A Framework for Text Analytics using the Bag of Words (BoW) Model for Prediction

Deepu S[1], Pethuru Raj[2] and S.Rajaraajeswari[3]
[1]2nd Year MCA, Raja Rajeswari College of Engineering, Bangalore-560074, India
[2]Infrastructure Architect, IBM Global Cloud Center of Excellence, IBM India, Bangalore-560045, India
[3]Department of MCA, Raja Rajeswari College of Engineering, Bangalore-560074, India

ABSTRACT- With the steady accumulation of unstructured data, the domain of natural language processing (NLP) is gaining widespread attraction amongst researchers and practitioners in order to quickly and easily extracts prediction-like insights in a simplified and streamlined fashion. The subject of text mining and analytics is going through a variety of delectable advancements. There are a number of articulations on the subjects of NLP and machine learning (ML). Very recently, the model of the bag of words has become so popular in order to produce accurate predictions out of unstructured text data. In this paper, we have explained an easy-to-use framework for accelerated usage of the BoW model towards pioneering text mining and processing. We have demonstrated a simple example by leveraging this framework in order to showcase the utility of this generic framework that can be easily replicated across in manyother associated scenarios.

Index Terms—Bag of Words, Machine Learning; Natural Language Processing, Text Mining, Predictive Analytics, Bernoulli Document Model.

I. INTRODUCTION

Predictive analytics has become the key subject of study and research these days [1]. As data become big data due to the explosion of unstructured text documents, the researchers across the globe are focusing on unearthing and experimenting a variety of powerful algorithms and approaches for speeding up the process of processing online as well as off-line text files with the aim of producing something useful to act upon with all the clarity and confidence. There are algorithms in plenty for doing various analytical activities in order to extract actionable insights in time out of big, fast, streaming and IoT. Machine learning [2] is emerging as one of the hot topics in the industry as well as in academic institutions to smoothen the analytical tasks.

With the steady growth of text content, the paradigm of natural language processing (NLP) [3] is fast-evolving these days. Inspired minds and luminaries are striving hard and stretching further to bring forth a bevy of workable methods and mechanisms that in turn facilitate pragmatic knowledge and predictions. We have found through our extensive experiences that the emerging BoW model seems to be a better fit for predicting something by processing text content. This paper is for explaining the nitty-gritty of the proposed framework for expediting the prediction from large-scale unstructured text data.

II. EXPLAINING THE BAGOF WORDS PREDICTION MODEL

The Bag of Words (BoW) model learns a vocabulary from all of the documents, and then models each document by counting the number of times each word appears. The BoW model is a simplifying representation used in natural language processing and information retrieval (IR). In this model, a text (such as a sentence or a document) is represented as the bag (multiset) of its words, disregarding grammar and even word order but keeping multiplicity. Recently, there are different domains showing a lot of interesting in leveraging this BoW technique towards efficient text analytics. The BoW model is commonly used in methods of document classification, where the (frequency of) occurrence of each word is used as a feature for training a classifier [4, 5, 6].

For example, consider the following two sentences:

Sentence 1: "The cat sat on the hat"
Sentence 2: "The dog ate the cat and the hat"

From these two sentences, our vocabulary is as follows:

{the, cat, sat, on, hat, dog, ate, and }

To get our bags of words, we count the number of times each word occurs in each sentence. In Sentence 1, "the" appears twice, and "cat,‖ "sat", "on", and "hat" each appears once, so the feature vector for Sentence 1 is:

{the, cat, sat, on, hat, dog, ate, and }

Sentence 1: {2, 1, 1, 1, 1, 0, 0, 0}

Similarly, the features for Sentence 2 are: {3, 1, 0, 0, 1, 1, 1, 1}

For example, as per the Wikipedia, the BoW model has also been used for computer vision. In computer vision, the BoW model can be applied to image classification, by treating image features as words. In document classification, a bag of words is a sparse vector of occurrence counts of words; that is, a sparse histogram over the vocabulary. In computer vision, a bag of visual words is a vector of occurrence counts of a vocabulary of local image features.

