CENG 499 – Introduction to Machine Learning

Project Report

June 12, 2017

Team Members

1. Ata Duru, e1942002@ceng.metu.edu.tr
2. Enver Evci, e1942085@ceng.metu.edu.tr
3. Onat Büyükakkuş, e2035772@ceng.metu.edu.tr
4. Onur Adıgüzel, e1941665@ceng.metu.edu.tr

Problem Definition

In parallel with the growth of Internet, Every day, news websites produce an astonishing number of articles about an astonishing number of topics.(92000 articles, 2 million blog posts) Classification of online news, has often been done manually This requires too much time and human effort.

Data Set Description

We found the data in database of BBC. Since we decided to categorize the news in 5 different titles, we took the 150 data for each category. We used 100 of them for training and 50 of them for testing.
The training and the testing data can be found in Data directory in the zip file.

The data was collected between 2004-2005 from BBC news website by BBC.

Since we use Naive Bayes Classifier, there is no field. Algorithm looks every word one by one and calculates a probability.

We have 5 class labels. Each of them refers one category. These labels are business, entertainment, politics, sport and tech. Each of them labels the news about that category.
Business is about personal finance, company, financial and economic news, plus analysis of global markets.
Entertainment is about celebrities, celebrity gossip, society, fashion news, movies and TV series.
Politics is about opinion and analysis of global politics. More specifically, it is about political parties, political campaigns, world and international politics.
Sport is about football, golf, rugby, cricket, tennis, F1, boxing, plus the latest sports news, transfers & scores.
Tech is about latest technology news and highlights covering mobile phones, computer hardware, software and IT jobs.

Project Work

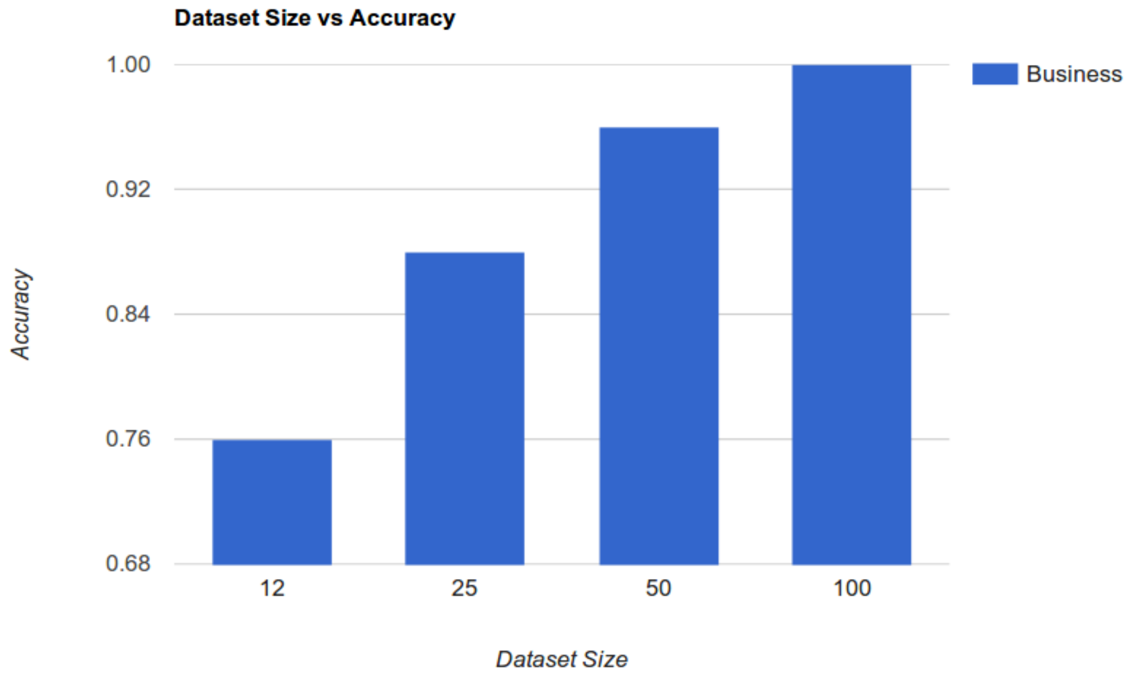
We used TextBlob[1] library for text classification. TextBlob is a Python (2 and 3) library for processing textual data. It provides a simple API for diving into common natural language processing (NLP) tasks such as part-of-speech tagging, noun phrase extraction, sentiment analysis, classification, translation, and more.

