IJCSE International Journal of Computer Sciences and Engineering Open Access
Survey Paper Vol.-7, Issue-4, April 2019 E-ISSN: 2347-2693

Survey of Feature Selection and Text Classification Methods for Genetic Mutation Classification

Varun Saproo[1*], Rujuta Upadhyay[2], Manisha Valera[3]

[1,2,3] Department of Computer Engineering, Indus University, Ahmedabad, India

Corresponding Author:saproo@outlook.in, Tel.: +91-8141253153

DOI: https://doi.org/10.26438/ijcse/v7i4.933937 | Available online at: www.ijcseonline.org

Accepted: 20/Apr/2019, Published: 30/Apr/2019

Abstract— Genetic testing and precision medicine have changed how a disease like cancer is treated. It's a very time-consuming task where a clinical pathologist has to manually review and classify every single genetic mutation based on evidence from text-based clinical literature takes up a considerable amount of human efforts and time. In this paper, we survey different machine learning models with an intent to automate the mutation classification. Additionally, to speed up the learning process while maintaining accuracy, Jeffreys-Multi-Hypothesis (JMH) divergence method is used to select words with large discriminative capacity for classification of text. Text Encoding Schemes like BoW (Bag-of-Words), TF-IDF (Term Frequency-Inverse Document Frequency, and Graph-based TW-IDF (Term Weight - Inverse Document Frequency) is used to encode text to numerical form. Macro-based F1-score is used to score performance during feature selection and model evaluation. This paper surveys the specified methods based on comparisons and tries to conclude which turns out to be better.

Keywords— BoW, TF-IDF,TW-IDF, JMH Divergence, Precision, Recall, F1-Score

I. INTRODUCTION

The Sloan Cancer Centre in New York has made an expert annotated knowledge base available for public use, where world-class researchers have manually annotated several mutations. The challenge is to distinguish the data into nine different classes genetic mutations i.e.

1 - Likely Loss-of-function

2 - Likely Gain-of-function

3 - Neutral

4 - Loss-of-function

5 - Likely Neutral

6 - Inconclusive

7 - Gain-of-function

8 - Likely Switch-of-function

9 - Switch-of-function

thus aims to distinguish tumor causing mutations (drivers) from the neutral mutations (passengers) for a given gene-variation pair. A clinical pathologist has to manually review and classify every single genetic mutation based on evidence from text-based clinical literature which takes up a considerable amount of human efforts and can result in quite some error. Thus, automating this task would make the entire process efficient and convenient for the pathologist. This survey would involve use of different supervised machine learning algorithm to perform multi-class text classification. The dataset used is provided by Memorial Sloan Cancer Kettering Centre located in New York in the United States[1].As for the data set, there are two of them, training_variants and test_variants. In training_variants there are 3321 samples, and in test_variants there are 368 samples. The data set is imbalanced, where "Likely Gain-of-function" and "Gain-of-function" classes are a majority. While "Neutral" and "Likely Switch-of-Function" are a minority.

II. TEXT ENCODING SCHEMES

The dataset has a text corpus which is a collection of documents with their relevant class labels. To classify text, one has to convert text into numerical form. Following are the approaches -:

A. Bag-of-Words (BoW) - A document represented in the form of a vector of words; each document gathered from the dataset. Each component of the vector is weighted based on its frequency in the document. Bag-of-Words representation of documents and the corresponding scoring functions do not retain information about the ordering and position of the terms in the document. Bag-

of-Words is not perfect. Breaking down a sentence into single words can destroy the semantic meaning. [2].

B. Term Frequency-Inverse Document Frequency (TF-IDF) - It is a numerical statistic which reveals the importance of the word in a document. TF-IDF value of a word *t* in document *d* is proportional to the frequency of a term in a given document, and inversely proportional to the frequency of documents which contain the term *t.*

$$tf - idf(t,d) = tf(t,d) * idf(t,D)$$

$$idf(t) = \log\left(\frac{1+n}{1+df(t)}\right) + 1$$

C. Term Frequency-Inverse Document Frequency-Class Frequency (TF-IDF-CF)[3] – It is an addition to TF-IDF, which calculates the term frequency in documents within one class. It is calculated as,

$$a_{ij} = \log(tf_{ij} + 1) * \log\left(\frac{N + 1.0}{n_j}\right) * \frac{n_{cij}}{N_{ci}}$$

n_{cij} represents the frequency of documents where term *j* belongs to document *i* within same class *c*.

n_{ci} is the number of documents within same class *c*

D. Graph based Term Weight-Inverse Document Frequency(TW-IDF)[4] – This scheme uses the graph-based representation of documents and derives novel term weighting scheme by considering the local and global criterion.

