

# Classifying Legal Questions into Topic Areas Using Machine Learning

**Brian Lao**  
Stanford University  
bjlao@stanford.edu

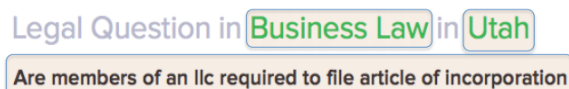
**Karthik Jagadeesh**  
Stanford University  
kjag@stanford.edu

## Abstract

In this paper we describe the steps taken to build a machine learning classifier that successfully classifies legal questions into the most relevant practice area. We have created 16 different general categories that legal questions fall into. Categorizing legal questions into the correct practice area has many useful applications such as facilitating improved realtime feedback, information retrieval, relevant lawyer recommendations, and responses to users asking questions on Q&A websites.

## 1 Introduction

We built a multiclass classification model for categorizing everyday user's legal questions into the relevant legal topic area. Application of machine learning to the legal domain remains a relatively new task.[1] With the increasing ubiquity of the internet, individuals are looking more to internet resources to find relevant attorneys and to obtain answers to their legal questions. Legal Q&A websites exist where users ask a question and are given the choice of tagging their question with the relevant topic area. For instance, a user's legal question may be, "How do I file for divorce?" The relevant legal area tag would be "Family Law." However, people sometimes lack the legal expertise to select the topic areas most relevant to their question. Thus, classifying legal questions into their associated topic area would facilitate: (1) the retrieval of more relevant information and answers to educate the user; and (2) improved recommendations of relevant lawyers with expertise in the particular topic area. We conducted multiclass classification with 5 models: (1) logistic regression (Logistic); (2) multinomial Naive Bayes (MNB); (3) linear SVM with Newton method (SVM (Newton)); (4) SVM with stochastic gradi-



Legal Question in Business Law in Utah  
Are members of an llc required to file article of incorporation

Figure 1: We used Apache Nutch to scrape 200,000 user-asked legal questions, the legal categories associated with each question, and the locations in which the questions were asked.

ent descent (SVM (SGD)); and (5) one-layer neural network (1-Layer NN). Linear SVM with Newton method performed the best with a 69.1% test accuracy in multiclass classification.

## 2 Data

In order to successfully build a supervised learning model to categorize legal questions, we needed a large dataset of user-asked legal questions that have already been labeled with the correct practice area. Legal Q&A websites contain publicly released user-questions that have been manually labeled - by users or lawyers - with the relevant topic area. We used the Apache Nutch Web Crawler (<http://nutch.apache.org/>) to crawl the following legal websites: (1) [avvo.com/legal-answers/](http://avvo.com/legal-answers/); (2) [ask-a-lawyer.freeadvice.com](http://ask-a-lawyer.freeadvice.com); (3) [lawguru.com/legal-questions](http://lawguru.com/legal-questions). We obtained 200,000 user-asked legal questions, the associated legal area tags, and the locations in which the questions were asked. Figure 1 shows an example of the different components of a legal question that we scraped using Apache Nutch.

A typical example of a legal question may include (a) two sentences of background about the user's situation and (b) a sentence containing the actual question such as "What legal remedies do I have after being hit by a negligent car driver?" The associated tag for this example would be "Personal Injury Law" and the location would be a U.S. state such as "California." Although the questions were

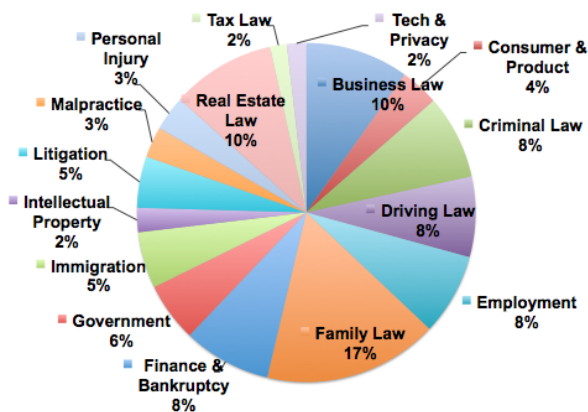


Figure 2: Distribution of our 16 legal category labels after preprocessing the data

often short in length, the relatively large number of documents facilitated the training of a model that could predict the legal area with relatively high accuracy. The rows in our raw data file are individual questions, and the columns include (1) the legal question, (2) associated topic area, and (3) associated location.