We used its classification feature for our project. For classification, library uses Naive Bayes Classifier. The Bayesian Classification represents a supervised learning method as well as a statistical method for classification. Assumes an underlying probabilistic model and it allows us to capture uncertainty about the model in a principled way by determining probabilities of the outcomes. It can solve diagnostic and predictive problems.

As we mentioned above, we have many data to train and to test. We categorized every training data under directories named its label. While training, trainer.py is used. In the implementation of it, each directory is traversed one by one. After getting the data, we crop the stopwords by using the data in the following link which is on the cow course page:
http://xpo6.com/download-stop-word-list/.
After cropping, we remove all non-ascii characters and punctuation marks because each of them seems a word after cropping and this reduces the accuracy. After all these processes, we train the final data and put the trained model to a pickle file in order not to train the data every time because it requires some amount of time.
In testing part, main.py is used. When the application is run, a gui is opened. There are 2 option in the application. In the first one, some test data can be given and a news can be given as a text in the other option. Path of the pickle file is given from gui for both option because it stores the model. For the first option, path of the test data must be given. The path must contain category directories and each category must contain its data. It reads the data and creates tuples for each data with its category. We put the tuples to a list and give the list to accuracy function of library. Application checks each data and calculates the accuracy of the test. For the other option, a random news can be given as an input to the area as text and the application gives the correct category of the news.
Complete implementation can be found in Source directory of project.

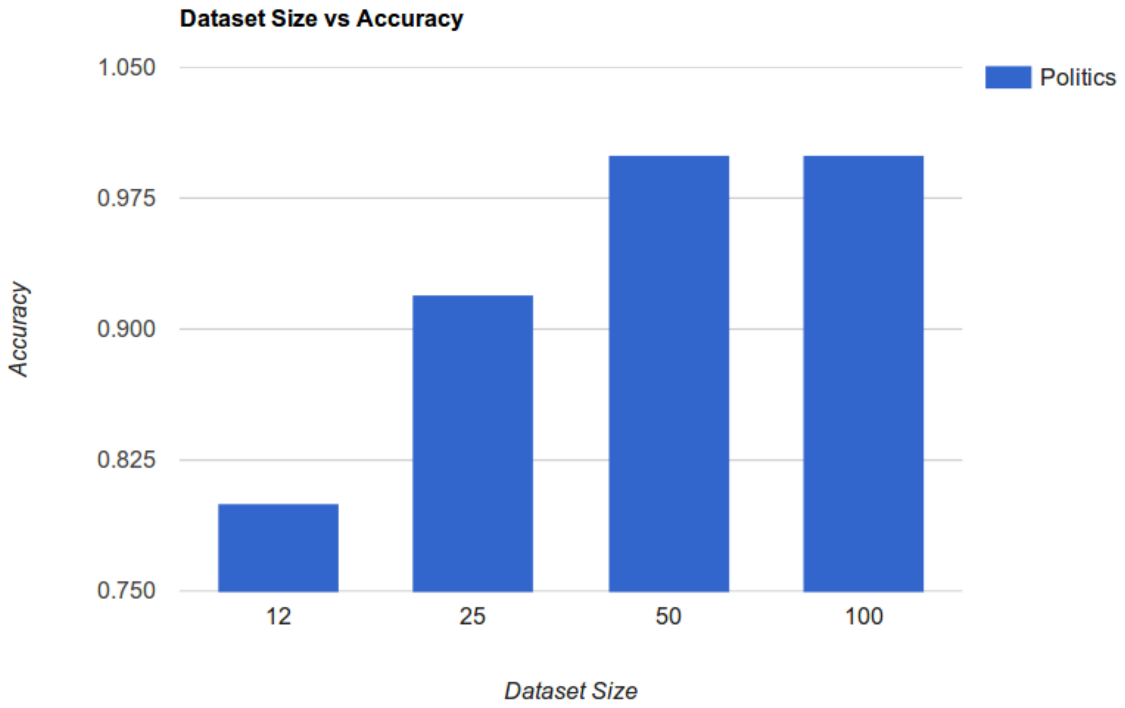
We expect high accuracy for our implementation because we used Naive Bayes Classification. However, sometimes accuracy may decrease because some news in entertainment could be so generic and application may classify it as business or some sport news such as transfer news may be classified as business. Despite all of these, we expect a accuracy higher than 85%.