Degree Centrality - It captures the local information of the node, that is how important the node is in its neighbours.

$$degree\_centrality(i) = \frac{N(i)}{|V| - 1}$$

Closeness Centrality - It is a global metric score which measures how close is one node to all other nodes in the graph.

$$closeness\_centrality(i) = \frac{|V|-1}{\sum_{j \in V} dist(i,j)}$$

The weight term is calculated based on the centrality score. TW-IDF of the document is specified as -:

$$tw - idf(t,d) = \frac{tw(t,d)}{1 - b + b * \frac{|d|}{avgdl}} * idf(t,D)$$

where *t* is the term, *d* is the current document, *D* is the collection of documents.

Before encoding the text to numerical form, cleaning the text is an important task. The cleaning process involves removing both punctuations and stop-words. Stop-words are language-specific functional words, that carry no information[10].

III. FEATURE SELECTION

In the previous section, we have used different encoding schemes to encode text into numerical form. All text classification problems treat terms as features. A large corpus is evident enough to guess how ample the feature space would be. A large corpus would use different terms in large number; thus ample is the feature space. But there would be terms which are irrelevant i.e. their absence in the feature space won't affect the classifier's performance by much.
Feature Selection is the process of selection features which largely contribute to classifier's performance. There are two types of feature selection methods, filter, and wrapper. The filter approach selects feature subsets based on the general characteristics of the data without involving the learning algorithms. These methods score features based on their importance. The wrapper approach greedily searches for better features with an evaluation criterion based on some learning algorithm[5]. Although the wrapper approach usually performs better than the filter approach, it has much more computational cost than the filter approach, which sometimes makes it impractical [5].

A. Document Frequency – It is the most straightforward measure of term importance which is the frequency of documents, containing the term *t* in corpus *D.* Term with less document frequency is ranked more.

B. Mutual Information[6] - Mutual Information is a statistical measure which measures the dependency between two random variables. In other words, it measures the reduction of uncertainty of a variable given the knowledge of another variable.

Given term *t* and category *c*, the *pointwise mutual information* criteria between *t* and *c* defined as-:

$$PI(t,c) = \log(\frac{p(t \wedge c)}{p(t) \times p(c)})$$

where $p(t \wedge c)$ represents the joint probability between *t and c*, $p(t)$ and $p(c)$ are marginal probabilities. These category-specific scores of a term are then combined to measure the goodness of the term at a global level. Typically it can be calculated in one of two ways-:

$$PI_{avg}(t) = \sum_{i=1}^{m} p(c_i) PI(t, c_i)$$
$$PI_{max}(t) = \max_{1 \le i \le m} PI(t, c_i)$$

If *t* and *c* are complementary distributions, the *PI(t, c)* will be much less than *p(t) p(c)* , forcing *PI(t, c) << 0.* That is, pointwise mutual information can also be negative.

According to Information Theory, Mutual Information is non-negative, therefore pointwise mutual information defined above is not the actual "mutual information" defined above[6].

Information theoretic Mutual Information between two discrete random variables X and Y, is defined as

$$(X;Y) = \sum_{x \in X} \sum_{y \in Y} p(x,y) \log_2 \left(\frac{p(x,y)}{p(x)p(y)}\right)$$

Let *T* and *C* be two random variables where *C* represents the complete set of categories and T = {*t*, t^C }.

$$I(t;C) = \sum_{c \in C} p(t,c) \log_2 \left(\frac{p(t,c)}{p(t)p(c)}\right) + \sum_{c \in C} p(t^c,c) \log_2 \left(\frac{p(t^c,c)}{p(t^c)p(c)}\right)$$

Information Theoretic Mutual Information is non-negative.

C. **JMH Divergence[5]** - This method intends to select features with large discriminative capacity in the classification.

Let *P = {P1, P2, P3, ……,Pn}* be the set of *N* distributions. The Jeffreys-Multi-Hypothesis divergence, denoted by JMH(*P1, P2, P3, ……,Pn)* is defined to be

$$JMH(P_1, P_2, \ldots . P_n) = \sum_{i=1}^{n} KL(P_i, P^c{}_i)$$

where $P^c{}_i$ is the combination of all remaining *N-1* distributions, π_{ki} are the prior coefficients

$$P_i{}^c = \sum_{k=1, k \ne i}^{N} \pi_{ki} P_k$$

$$\pi_{ki} = \frac{p_{c_k}}{\sum_{m=1, m \ne i}^{N} p_{c_m}}$$

p_{c_k} are prior probabilities of each class c_k

IV. TRAINING MODELS

Machine Learning is widely applied in various areas such as; Biological Signature differentiation, search engine, medical diagnosis, bond-market analysis etc. [11].

