Legal Docket-Entry Classification: Where Machine Learning stumbles

Ramesh Nallapati and Christopher D. Manning
Natural Language Processing Group
Department of Computer Science
Stanford University
Stanford, CA 94305
{nmramesh,manning}@cs.stanford.edu

Abstract

We investigate the problem of binary text classification in the domain of legal docket entries. This work presents an illustrative instance of a domain-specific problem where the state-of-the-art Machine Learning (ML) classifiers such as SVMs are inadequate. Our investigation into the reasons for the failure of these classifiers revealed two types of prominent errors which we call conjunctive and disjunctive errors. We developed simple heuristics to address one of these error types and improve the performance of the SVMs. Based on the intuition gained from our experiments, we also developed a simple propositional logic based classifier using hand-labeled features, that addresses both types of errors simultaneously. We show that this new, but simple, approach outperforms all existing state-of-the-art ML models, with statistically significant gains. We hope this work serves as a motivating example of the need to build more expressive classifiers beyond the standard model classes, and to address text classification problems in such non-traditional domains.

1 Introduction

Text Classification is a widely researched area, with publications spanning more than a decade (Yang and Liu, 1999). Although earlier models used logic based rules (Apté et al., 1994) and decision trees (Lewis and Ringuette, 1994), recently the emphasis has been on statistical classifiers such as the naive Bayes model (McCallum and Nigam, 1998), logistic regression (Zhang and Oles, 2001) and support vector machines (Joachims, 1998). Although several complex features were considered for classification, eventually researchers have settled down to simple bag-of-words features such as unigrams and some times bigrams (Dumais et al., 1998), thereby completely ignoring the grammar and other semantic information in the text. Despite this fact, the state-of-the-art performance is close to or above 90% on F1 scores on most standard test collections such as Reuters, 20 newsgroups, *etc.* (Bekkerman et al., 2003). As such, most researchers and practitioners believe text classification technology has reached a mature state, where it is suitable for deployment in real life applications.

In this work, we present a text classification problem from the legal domain which challenges some of our understanding of text classification problems. In the new domain, we found that the standard ML approaches using bag-of-words features perform relatively poorly. Not only that, we noticed that the linear form (or even polynomial form) used by these classifiers is inadequate to capture the semantics of the text. Our investigation into the shortcomings of the traditional models such as SVMs, lead us to build a simple propositional logic based classifier using hand-labeled features that outperforms these strong baselines.

Although the new model by itself is interesting, the main objective of our work is to present the text classification community with an interesting problem where the current models are found inadequate. Our hope is that the new problem will encourage researchers to continue to build more sophisticated models to solve classification problems in diverse,

Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing, pages 438–446,
Honolulu, October 2008. ©2008 Association for Computational Linguistics

non-traditional domains.

The rest of the paper is organized as follows. In section 2, we introduce the problem of legal docket entry classification and describe the data with some representative examples. In section 3, we describe the experiments performed with SVMs and several of its variants. We also identify the shortcomings of the current classifiers in this section. In section 3.2, we present results from using human selected features for the classification problem and motivate their application for the docket entry classification using propositional logic in subsection 3.3. We also show that simple propositional logic using human selected features and their labels outperforms the state-of-the-art classifiers. We conclude the discussion in section 4, where we argue the case for more sophisticated classifiers for specialized domains.

2 Docket Entry Classification

In this section, we introduce the problem of legal docket entry classification.

In any US district court of law, information on the chronological events in a case is usually entered in a document called the *case docket*. Each entry in a docket lists an event that occured on a specific date such as pleading, appeal, order, jury trial, judgment, etc. The entries are brief descriptions of the events in natural language. Sometimes, a single docket entry can list multiple events that take place on the same day. Table 1 displays a sample docket for a case.

Identifying various events in a court case is a crucial first step to automatically understanding the progression of a case and also in gathering aggregate statistics of court cases for further analysis. While some events such as "Complaint" may be easy to identify using regular expressions, others are much more complex and may require sophisticated modeling.