Results



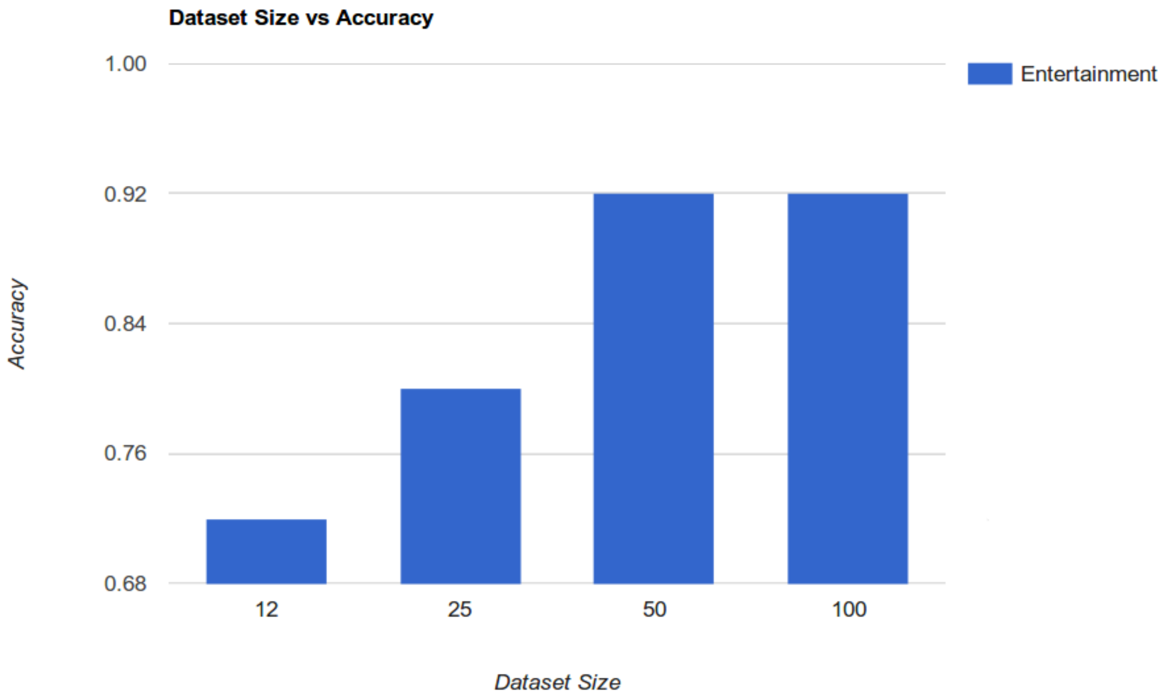
Graph 1 - Result of Business News

We tried

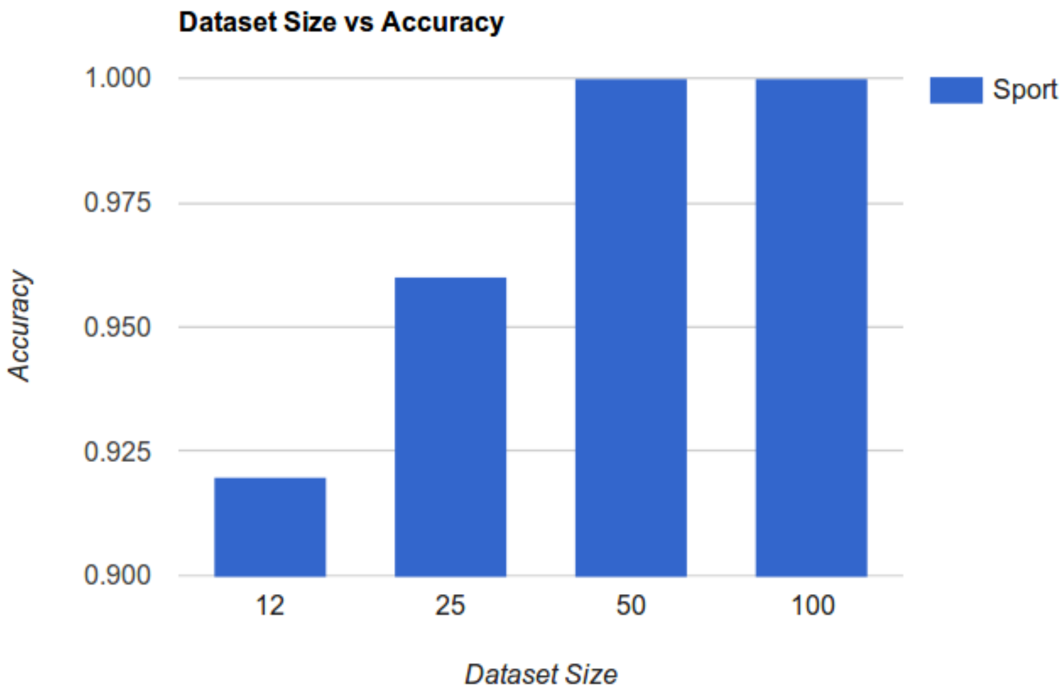


Grapc 2 - Result of Politics News

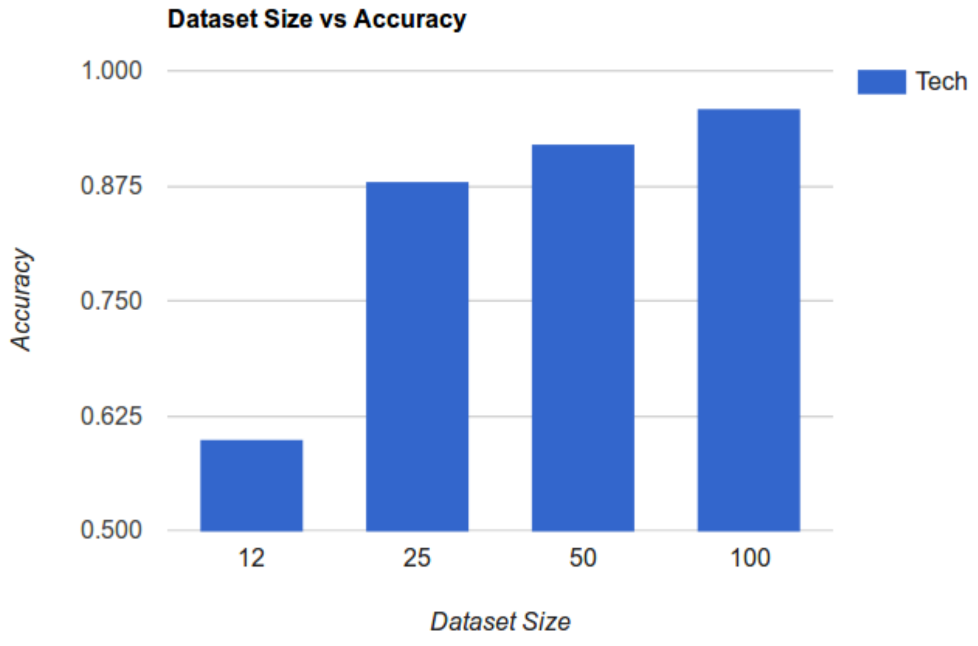
different size of training dataset with different size of test dataset. We trained every category with 12, 25, 50 and 100 data. Then, we tested them some dataset whose size was half of the train dataset. The results can be seen below.