### 3 Features

#### 3.1 Data Preprocessing

For the legal Q&A websites that we scraped, each website used different vocabulary for their topic area labels. For instance, Avvo has the legal categories, "DUI" and "Speeding Ticket," while FreeAdvice has the legal category "Drunk Driving." To create a unified set of topic area labels for the legal questions, we created our own set of 16 practice areas. We then used our legal knowledge to manually map the legal Q&A websites' different legal categories into our own categories. For example, 1 of our 16 practice areas is "Driving Law," which encompasses Avvo's "DUI" and "Speeding Ticket" categories, FreeAdvice's "Drunk Driving" category, and other categories related to driving law. Our set of practice labels and the labels' distribution amongst the legal questions can be seen in Figure 2.

We further processed the legal questions by (1) removing stop words, (2) stemming words with the Porter stemmer, and (3) lower-casing all words in the questions.

#### 3.2 Word Unigram Features

This is the most basic feature vector used for any text based model. We look at each unique word

as a feature, and created a feature vector consisting of the counts of how often each word appears in the question. The intuition behind using this feature vector is that questions with a shared topic area will contain similar words that will allow us to discriminate between legal topic areas. For example, the category "Intellectual Property" will likely have a high proportion of questions that use the word "Patent."

#### 3.3 Word Bi-gram Features

While the unigram feature vector can be effective for certain tasks, it is often useful to look at the context of a word in a sentence. One straightforward method of capturing the context of a word is to look at its surrounding words. Rather than keeping track of individual word counts, the bi-gram feature vector keeps track of the number of times that pairs of words appear in a question. Intuitively, bi-grams allows us to capture the discriminative power in two-word terms such as "home accident," which would likely be associated with the legal area of "Personal Injury Law." If only unigram features are used, "home" could be associated with "Real Estate Law" for instance. By combining both unigram and bi-gram features, a much larger feature space is obtained, resulting in a sparser set of features.

#### 3.4 TF-IDF weighting

We use Term Frequency - Inverse Document Frequency (TF-IDF) to give weights to unigrams and bigrams based on their relative importance to our corpus of legal questions. In the context of our task, TF-IDF will give high importance to legal words such as "annulment" that may show up with high frequency within an individual "Family Law" question, but may not be common in the context of the entire corpus. Legal terms are often topic area-specific, and we want to ensure that our features capture the discriminative power of the legal terms being used.

#### 3.5 Word2Vec features

We used a pretrained set of word vector features from the Word2Vec project. This system contains a vector for each word in the dictionary and uses a neural network trained on New York Times to provide a context-relevant vector representation of the word. We represent a paragraph by looking at the average of values of the word vectors in the document.

## 4 Models

In order to classify legal questions into 1 of the 16 legal categories, we experimented with 5 different models and evaluated their label classification abilities.

### 4.1 Logistic Regression

This logistic regression model is a probabilistic classifier using the stochastic gradient descent model of learning.

### 4.2 Multinomial Naive Bayes [2]

This MNB model uses Laplace smoothing for maximum likelihood of parameters where we used a Laplace smoothing parameter  $\alpha = 1$ .

$$\theta_{yi} = \frac{N_{yi} + \alpha}{N_{y\cdot} + \alpha \times n}$$

The standard Naive Bayes model is useful when features have values of 0 or 1 (binary). The model described here allows us to generalize and use a Naive Bayes approach for features that take on values of  $0 \dots k$ .

Due to some of the assumptions made in this model, skewed training data will result in a shift of the weights to the biased classes. This is especially problematic for us since there is a large difference in # of samples in each class.

### 4.3 Linear SVM [3]

This SVM model solves the optimization problem:  $\min_w \frac{1}{2} w^T w + C \sum_{i=1}^l \xi(w; x_i, y_i)$  where we used the L2-SVM loss function  $\max(1 - y_i w^T x_i, 0)^2$  and a penalty parameter  $C=0.1$ .

### 4.4 SVM with Stochastic Gradient Descent

This SVM model uses a hinge loss function of  $l(y) = \max(0, 1 - ty)$ , a L2-penalty, and employs the SGD algorithm,  $w = w - \alpha \delta Q_i(w)$

### 4.5 One-layer Neural Network

We designed a neural network [4] with 35K input units, 100 hidden units, and a softmax function for the output layer:

$$\sigma(z)_j = \frac{e^{z_j}}{\sum_{k=1}^K e^{z_k}} \text{ for } j = 1, \dots, K.$$

This model is able to learn complex patterns if a large training data set is available. It is especially useful when we are trying to learn from a sparse feature space.