In this work, we are primarily interested in identifying one such complex event called "Order re: Summary Judgment". Summary Judgment is a legal term which means that a court has made a determination (a judgment) without a full trial.[1] Such a judgment may be issued as to the merits of an entire case, or of specific issues in that case. Typically, one of the parties (plaintiff or defendant) involved in the case moves a motion for summary judgment, (usually) in an attempt to eliminate the risk of losing a trial. In an "Order re: Summary Judgment" event, the court may grant or deny a motion for summary judgment upon inspecting all the evidence and facts in the case. The task then, is to identify all docket entries in a set of cases that list occurrences of "Order re: Summary Judgment" events. We will call them OSJ events in short.

A few typical positive and negative docket entries for the OSJ event from various cases are shown in table 2. The examples require some explanation. Firstly, all orders granting, denying or amending motions for full or partial summary judgment are considered OSJs. However, if the motion is denied as moot or denied without prejudice, it is not an OSJ event, as shown in the negative examples 1 and 2 in table 2. This is because in such cases, no decision was made on substantive issues of the case. Also, there are other kinds of orders that are issued with reference to a summary judgment motion that do not fall into the category of OSJ, such as negative examples 3 through 9. To elaborate further, negative example 3 is about amending the deadline for filing a summary judgment motion, but not a summary judgment motion itself. Likewise, in negative example 4, the judge denies a motion to shorten time on a motion to vacate the order on summary judgment, but not the motion on summary judgment itself. The other negative examples are very similar in spirit and we leave it as an exercise to the reader to interpret why they are negatively labeled.

On first glance, it appears that a standard classifier may do a good job on this data, since the classification seems to depend mostly on certain key words such as 'granting', 'denying', 'moot', etc. Also notice that some of the docket entries contain multiple events, but as long as it contains the 'order re: summary judgment' event, it falls into the positive class. This seems very similar to the standard case, where a document may belong to multiple topics, but it is still identified as on-topic by a binary classifier on the corresponding topic.

Hence, as a first step, we attempted using a standard SVM classifier.

[1] See *e.g.,* Wikipedia for more information: http://en.wikipedia.org/wiki/Summary_judgment

#	Date Filed	Text
1	10/21/2002	Original Complaint with JURY DEMAND filed. Cause: 35:271 Patent Infringement Modified on 10/24/2002 (Entered: 10/22/2002)
2	10/21/2002	Form mailed to Commissioner of Patents and Trademarks. (poa)
3	10/28/2002	Return of service executed as to Mathworks Inc 10/23/02 Answer due on 11/12/02 for Mathworks Inc (poa) (Entered: 10/28/2002)
4	11/4/2002	Unopposed Motion by Mathworks Inc The to extend time to answer or otherwise respond to pla's complaint (ktd) (Entered: 11/05/2002)
5	11/5/2002	ORDER granting [4-1] motion to extend time to answer or otherwise respond to pla's complaint, ans reset answer due on 11/27/02 for Mathworks Inc
. . .	. . .	. . . . . .

Table 1: An example (incomplete) docket: each row in the table corresponds to a docket-entry

2.1 Data

We have collected 5,595 docket entries from several court cases on intellectual property litigation, that are related to orders pertaining to summary judgment, and hand labeled them into OSJ or not OSJ categories.[2] The hand-labeling was done by a single legal expert, who practised law for a number of years. In all, 1,848 of these docket entries fall into the OSJ category.

In all our experiments, we split the entire data randomly into 20 disjoint subsets, where each set has the same proportion of positive-to-negative examples as the original complete set. For all the classifiers we used in this work, we performed 20-fold cross validation. We compute F1 scores on the held-out data of each run and report overall F1 score as the single point performance measure. We also perform statistical significance tests using the results from the 20 cross-validation runs.

2.2 Preprocessing

Before we ran our classifiers, we removed all punctuation, did casefolding, removed stopwords and stemmed the words using the Porter stemmer. We used unigrams and bigrams as our basic features.[3] We considered all the words and bigrams as binary features and did not use any TF-IDF weighting. Our justification for this decision is as follows: the docket text is typically very short and it is usually rare to see the same feature occurring multiple times in a docket entry. In addition, unlike in standard text classification, some of the features that are highly frequent across docket entries such as 'denying','granting', etc., are also the ones that are highly discriminative. In such a case, down-weighting these features using IDF weights might actually hurt performance. Besides (Dumais et al., 1998) found that using binary features works as well as using TF-IDF weights.