III. THE BOW BASED ANALYTICS METHODOLOGY

Text classification is the task of classifying documents by their content: that is, by the words of which they are comprised. Perhaps the best-known current text classification problem is email spam filtering: classifying email messages into spam and non-spam. We have arrived at and articulated an optimal methodology for executing the BoW model on text data sets based on our wide experiences gained out of various works that intrinsically leveraged the distinct power of BoW technique. The steps are enumerated below.

1. **Reading the Unstructured Text Data –** The unstructured text data that needs to be subjected to a variety of investigations. So the text files need to be download and ingested. For example, there are tweets, Facebook comments, blogs, the addresses of celebrities in various fields, the various sentiments expressed by users, etc. We have used R Studio to read the file to be examined.
2. **Data Preprocessing** – This is a phase wherein all kinds of noises of text data need to be filtered out. If there is any kind of markers and extra white spaces, they need to be identified and eliminated before jumping into the processing and analytics phase. If there is any HTML tags or punctuation marks, numbers, etc. in the text document, they need to be meticulously identified and erased in order to arrive at highly organized and optimized document. There are other cleaning operations to be accomplished through a number of reviews and refinements. There are already developed libraries, packages, and classes in order to remove all kinds of
3. **Knowledge Discovery** – As articulated in the beginning, the Bag of Words model learns a vocabulary from all of the documents and then models each document by counting the number of times each word appears. We have explained this phase in detail through a practical example below in order to enhance the readability and understandability of the BoW technique in extracting predictions out of massive scales of text data. We have chosen to use the Bernoulli Document model [7], which is detailed below.
4. **Knowledge Dissemination** – There are visualization platforms, dashboards, report-generation tools, etc. for displaying and demonstrating the extracted in a user-preferred manner. There are automated systems to showcase the knowledge in a 360-degree view.

This is a very generic framework and there are multiple technologies, tools, and techniques to be leveraged. Advanced algorithms can be easily sneaked in to ensure high performance and other non-functional requirements.

IV. ABOUT THE BERNOULLI DOCUMENT MODEL

Text classifiers typically don't use any kind of deep representation about the language. A document crafted in the language is represented as a bag of words. A bag is like a set that allows repeating elements. This is an extremely simple representation. That is, it only knows which words are included in the document and how many times each word occurs. This does not take the word order into account. There are two probabilistic models of documents (multinomial model and multivariate Bernoulli model). Both represent documents as a bag of words using the Naive Bayes (NB) assumption. Both models represent documents using feature vectors whose components correspond to word types. If we have a vocabulary V, containing |V| word types, then the feature vector dimension is d=|V|.

A. Bernoulli Document Model

A document is represented by a feature vector with binary elements taking value 1 if the corresponding word is present in the document and 0 if the word is not present. Bernoulli model uses binary occurrence information, ignoring the number of occurrences, whereas the multinomial model keeps track of multiple occurrences.

B. A Sample Application Development Steps

1) Installing R and RStudio

RStudio is an integrated development environment (IDE) for R. We have used the free version. Like R, RStudio installs painlessly and also detects your R installation. Text mining and certain plotting packages are not installed by default so one has to install them manually. The relevant packages are:

1. **tm** – the text mining package.
2. **SnowballC** – required for stemming
3. **ggplot2** – plotting capabilities
4. **wordcloud** – which is self-explanatory

The simplest way to install packages is to use RStudio's built-in capabilities (go to *Tools* > *Install Packages* in the menu).

2) Dataset Ingestion

For our experimentation, we have taken a data set from this link [9]
https://archive.ics.uci.edu/ml/datasets/Bag+of+Words
Download and ingest the dataset into R Studio. There is a provision for file ingestion.

3) Cleansing for Test Processing, Mining and Analysis

The text file has to go through a series of corrections such as the removal of numbers, capitalization, common words, punctuation and other noises as the first and foremost step for subjecting text documents to a variety of investigations.

4) Leveraging Bernoulli Document Model

By using the text mining software, which has intrinsically implemented the Bernoulli document model, it is possible to bring forth word matrix.

5) Plotting the Obtained

There are additional software modules such as ggplot2 and wordcloud to create effective visualizations.