With out current implementation of Neural Networks, the computational time is extremely slow given that we have on the order of 100K input

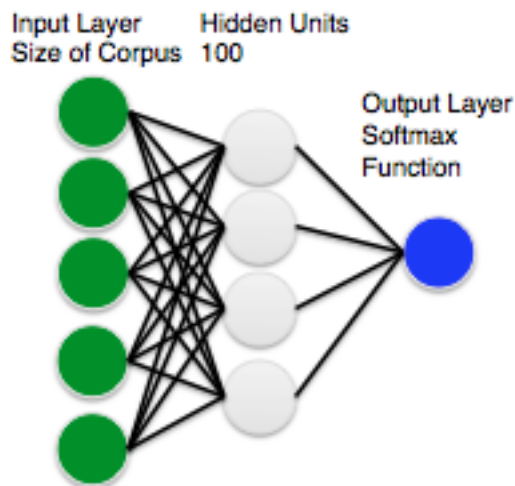


Figure 3: Overall 1 Layer Neural Network Architecture. When we are using word frequency and TF-IDF feature vectors, the # of input units is on the order of 50K.

units. One solution to this problem can be to implement and run the neural network on a GPU.

## 5 Results

We conducted 4-fold cross-validation on our data set, resulting in our SVM model with Newton implementation performing the best in multiclass classification with a 69.1% test accuracy (See Figures 4 and 5).

The "Immigration" and "Driving Law" categories possessed the best F1 scores, as seen in Figure 6.

## 6 Discussion

### 6.1 General

Given that SVMs have performed well in many text classification applications[4], we were not surprised by the Linear SVM model performing the best with a promising 69.1% test accuracy. Low F1-scores for the topic areas of "Personal Injury," "Litigation," and "Government" brought down the overall accuracy. Poor performance in these topics may be explained by these topics having overlap with other topics, where, for example, a classifier may have trouble categorizing a car accident question because of the question's relevance to both the "Personal Injury" and "Driving Law" categories. In the future, for the F1 scores for individual categories, we plan on obtaining F1

	<u>Training Set</u>	<u>Test Set</u>	<u>Training Accuracy</u>	<u>Test Accuracy</u>
Logistic	~150K	~50K	89.7%	67.9%
MNB	~150K	~50K	79.6%	59.7%
SVM (Newton)	~150K	~50K	99.8%	69.1%
SVM (SGD)	~150K	~50K	93.6%	65.6%
Neural (Word Vectors)	~150K	~50K	72.6%	49%
Neural (word2vec Vectors)	~150K	~50K	92.3%	68%

Figure 4: Training and test accuracy for our 5 models.

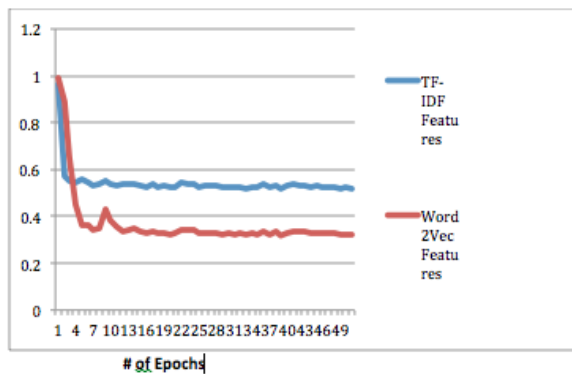


Figure 5: Error rate over 50 epochs starts to stabilize around 32% after 10 epochs

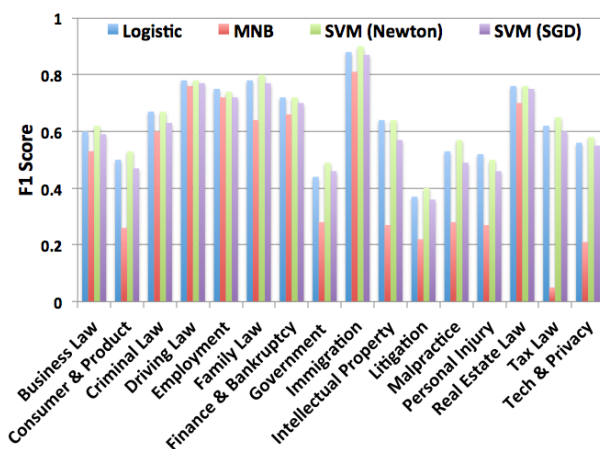


Figure 6: F1 Scores for the individual topic areas.

scores using the 1-layer neural network as well.

Our neural network implementation was initially extremely slow when we used the 35K input units from the training data word corpus. This prompted us to test out alternative input vector forms where we filtered and only kept the top K TF-IDF weighted features. Thus, for computational purposes, a lot of useful information was thrown out, which contributed to the lower test accuracy of 43%.