In addition, we also built a domain specific sentence boundary detector using regular expressions.[4] For constructing the features of a docket entry, we only consider those sentences in the entry that contain the phrase "summary judgment" and its variants.[5] Our preliminary experiments found that this helps the classifier focus on the relevant features, helping it to improve precision while not altering its recall noticeably.

3 Experiments and results

3.1 Basic SVM

First we implemented the standard linear SVM[6] on this problem with only word-based features (unigrams and bigrams) as the input. Quite surprisingly, the model achieves an F1 score of only 79.44% as shown in entry 1 of table 5. On inspection, we no-

[2] The data can be made available free of cost upon request. Please email the first author for more information.

[3] In our preliminary experiments, we found that a combination of unigrams and bigrams works better than unigrams alone.

[4] It works well in most cases but is far from perfect, due to the noisy nature of the data.

[5] The variants include "sum jgm", "S/J", "summary adjudication", "summary jgm", etc.

[6] All our SVM experiments were performed using the libsvm implementation downloadable from http://www.csie.ntu.edu.tw/∼cjlin/libsvm/

REPRESENTATIVE POSITIVE EXAMPLES

1. ORDER denying [36-1] motion for summary judgment on dfts Ranbaxy invalidity defenses by pltfs. (signed by Judge Garrett E. Brown, Jr.)
2. ORDER GRANTING IN PART AND DENYING IN PART DEFENDANTS' MOTION FOR SUMMARY JUDGMENT
3. ORDER re 78 MOTION to Amend/Correct Motion for Summary Judgment and supporting documents, filed by Defendant Synergetics USA, Inc. ; ORDERED GRANTED.
4. MEMORANDUM AND ORDER re: 495 Third MOTION for Partial Summary Judgment Dismissing Monsanto's Defenses Related to Dr. Barnes filed by Bayer BioScience N.V., motion is GRANTED IN PART AND DENIED IN PART.
5. ORDER GRANTING IN PART PLTF S/J MOT; GRANTING IN PART PLTF MOT/CLARIFY; GRANTING DEFT MOT/CLARIFY; PRTL S/J STAYED.
6. ORDER by Chief Judge Joe B. McDade. Court is granting in part and denying in part Deere's motion for reconsideration and clarification [42-2]; granting Toro's motion for summary judgment of non-infringement [45-1]; denying Deere's motion for summary judgment [58-1];
7. ORDER GRANTING DEFT. MOTION FOR S/J AND DENYING PLTF. MOTIONS FOR S/J AND TO SUPPLEMENT.

REPRESENTATIVE NEGATIVE EXAMPLES

1. ORDER - denying w/out prejudice 17 Motion for Summary Judgment, denying w/out prejudice 49 Motion to Amend/Correct . Signed by Judge Kent A. Jordan on 1/23/06.
2. Order denying as moot motion for summary judgment.
3. Order granting 53 Motion to Amend/Correct the deadline for filing summary jgm motions will be moved 12/1/03 to 12/8/03
4. ORDER by Judge Claudia Wilken denying plaintiff's motion to shorten time on motion to vacate portions of Court's order on cross-motion for summary judgment on patent issues [695-1] [697-1]
5. MEMORANDUM AND ORDER: by Honorable E. Richard Webber, IT IS HEREBY ORDERED that Defendant Aventis shall have 10 days from the date of this order to demonstrate why the Court should not grant summary judgment to Monsanto of non-infringement of claims 1-8 and 12 of the '565 patent and claim 4 of the '372 patent.
6. ORDER by Judge Claudia Wilken DENYING motion for an order certifying for immediate appeal portions of the courts' 2/6/03 order granting in part plaintiff's motion for partial summary judgment [370-1]
7. ORDER by Judge William Alsup denying in part 12 Motion to Consolidate Cases except as to one issue, granting in part for collateral estoppel 20 Motion for Summary Judgment
8. ORDER (Chief Mag. Judge Jonathan G. Lebedoff / 9/11/02) that the court grants Andersen's motion and orders that Andersen be allowed to bring its motions for summary judgment
9. ORDER by Judge Susan J. Dlott denying motion to strike declaration of H Bradley Hammond attached to memorandum in opposition to motion for partial summary judgment as to liability on the patent infringement and validity claims [40-1] [47-1] [48-1]

Table 2: Order: re Summary Judgment: positive and negative docket entries. The entries are reproduced as they are.

ticed that the SVM assigns high weights to many spurious features owing to their strong correlation with the class.