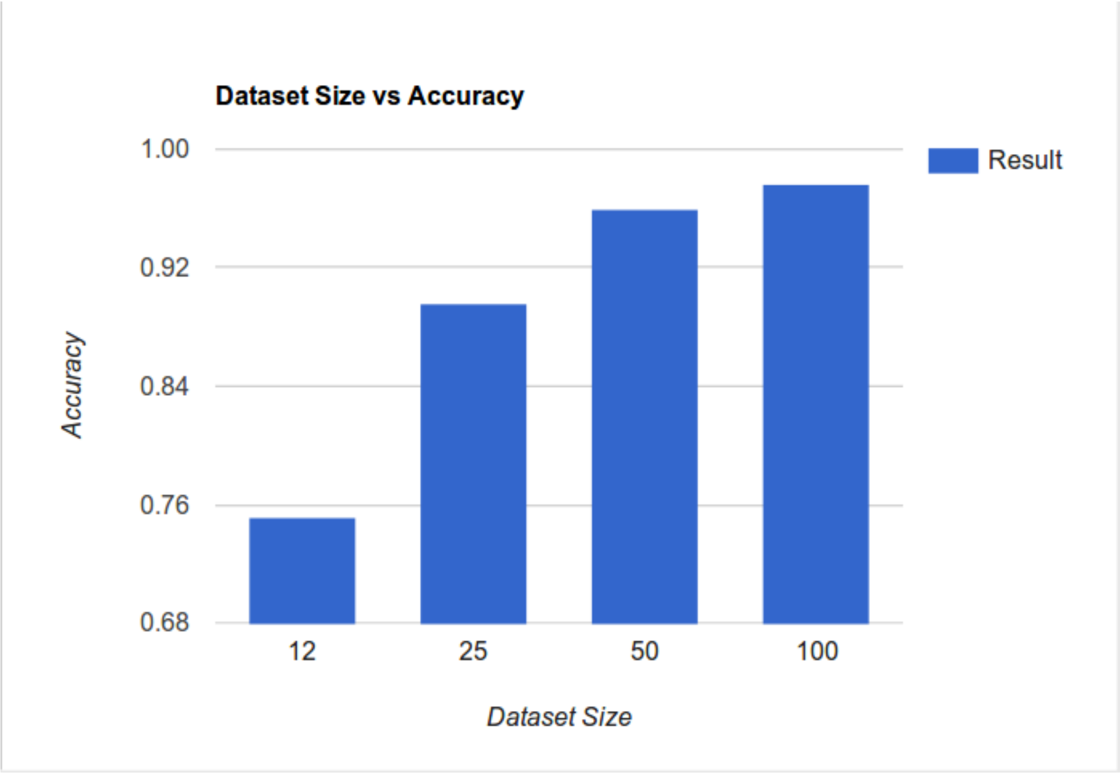
Graph 3 - Result of Entertainment News



Graph 4 - Result of Sport News



Graph 5 - Result of Tech News



Conclusion

Being an eager learner, naive Bayes classifiers are known to be relatively fast in classifying new instances. Eager learners are learning algorithms that learn a model from a training dataset as soon as the data becomes available. Once the model is learned, the training data does not have to be re-evaluated in order to make a new prediction. In case of eager learners, the computationally most expensive step is the model building step whereas the classification of new instances is relatively fast. In our project, creating and loading the model (pkl file) takes some time but after that, classification is quite fast. Also, as it can be seen in the graphs above, we tested our application with datasets with different sizes. As the size of dataset grows, so does the accuracy.

References

[1] - TextBlob - http://textblob.readthedocs.io/en/dev/_modules/textblob/classifiers.html

Graph 6 - Overall Results of All Types of News