V. THE PICTORIAL REPRESENTATION OF METHODOLOGY

The prominent steps are being illustrated through the following diagram.

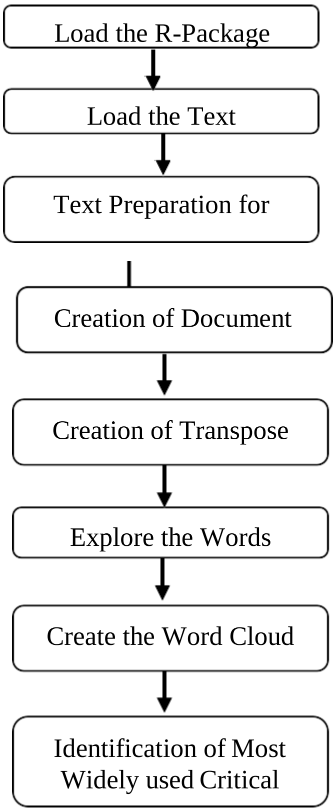


Figure 1 – The Process Flow Chart for Text Mining using –R‖

VI. THE SAMPLE OUTPUT

With the use of wordcloud solution, it is possible to craft different outputs as evidently indicated below.

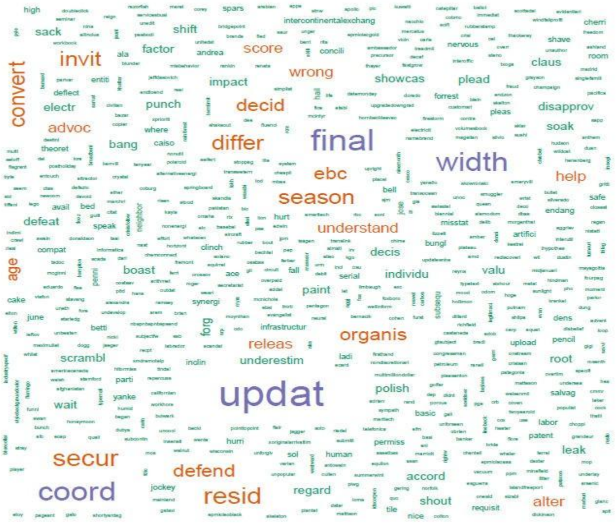


Figure 2 – The Word cloud Output

The ggplot2 is capable of producing the following diagram indicating the words that appear more than 5 times

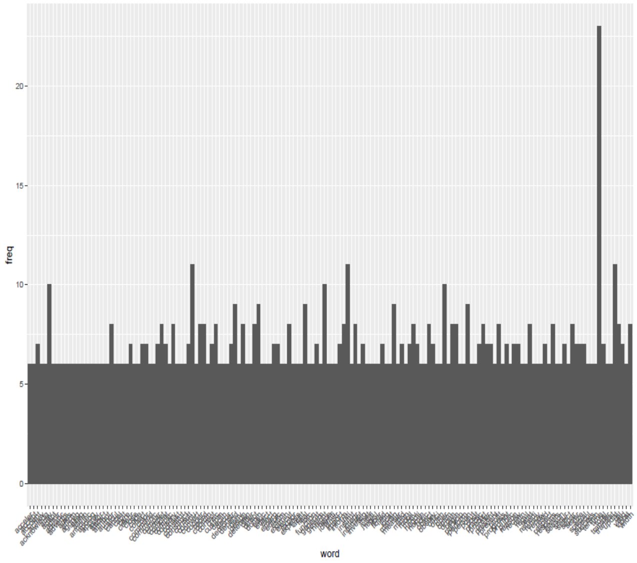
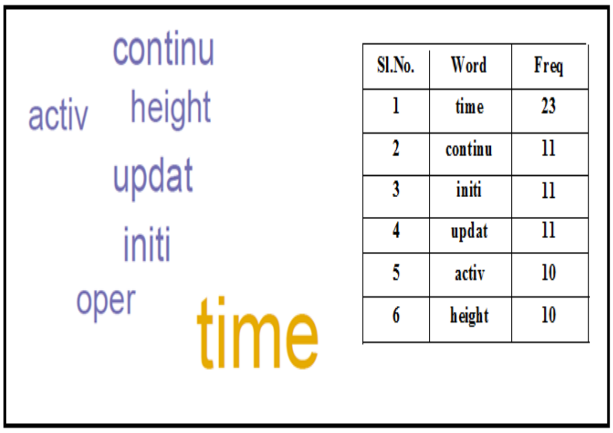