As a natural solution to this problem, we selected the top 100 features[7] using the standard information gain metric (Yang and Pedersen, 1997) and ran the SVM on the pruned feature set. As one would expect, the performance of the SVM improved significantly to reach an F1 score of 83.08% as shown in entry 2 of the same table. However, it is still a far cry from the typical results on standard test beds where the performance is above 90% F1. We suspected that training data was probably insufficient, but a learning curve plotting performance of the SVM as a function of the amount of training data reached a plateau with the amount of training data we had, so this problem was ruled out.

To understand the reasons for its inferior performance, we studied the features that are assigned the highest weights by the classifier. Although the SVM is able to assign high weights to several discriminative features such as 'denied', and 'granted', it also assigns high weights to features such as 'opinion', 'memorandum', 'order', 'judgment', etc., which have high co-occurrence rates with the positive class, but are not very discriminative in terms of the actual classification.

This is indicative of the problems associated with standard feature selection algorithms such as information gain in these domains, where high correlation with the label does not necessarily imply high discriminative power of the feature. Traditional classification tasks usually fall into what we call the 'topical classification' domain, where the distribution of words in the documents is a highly discriminative feature. On such tasks, feature selection algorithms based on feature-class correlation have been very successful. In contrast, in the current problem, which we call 'semantic classification', there seem to be a fixed number of domain specific operative words such as 'grant', 'deny', 'moot', 'strike', etc., which, almost entirely decide the class of the docket entry, irrespective of the existence of other highly correlated features. The information gain metric as well as the SVM are not able to fully capture such features in this problem.

We leave the problem of accurate feature selection to future work, but in this work, we address the issue by asking for human intervention, as we describe in the next section. One reason for seeking human assistance is that it will give us an estimate of upperbound performance of an automatic feature selection system. In addition, it will also offer us a hint as to whether the poor performance of the SVM is because of poor feature selection. We will aim to answer this question in the next section.

3.2 Human feature selection

Using human assistance for feature selection is a relatively new idea in the text classification domain. (Raghavan et al., 2006) propose a framework in which the system asks the user to label documents and features alternatively. They report that this results in substantial improvement in performance especially when the amount of labeled data is meagre. (Druck et al., 2008) propose a new Generalized Expectation criterion that learns a classification function from labeled features alone (and no labeled documents). They showed that feature labeling can reduce annotation effort from humans compared to document labeling, while achieving almost the same performance.

Following this literature, we asked our annotators to identify a minimal but definitive list of discriminative features from labeled data. The annotators were specifically instructed to identify the features that are most critical in tagging a docket entry one way or the other. In addition, they were also asked to assign a polarity to each feature. In other words, the polarity tells us whether or not the features belong to the positive class. Table 3 lists the complete set of features identified by the annotators.

As an obvious next step, we trained the SVM in the standard way, but using only the features from table 3 as the pruned set of features. Remarkably, the performance improves to 86.77% in F1, as shown in entry 3 of table 5. Again, this illustrates the uniqueness of this dataset, where a small number of hand selected features (< 40) makes a huge difference in performance compared to a state-of-the-art SVM combined with automatic feature selection. We believe this calls for more future work in improving feature selection algorithms.

[7] We tried other numbers as well, but top 100 features achieves the best performance.