A. *Naive Bayes* - It is a supervised learning algorithm which uses Bayes theorem with a strong "naive" assumption that features are conditionally independent of each other over class labels, i.e. probability distribution of term t1 is not affected by the value of term t2, given a class label. The features or attributes should independently affect the probability[9]. Despite their "naive" design and strong assumptions, NB classifiers have worked quite well in complex real-world scenarios. In some cases, they outperform Boosted Trees and Random Forests.
If a term is present in test data but not in train data, the classifier sets the probability related to the term as zero. To overcome this issue, Laplace smoothing is used.

B. *Support Vector Machine* - Support Vector Machine is a powerful and versatile machine learning model, capable of performing linear or non-linear classification, regression, and even outlier detection. SVM's are well suited for small/medium sized datasets[Hands on]. Linear-SVM works good even when the feature space is significant because the model looks for the specific hyperplanes among many which best separates data points of different classes and avoids overfitting. SVM uses hinge loss.

C. *Logistic Regression* - Logistic Regression is a supervised machine learning algorithm which estimates the probability of a data point belonging to a specific class using logistic sigmoid function and therefore, maps the data points to the best probable class. Because Logistic Loss diverges faster than hinge loss, it is sensitive to outliers. It is one of the disadvantages of Logistic Regression.

D. *K-Nearest Neighbours[7]* - K-NN is a supervised machine learning algorithm which is based on the principle that the samples which are similar to each other, will lie in close proximity. It is a type of lazy-learning where the function is approximated locally and all computation is deferred until classification. In the classification phase, K is a constant and an unlabelled query point or test point is classified by assigning the

label which is most frequent among K-training samples nearest to the query point. K-NN requires more time for classifying objects when a large number of training examples are given.

V. PERFORMANCE METRICS

To measure the performance of the classifier, a part of the dataset is untouched for testing purpose. One of the most straightforward metrics to use is accuracy which is the percentage of data points that are correctly classified. The problem with accuracy is the overestimation of the classifier performance when the classes are imbalanced. Misclassification of minority classes does not affect accuracy much. F1-score is used as an alternative to accuracy and works even when classes are imbalanced. It is the harmonic mean of precision and recall. The range of F1-score is [0, 1].

$$F_1 = \frac{2 * precision * recall}{precision + recall}$$

A. Precision -: It is the percentage of retrieved documents that are relevant to the query

B. Recall -: It is the percentage of relevant documents which are successfully retrieved

$$precision = \frac{|\{relevant\ documents\} \cap \{retrieved\ documents\}|}{|\{retrieved\ documents\}|}$$

$$recall = \frac{|\{relevant\ documents\} \cap \{retrieved\ documents\}|}{|\{relevant\ documents\}|}$$

C. Performance Evaluation - The following graph shows performance over different term counts. Clearly, JMH Divergence is performing better than remaining feature selection methods. JMH Divergence with top 3000 terms gives highest score. This experiment used Naive Bayes classifier with 5-Fold Cross-Validation test and Macro-based F1-score for comparisons. Here BoW is used as Text-Encoding scheme.

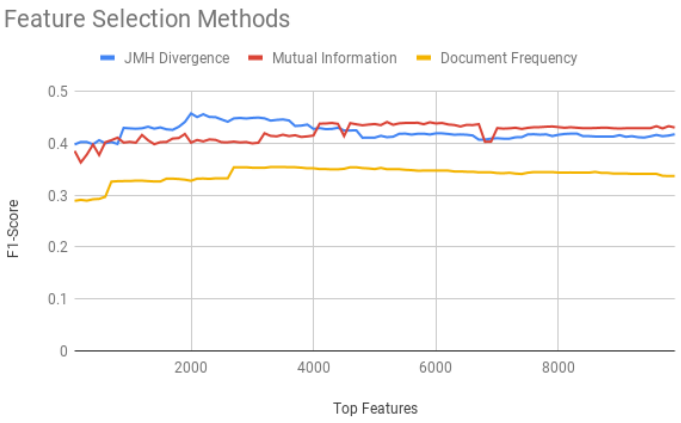

Figure 1. Feature Selection Scheme comparison based on F1-score

Figure 1 represents the feature selection capacity over the data set. F1-score is maximum for top 2000 features by JMH Divergence. Further experiments use these 2000 terms in text encoding along with the one-hot encoded gene and variation feature.