Figure 3 - Words Appeared More Than Five Times

The next output indicates the number of words that appear more than 10 or more times.


continu
activ
height
updat
initi
oper
time

Sl.No.	Word	Freq
1	time	23
2	continu	11
3	initi	11
4	updat	11
5	activ	10
6	height	10

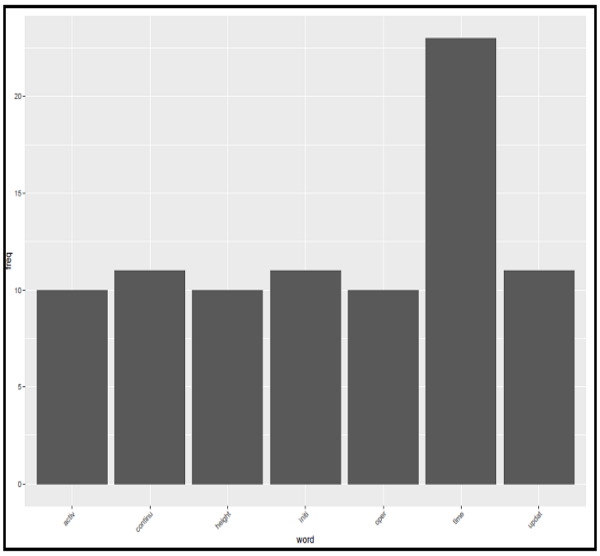


Figure 4 – The Words Appeared More Than Ten Times

VII. CONCLUSION

We all know that data heaps are blessed with a variety of pragmatic knowledge. However, there are two major barriers. There is a rapid growth of unstructured text files originating from different sources and social sites and the amount of data getting generated, captured, and subjected to purpose-specific processing, mining and analytics are growing exponentially. In order to extract any hidden patterns among text files, there comes a number of ground-breaking algorithms, game-changing software solutions, integrated platforms, NoSQL databases, and cloud infrastructures. In this paper, we have brought in an enabling framework to make sense out of word documents quickly and easily. We have also demonstrated the efficiency and effectiveness of our framework through a sample implementation. There are a few BoW models and we plan to do a comparative study of them before incorporating the best-in-class BoW model in order to strengthen our framework.

REFERENCES

[1] Frank Buytendijk and Lucie Trepanier, (2010), Predictive Analytics: Bringing the Tools to the Data, Oracle Whitepaper

[2] Pedro Domingos, A Few Useful Things to Know about Machine Learning, *https://homes.cs.washington.edu/~pedrod/papers/cacm12.pdf*

[3] Ronan Collobert, JasonWeston, L´eon Bottou, Michael Karlen, Koray Kavukcuoglu and Pavel Kuksa, (2011), Natural Language Processing (Almost) from Scratch, Journal of Machine Learning Research, 12, 2493-2537

[4] Dani Yogatama and Noah A. Smith, (2014), Making the Most of Bag of Words: Sentence Regularization with Alternating Direction Method of Multipliers, Proceedings of the 31st International Conference on Machine Learning, Beijing, China, JMLR: W&CP, volume 32

[5] Cordelia Schmid, Bag-of-features for category classification, *www.cs.umd.edu/~djacobs/CMSC426/BagofWords.pdf*

[6] Jialu Liu, Image Retrieval based on Bag-of-Words model, *jialu.cs.illinois.edu/technical_notes/CBIR_BoW.pdf*

[7] Hiroshi Shimodaira, (2015), Text Classification using Naive Bayes, Learning and Data Note 7, Informatics 2B

[8] https://cran.r-project.org/mirrors.html

[9] https://archive.ics.uci.edu/ml/datasets/Bag+of+Words