Label	Features
Positive	grant, deny, amend, reverse, adopt, correct, reconsider, dismiss
Negative	strike, proposed, defer, adjourn, moot, exclude, change, extend, leave, exceed, premature, unseal, hearing, extend, permission, oral argument, schedule, ex parte, protective order, oppose, without prejudice, withdraw, response, suspend, request, case management order, to file, enlarge, reset, supplement placing under seal, show cause reallocate, taken under submission

Table 3: Complete set of hand-selected features: morphological variants not listed

Notice that despite using human assistance, the performance of the SVM is still not at a desirable level. This clearly points to deficiencies in the model other than poor feature selection. To understand the problem, we examined the errors made by the SVM and found that there are essentially two types of errors: *conjunctive* and *disjunctive*. Representative examples for both kinds of errors are displayed in table 4. The first example in the table corresponds to a conjunctive error, where the SVM is unable to model the binary switch like behavior of features. In this example, although ‘deny’ is rightly assigned a positive weight and ‘moot’ is rightly assigned a negative weight, when both features co-occur in a docket entry (as in ‘deny as moot’), it makes the label negative.[8] However, the combined weight of the linear SVM is positive since the absolute value of the weight assigned to ‘deny’ is higher than that of ‘moot’, resulting in a net positive score. The second example falls into the category of disjunctive errors, where the SVM fails to model disjunctive behavior of sentences. In this example, the first sentence contains an OSJ event, but the second and third sentences are negatives for OSJ. As we have discussed earlier, this docket entry belongs to the OSJ category since it contains at least one OSJ event. However, we see that the negative weights assigned by the SVM to the second and third sentences result in an overall negative classification.

As a first attempt, we tried to reduce the conjunctive errors in our system. Towards this objective, we built a decision tree[9] using the same features listed in table 3. Our intuition was that a decision tree makes a categorical decision at each node in the tree, hence it could capture the binary-switch like behavior of features. However, the performance of the decision tree is found to be statistically indistinguishable from the linear SVM as shown in entry 4 of table 5. As an alternative, we used an SVM with a quadratic kernel, since it can also capture such pairwise interactions of features. This resulted in a fractional improvement in performance, but is again statistically indistinguishable from the decision tree. We also tried higher order polynomial kernels and the RBF kernel, but the performance got no better.[10] It is not easy to analyze the behavior of non-linear kernels since they operate in a higher kernel space. Our hypothesis is that polynomial functions capture higher order interactions between features, but they do not capture conjunctive behavior precisely.

As an alternative, we considered the following heuristic: whenever two or more of the hand selected features occur in the same sentence, we merged them to form an n-gram. The intuition behind this heuristic is the following: using the same example as before, if words such as ‘deny’ and ‘moot’ occur in the same sentence, we form the bigram ‘deny-moot’, forcing the SVM to consider the bigram as a separate feature. We hope to capture the conjunctive behavior of some features using this heuristic. The result of this approach, as displayed in entry 6 of table 5, shows small but statistically significant improvement over the quadratic SVM, confirming our theory. We also attempted a quadratic kernel using sentence level n-grams, but it did not show any improvement.

Note that all the models and heuristics we used above only address conjunctive errors, but not disjunctive errors. From the discussion above, we suspect the reader already has a good picture of what

[8] This is very similar to the conjunction of two logical variables where the conjunction of the variables is negative when at least one of them is negative. Hence the name conjunctive error.

[9] We used the publicly available implementation from www.run.montefiore.ulg.ac.be/∼francois/software/jaDTi/

[10] We also tried various parameter settings for these kernels with no success.

1. DOCKET ENTRY: order denying as moot [22-1] motion for summary judgment (signed by judge federico a. moreno on 02/28/06).
   FEATURES (WEIGHTS): denying (1.907), moot (-1.475)
   SCORE: 0.432; TRUE LABEL: Not OSJ; SVM LABEL: OSJ

2. DOCKET ENTRY: order granting dfts' 37 motion for summary judgment. further ordered denying as moot pla's cross-motion 42 for summary judgment. denying as moot dfts' motion to strike pla's cross-motion for summary judgment 55 . directing the clerk to enter judgment accordingly. signed by judge mary h murguia on 9/18/07
   FEATURES (WEIGHTS): granting (1.64), denying (3.57), strike(-2.05) moot(-4.22)
   SCORE: -1.06; TRUE LABEL: OSJ; SVM LABEL: Not OSJ

Table 4: Representative examples for conjunctive and disjunctive errors of the linear SVM using hand selected features

an appropriate model for this data might look like. The next section introduces this new model developed using the intuition gained above.

