Enterprise Level Document Categorization

Abhinav Venkateswar Venkataraman
abhinav@soe.ucsc.edu

June 4, 2015

Abstract

The main objective of this project is to categorize enterprise level document into predetermined set of categories like legal, financial, revenue, sales etc. This problem is treated as a partial multi labeling problem for each client using the product.

A general corpus was built by scrapping documents pertaining to each category through web. A binary classifier model is built for every client and for each category in order to yield high precision results. These classifiers are built on Support Vector Machines(SVM) whose features are tf-idf scores of unigrams. They are regarded as the base model. The training was performed off line while the prediction was done on line. In addition to that, these models were modified by weighing the sample documents uploaded by the clients for whom the model was being built.

This report shows that weighing the sample documents provided by each client actually improves the performance of the model in specific to a particular client in relative to the base model.

1 Introduction

Cirrosecure, Inc is a SaaS based cloud monitoring product whose basic objective is to protect your enterprise cloud and identify the exposures associated with them. For eg, pay scale of different employees are written in a document which is created by HR department. If this document is shared to at least one of their other employees then that would affect the morale of the employees, where as this document needs be shared to other HR personnels and co-founders for some official purposes. Cirrosecure, Inc identifies

such unethical shares and lets us know about them. It has a scoring mechanism which determines how bad the exposure is and also provides details like whom its been exposed to, what information the document has been exposed(for eg, credit card information, ssn number etc...)

In addition to these features, document categorization is an important feature which would let people using this product know what kind of document has been exposed. Initially this was accomplished by having a bag of words model where if a document has words present in the bag of words and if the word count reaches a particular threshold then it categorized to a particular category. This could be easily improved by adding intelligence to the system which brings to the document categorization problem using machine learning techniques.

2 Project Outline

2.1 Building the Corpus

Initially common categories of enterprise documents were studied based on the popularity in enterprise level. After a lot of analysis they were tagged as Lawsuits, Revenue, Budget, Sales, Patent, Investors and Personnel. It was made sure that each category had well defined scope which would let us collect different kinds of documents in the same category. This would make sure we are as versatile as possible within one category. For example, category Personnel consist of payroll, offer letter, NDA and FLMA documents. It is to be noted that there can be overlap between the categories like every document which is a budget can also be categorized as revenue. Once the scope of each category was well defined, we start gathering data. We build the corpus for every category by scrapping data through web. We tend to be as versatile as possible in covering different kinds of documents and also collect in an equal fashion so that of them are weighed equally.

2.2 Preprocessing

Having built the corpus we now need to do preprocessing on the data which deals with maintaining the same encoding through out the corpus, dealing with white spaces, special symbols. All characters except for letters are removed. After the first step of data cleaning, we then remove stop words and

all words that occur only once in the entire corpus. This is done because they are totally insignificant and they contribute nothing to the score as a feature to the category.

2.3 Data Analysis

n-gram count of text was taken and it was seen that the data becomes really sparse when it comes $n > 1$ i.e the majority count for other grams is 1 and hence we consider only unigrams for this project.

After pruning out some of the unigrams tf-idf measure for each unigram was calculated over the corpus. tf-idf is a well defined and standard measure that tells how important the word is for a document. Once tf-idf is calculated for every document, term cell in a document term matrix an estimate for sparsity of the matrix was done and it showed that the matrix was really sparse.

In order to reduce the sparseness of the data, we performed latent semantic indexing(LSI) of unigrams which is equivalent of doing a single valued decomposition (SVD) on any such measure.

2.4 Learning and Validation

After having representing the data in document term matrix in a most optimal way(which is performing LSI over unigrams) the matrix was fed into support vector machine(SVM) with linear kernel in order learn the binary classifier which was later cross validated using 10-fold cross validation technique.

3 Experimental Setup

Each category has a positive and negative corpus. Negative corpus were predominantly random English documents with few docs that belong to complimentary category so that they are not mislabeled. For eg, a patent document should never be marked as budget. So the negative corpus of patent category would include budget category documents as negative cor-

pus too.

Every positive corpus has approximately 150 documents and 300 documents in negative corpus. Initially 10% from both positive and negative corpus are taken out and treated test data set. We train with the remaining 80% data and test on the withheld set. In order to modifying the model, the sample files are added to the positive corpus in a way that they add more weight in the positive corpus. In this project, they are replicated so that the skewness of the model is actually projected as a proper way.