Topic Classification of Dialogues from National Public Radio Excerpts

Daniel Galeano Timardeep Kaur Arneja Sandra Maria Nawar

260623579 260617404 266065071

McGill University McGill University Concordia University

Montreal, Quebec, Canada Montreal, Quebec, Canada Montreal, Quebec, Canada

daniel.galeanogranados@mail.mcgill.ca [timardeep.arneja@mail.mcgill.ca](mailto:timardeep.arneja@mail.mcgill.ca) sandra.nawar@mail.mcgill.ca

***Abstract-*-We are living in the world where we are generating lots of volumes of data on daily basis. How cumbersome it will be to manage the BIG DATA if we have to classify each and individual document manually and put it into its repository. In such scenarios, Automatic Text Classification comes as a handy tool to minimize human effort and provide an efficient way of classifying and managing the documents that are scattered and whose numbers is continuously increasing. Text Categorization, from decades, has drawn the attention of researchers and is one of the well-studied problems in the field of machine learning. In this work we aim to classify the excerpts of conversations transcribed from interviews on National Public Radio into one of the categories ‘author’, ’music’, ’movies’ and ‘interviews’ .We will compare the performance of Simple Naïve Bayes, k-nearest neighbors algorithm, and other advanced algorithms such as Support Vector Machines using various kernels on the test set provided. Here we will provide some information as to which algorithm performed best**

1. INTRODUCTION

Text Classification is a task of assigning categories to different texts and this classification can provide conceptual view of document collection and has important applications in real world [3]. With the advent of Big Data and volumes of data that is generated these days, it has become practically impossible to go over all the data and then provide labels for them. So, instead of classifying labels for all the texts manually, Statistical Text categorization uses machine learning methods to learn the classification rules based on human labelling of Training Dataset. In this study, we aim to classify the excerpts of conversations from National Public Radio into one of the categories ‘author’, ‘music’, ’movies’ and ‘interviews’. Our Training Dataset contains a lot of text, but all the text is not useful and does not convey to us any meaningful or intuitive information as to which category particular text from test dataset belongs to. So, we are implementing bag-of-words method to extract useful words from our training set. The success of any Text Classification Algorithm to a large extent depends on the feature selection. In this work, we are using Mutual Information (MI selector). After successfully extracting the features from training set, we are using the training data to train the Classifiers such as Naïve Bayes, Support Vector Machines and k-nearest neighbors. We are also considering certain heuristics which are specific to the dataset while selecting the features with the sole aim to make classifier perform better. Finally, we will present our results and discussion on each of the classifier implemented in this paper.

1. PRE-PROCESSING THE DATA
2. FEATURE SELECTION

­­­­A major challenge in text classification consists on the selection of features due to the rich nature of natural languages. The English language for example contains more than a million words; which represents a very large feature space, and which results computationally infeasible.

*Mutual Information*

In this project we selected a feature selection method based on our need to reduce computational cost, and the specific intention to classify text. We implemented the Mutual Information (MI) feature selector, which measures how much information the presence/absence of a word contributes to making the correct classification decision of a conversation.

The concept of MI is defined in information theory for two discrete random variables as follows:

In the case of text classification, one of the random variables indicates if a document contains a given word, while the other variable indicates if the conversation belongs to a given topic.

Chapter 13 in [4] does a good job at describing the algorithm to implement mutual information in order to extract features in a text classification problem. In this chapter the equation to calculate MI is represented in terms of counters for implementation purposes.

Where for example represents the number of conversations where word X is absent, but label with class Y. Please refer to [4] for more in detail description of this equation.

Since MI indicates how much information a word contains about a given class, we selected the words with the highest MI values for each class to build the features space of our classifier.