We then implemented a method that incorporates word2vec learn word representations as an alternative method to reduce the dimensionality of the input feature space while maintaining information about the data. We were pleased with the improvement to 68% test accuracy, and we plan on tweaking the model more in an attempt to further improve accuracy.

## 6.2 Sources of Error

### 6.2.1 Sentences Related to Multiple Practice Areas

Some legal questions inherently belong to multiple practice areas. For example, many questions might have relevance to both the "Consumer & Product" and the "Tech & Privacy" topic areas. Although each of the legal questions in our dataset only has one topic area label, the dataset inevitably contains legal questions that could fall under multiple categories. Our multiclass classifiers will have more trouble classifying these questions, which may explain the lower accuracy on topics like "Litigation."

### 6.2.2 Class Imbalance

Some of our topic area classes have fewer training samples than others. For example, "Intellectual Property" questions only have 3.5K questions compared with the average of 10.5K questions in other categories. To help remedy the unbalanced class problem, we can try certain techniques such as sub-sampling and up-sampling.

### 6.2.3 Sentence Imperfections

A unique aspect of our data is that the questions are unedited from users on the internet. Thus, there are issues regarding spelling, grammar, and homonyms of words being used which lead to different meanings. We can try using publicly available spelling correction APIs to find a correctly spelled word with the closest edit distance.

## 7 Future

There are various approaches that we plan to take to improve our current results. We will look to parse the input data in new ways and create a larger set of features upon which we can build a model. Additionally, we will consider methods of feature selection to reduce the feature space and look at alternative statistical models and methods which may be more conducive to the classification task at hand.

### 7.1 New Features

We plan on modifying our feature set to (1) include the location tags associated with each question and (2) give more weight to words in sentences depending on the sentiment of a sentence, and (3) using the Part of Speech tag for each word in the sentence. We will test out various new features that are more tailored to the task of classifying legal questions.

#### 7.1.1 Location Tags

For each question, we have information regarding the location in which the question was asked. This feature may prove to have discriminative power if, for example, more "Driving Law" questions are asked in California versus other states.

#### 7.1.2 Weighting

It may be useful to have features that give more weight to words in sentences ending with a question mark punctuation. A legal question in our dataset may contain multiple sentences. For example a single legal question may have 1-2 sentences of background before a sentence that asks the legal question itself. Features based on the legal question itself may be more indicative of a legal topic than the features based on the sentences about the background of the user's situation.

#### 7.1.3 Sentiment based features

We will also try to implement features that capture the sentiment of the legal question. For instance, we would apply sentiment analysis to assess the polarity of a given legal question, whether negative, neutral, or positive. For example, we would intuitively expect legal questions in "Personal Injury Law" to be more negative, as the user asking the question was likely physically injured. However, we may expect questions under "Business Law" to be more neutral or positive, such as

a question of "How do I incorporate my new business?"

#### 7.1.4 POS Tags

We plan to implement semantic features such as Part-of-Speech (POS) Tags. In POS tagging, we will use the definition of the words within the context of the specific legal question in order to annotate each word with its part of speech, such as labeling the word as a verb or a noun. This added information will help to differentiate between the same words that are used with different meanings.

### 7.2 Feature Selection

We plan on applying various feature selection methods [5] such as: (1) Chi-squared; (2) information gain; (3) removing frequent or rare words; (4) clustering related word features. These methods will allow us to prune the noisy features and only incorporate features that are most relevant to the classification task at hand.

## 8 Conclusion

We evaluated the ability of various different models to classify user-asked legal questions into the appropriate legal areas. As expected, a linear SVM model performed the best with a 69.1% test accuracy in multiclass classification. We plan to continue improving our current results using the new methods and metrics that we have mentioned.

## References

- [1] R. Nallapati, C.D. Manning. "Legal Docket Classification: Where Machine Learning Stumbles." EMNLP. (2008)
- [2] C.D. Manning, P. Raghavan and H. Schuetze. "Introduction to Information Retrieval," Cambridge University Press, (2008): pp. 234-265.
- [3] R.E. Fan, et al. "LIBLINEAR: A library for large linear classification," The Journal of Machine Learning Research 9, (2008): 1871-1874.
- [4] R.D. Goyal. "Knowledge Based Neural Network for text Classification," IEEE International Conference on Granular Computing. (2007): 542.
- [5] T. Joachims, "Text categorization with support vector machines: learning with many relevant features," In Proceedings of ECML, 10th European Conference on Machine Learning, 1998.
- [6] G. Forman. "An Extensive Empirical Study of Feature Selection Metrics for Text Classification." JMLR, 3:12891305, 2003