 23:33–64.

Jiang, J., and Zhai, C. 2007. Instance weighting for domain adaptation in NLP. In *ACL-07*.

Manning, C. D.; Raghavan, P.; and Schtze, H. 2008. *Introduction to Information Retrieval*. Cambridge University Press. chapter 13: Text Classification and Naive Bayes.

Myers, K.; Kearns, M.; Singh, S.; and Walker, M. A. 2000. A boosting approach to topic spotting on subdialogues. In Langley, P., ed., *ICML-00*.

Nigam, K.; McCallum, A. K.; Thrun, S.; and Mitchell, T. M. 2000. Text classification from labeled and unlabeled documents using EM. *Machine Learning* 39(2/3):103–134.

Pevzner, L., and Hearst, M. A. 2002. A critique and improvement of an evaluation metric for text segmentation. *Computational Linguistics* 28(1):19–36.

Raina, R.; Battle, A.; Lee, H.; Packer, B.; and Ng, A. Y. 2007. Self-taught learning: Transfer learning from unlabeled data. In *ICML-07*.

Sahami, M., and Heilman, T. D. 2006. A web-based kernel function for measuring the similarity of short text snippets. In *WWW-06*.

Tsochantaridis, I.; Hofmann, T.; Joachims, T.; and Altun, Y. 2004. Support vector learning for interdependent and structured output spaces. In *ICML-04*.

Yang, Y., and Pedersen, J. O. 1997. A comparative study on feature selection in text categorization. In Fisher, D. H., ed., *ICML-97*.