The following table shows the top 5 words, with descending priority, from each of the classifiers when using the MI algorithm to prioritize features, which are actually making sense for each of these classes.

|  |  |
| --- | --- |
| **Topic** | **Top Five Features** |
| *Author* | book, write, read, author, story |
| *Movies* | film, movie, scene, actor, director |
| *Music* | song, album, band, record, play |
| *Interview* | president, time, say, government, look |

1. ALGORITHMIC IMPLEMENTATION
2. NAÏVE BAYES CLASSIFIER:
3. METHODOLOGY
4. RESULTS AND DISCUSSIONS

1. K-NEAREST NEIGHBORS:
2. METHODOLOGY
3. RESULTS AND DISCUSSIONS
4. SUPPORT VECTOR MACHINES (SVM)

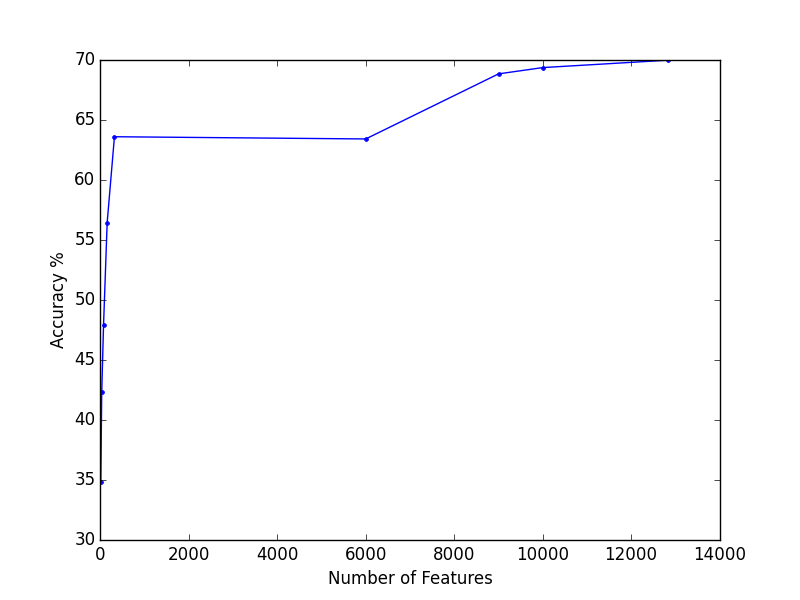
A support vector machine constructs a hyper-plane or set of hyper-planes in a high or infinite dimensional space, which can be used for classification. Separation is achieved by the hyper-plane that has the largest distance to the nearest training data points of any class, the margin, since in general the larger the margin the lower the generalization error of the classifier. This indicates that classes have been well separated. When fitting vectors for multiple classes the training vectors are implicitly mapped into a higher dimensional space by the function.

I. METHODS

Support Vector Machines has been implemented with different kernels (Linear, Radial basis (Gaussian), 3rd degree Polynomial). It is recommended to use linear kernels for text categorization, as most of text classification problems are linearly separable. We will show the results of different kernels and compare it to answer the question if it is worth it to fit more complex kernels. Problem with SVM in python [using scikit-learn runs endlessly and never completes execution](http://datascience.stackexchange.com/questions/989/svm-using-scikit-learn-runs-endlessly-and-never-completes-execution). So we only run it for a small number of features and training set.

Another method that has executes faster is SVM with gradient descent optimization.

RESULTS AND DISCUSSION



1. DISCUSSION
2. FUTURE WORK

ACKNOWLEDGMENT

REFERENCES

[1] Vapnik, Vladimir. "The Support Vector Method of Function Estimation." Nonlinear Modeling. N.p.: Springer US, 1998. 55-85.

[2] Joachims, Thorsten. "Text Categorization with Support Vector Machines: Learning with Many Relevant Features." (n.d.): n. pag. Web.

[3] <http://www.scholarpedia.org/article/Text_categorization>

[4] [Christopher D. Manning](http://nlp.stanford.edu/~manning/), [Prabhakar Raghavan](http://theory.stanford.edu/~pragh/) and [Hinrich Schütze](http://www.cis.uni-muenchen.de/personen/professoren/schuetze/), Introduction to Information Retrieval, Cambridge University Press. 2008.