3.3 Propositional Logic using Human Features and Labels

So far, the classifiers we considered received a performance boost by piggybacking on the human selected features. However, they did not take into account the polarity of these features. A logical next step would be to exploit this information as well. An appropriate model would be the generalized expectation criterion model by (Druck et al., 2008) which learns by matching model specific label expectations conditioned on each feature, with the corresponding empirical expectations. However, the base model they use is a logistic regression model, which is a log-linear model, and hence would suffer from the same limitations as the linear SVM. There is also other work on combining SVMs with labeled features using transduction on unlabeled examples, that are soft-labeled using labeled features (Wu and Srihari, 2004), but we believe it will again suffer from the same limitations as the SVM on this domain.

In order to address the conjunctive and disjunctive errors simultaneously, we propose a new, but simple approach using propositional logic. We consider each labeled feature as a propositional variable, where true or false corresponds to whether the label of the feature is positive or negative respectively. Given a docket entry, we first extract its sentences, and for each sentence, we extract its labeled features, if present. Then, we construct a sentence-level formula formed by the conjunction of the variables representing the labeled features. The final classifier is a disjunction of the formulas of all sentences in the docket entry. Formally, the propositional logic based classifier can be expressed as follows:

$$C(D) = \vee_{i=1}^{N(D)} (\wedge_{j=1}^{M_i} L(f_{ij})) \tag{1}$$

where D is the docket entry, $N(D)$ is its number of sentences, M_i is the number of labeled features in the i^{th} sentence, f_{ij} is the j^{th} labeled feature in the i^{th} sentence and $L()$ is a mapping from a feature to its label, and $C(D)$ is the classification function where 'true' implies the docket entry contains an OSJ event.

The propositional logic model is designed to address the within-sentence conjunctive errors and without-sentence disjunctive errors simultaneously. Clearly, the within-sentence conjunctive behavior of the labeled features is captured by applying logical conjunctions to the labeled features within a sentence. Similarly, the disjunctive behavior of sentences is captured by applying disjunctions to the sentence-level clauses. This model requires no training, but for reasons of fairness in comparison, at testing time, we used only those human features (and their labels) that exist in the training set in each cross-validation run. The performance of this new approach, listed in table 5 as entry 7, is slightly better than the best performing SVM in entry 6. The difference in performance in this case is statistically significant, as measured by a paired, 2-tailed t-test at 95% confidence level (p-value = 0.007).

Although the improvement for this model is statistically significant, it does not entirely match our

#	Model	Recall (%)	Precision (%)	F1 (%)
1	Linear SVM with uni/bigrams only	75.19	84.21	79.44
2	Linear SVM with uni/bigrams only FS100	82.47	83.69	83.08*
3	Linear SVM with HF only	84.68	88.97	86.77*
4	Decision Tree with HF only	85.22	89.38	87.25
5	Quadratic SVM with HF only	84.14	90.98	87.43
6	Linear SVM with HF sentNgrams	84.63	93.37	88.78*
7	Propositional Logic with HF and their labels	**85.71**	**93.45**	**89.67***

Table 5: Results for 'Order re: Summary Judgment': FS100 indicates that only top 100 features were selected using Information Gain metric; HF stands for human built features, sentNgrams refers to the case where all the human-built features in a given sentence were merged to form an n-gram feature. A '*' next to F1 value indicates statistically significant result compared to its closest lower value, measured using a paired 2-tailed T-test, at 95% confidence level. The highest numbers in each column are highlighted using boldface.

expectations. Our data analysis showed a variety of errors caused mostly due to the following issues:

- *Imperfect sentence boundary detection:* since the propositional logic model considers sentences as strong conjunctions, it is more sensitive to errors in sentence boundary detection than SVMs. Any errors would cause the model to form conjunctions with features in neighboring sentences and deliver an incorrect labeling.

- *Incomplete feature set:* Some errors are caused because the feature set is not complete. For example, negative example 4 in table 2 is tagged as positive by the new model. This error could have been avoided if the word 'shorten' had been identified as a negative feature.

- *Relevant but bipolar features:* Although our model assumes that the selected features exhibit binary nature, this may not always be true. For example the word *allow* is sometimes used as a synonym for 'grant' which is a positive feature, but other times, as in negative example 8 in table 2, it exhibits negative polarity. Hence it is not always possible to encode all relevant features into the logic based model.

- *Limitations in expressiveness:* Some natural language sentences such as negative example 5 in table 2 are simply beyond the scope of the conjunctive and disjunctive formulations.

4 Discussion and Conclusions

Clearly, there is a significant amount of work to be done to further improve the performance of the propositional logic based classifier. One obvious line of work is towards better feature selection in this domain. One plausible technique would be to use shallow natural language processing techniques to extract the operative verbs acting on the phrase "summary judgment", and use them as the pruned feature set.

Another potential direction would be to extend the SVM-based system to model disjunctive behavior of sentences.[11] One way to accomplish this would be to classify each sentence individually and then to combine the outcomes using a disjunction. But for this to be implemented, we would also need labels at the sentence level during training time. One could procure these labels from annotators, but as an alternative, one could learn the sentence-level labels in an unsupervised fashion using a latent variable at the sentence level, but a supervised model at the docket-entry level. Such models may also be appropriate for traditional document classification where each document could be multi-labeled, and it is something we would like attempt in the future.

In addition, instead of manually constructing the logic based system, one could also automatically learn the rules by using ideas from earlier work on ILP (Muggleton, 1997), FOIL (Quinlan and Cameron-Jones, 1993), etc.

[11] Recall that the heuristics we presented for SVMs only address the conjunctive errors.

To summarize, we believe it is remarkable that a simple logic-based classifier could outperform an SVM that is already boosted by hand picked features and heuristics such as sentence level n-grams. This work clearly exposes some of the limitations of the state-of-the-art models in capturing the intricacies of natural language, and suggests that there is more work to be done in improving the performance of text based classifiers in specialized domains. As such, we hope our work motivates other researchers towards building better classifiers for this and other related problems.

Acknowledgments

The authors would like to thank Stanford university for financial support.

References

Chidanand Apté, Fred Damerau, and Sholom M. Weiss. 1994. Automated learning of decision rules for text categorization. *ACM Trans. Inf. Syst.*, 12(3):233–251.

R. Bekkerman, R. El-Yaniv, N. Tishby, and Y. Winter. 2003. Distributional word clusters vs. words for text categorization. *Journal of Machine Learning Research*.

Gregory Druck, Gideon Mann, and Andrew McCallum. 2008. Learning from labeled features using generalized expectation criteria. In *Proceedings of ACM Special Interest Group on Information Retreival, (SIGIR)*.

Susan Dumais, John Platt, David Heckerman, and Mehran Sahami. 1998. Inductive learning algorithms and representations for text categorization. In *CIKM '98: Proceedings of the seventh international conference on Information and knowledge management*, pages 148–155, New York, NY, USA. ACM.

T. Joachims. 1998. Text categorization with support vector machines: Learning with many relevant features. In *ECML-98: 10th European Conference on Machine Learning*.

D.D. Lewis and M. Ringuette. 1994. Comparison of two learning algorithms for text categorization. In *Third Annual Symposium on Document Analysis and Information Retrieval*.

A. McCallum and K. Nigam. 1998. A comparison of event models for Naïve Bayes text classification . In *AAAI-98 Workshop on Learning for Text Categorization*.

Stephen Muggleton. 1997. *Inductive Logic Programming: 6th International Workshop: Seleted Papers*. Springer.

J.R. Quinlan and R.M. Cameron-Jones. 1993. Foil: a mid-term report. In *Proceedings of European Conference on Machine Learning*.

Hema Raghavan, Omid Madani, and Rosie Jones. 2006. Active learning with feedback on features and instances. *J. Mach. Learn. Res.*, 7:1655–1686.

Xiaoyun Wu and Rohini Srihari. 2004. Incorporating prior knowledge with weighted margin support vector machines. In *KDD '04: Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 326–333, New York, NY, USA. ACM.

Yiming Yang and Xin Liu. 1999. A re-examination of text categorization methods. In *SIGIR '99: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 42–49, New York, NY, USA. ACM.

Yiming Yang and Jan O. Pedersen. 1997. A comparative study on feature selection in text categorization. In *ICML '97: Proceedings of the Fourteenth International Conference on Machine Learning*, pages 412–420, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Tong Zhang and Frank J. Oles. 2001. Text categorization based on regularized linear classification methods. *Information Retrieval*, 4(1):5–31.