TEXT ENCODING	MODELS					
	NB	SVM(B)	SVM(B)	LR(B)	LR(IB)	K-NN
BoW	0.522	0.507	0.518	0.479	0.515	0.563
TF-IDF	0.567	0.543	0.520	0.525	0.519	0.538
TW-IDF(Degree Centrality)	0.541	0.536	0.521	0.512	0.541	0.502
TF-IDF-CF	0.553	0.542	0.494	0.498	0.508	0.531

Figure 2. F1-Scores of Machine Learning Models on different Text Encoding Schemes

The above scores conclude that both TF-IDF and TW-IDF give better classification results. For TF-IDF-CF, Linear SVM (Class Balancing) is an exception. Most of the configurations are decent at predicting Majority Classes, i.e. ('Likely Loss-of-function', 'Likely Gain-of-function', 'Loss-of-function', 'Gain-of-function'). The problem arise with minority class, i.e. ('Neutral', 'Likely Neutral'). None of the configurations are able to predict class "Likely Switch-of-Function". Linear-SVM (class balanced) + TF-IDF-CF best predicts class 'Neutral' because its f1-score for class 'Neutral' is maximum i.e. 0.6153. Considering the overall performance, Naïve Bayes with TF-IDF configuration scores maximum F1-Score, i.e. 0.567 with a decent 66% accuracy. The F1-Score for mutation class "Neutral" of the configuration (NB+TF-IDF) is 0.476 which is low compared to configuration (Linear-SVM (class balanced) + TF-IDF-

CF) F1-Score which means that (NB+TF-IDF) does not predicts class "Neutral" well.

VI. CONCLUSION

As mentioned in the above four sections, this paper has surveyed different machine learning models namely Naive Bayes, SVM, Logistic Regression, K-NN. Different feature selection schemes have been implemented to use relevant terms. The F1-scores as mentioned in the figures above shows the performance of these model. On comparing scores, JMH Divergence does better than MI and Document Frequency. Overall, NB with TW-IDF-CF encoding scheme gives the best performance at a value of 0.567.
Further improvements are shown in [8] for the future scope. This paper uses original document view, entity text view, an entity name view with domain knowledge to gain better classification performance. The paper uses an ensemble of nine models.

REFERENCES

[1] Chakravarty et.al, "*OncoKB: A Precision Oncology Knowledge Base*", JCO Precision Oncology, pp **1-16**, **2017**
[2] Zheng, "*Feature Engineering for Machine Learning*", O'REILLY Publisher, USA, pp **43-45**, **2018**
[3] M. Liu, "*An improvement of TFIDF weighting in text categorization*", In the Proceedings of the 2012 International Conference on Computer Technology and Science, Hong Kong, pp **44-45**, **2012**
[4] F.D. Malliaros, "*Graph-Based Term Weighting for Text Categorization*", In the Proceedings of the 2015 Advances in Social Networks Analysis and Mining, Canada, pp **1473-1479**, **2015**
[5] Tang, "*Toward Optimal Feature Selection in Naive Bayes for Text Categorization*", IEEE Transactions on Knowledge and Data Engineering, Vol.**28,** Issue.**9**, pp **2508-2521**, **2016**
[6] Y. Xu, "*A Study on Mutual Information-based Feature Selection for Text Categorization*", Journal of Computational Information Systems, Vol.**3**, pp **1007-1012**, **2007**
[7] S.D. Jadhav, "*Comparative Study of K-NN, Naive Bayes and Decision Tree Classification Techniques*", International Journal of Science and Research, Vol.5, Issue.**1**, pp **1842-1845**, **2016**
[8] Zhang et.al, "*Multi-view Ensemble Classification for Clinically Actionable Genetic Mutations",* Springer International Publishing, pp **79-99**, **2018**
[9] R. Nair, "*An Efficient Approach for Sentiment Analysis Using Regression Analysis Technique*", International Journal of Computer Sciences and Engineering, Vol.**7**, Issue.**3**, pp **161-165**, **2019**
[10] Sharma, "*Evaluation of Stemming and Stop Word Techniques on Text Classification Problem*", International Journal of Scientific Research in Computer Science and Engineering, Vol.**7**, Issue.**2**, pp **1-4**, **2015**
[11] P. Rutravigneshwaran, "*A Study of Intrusion Detection System using Efficient Data Mining Techniques*", International Journal of Scientific Research in Network Security and Communication, Vol.**5**, Issue.**6**, pp **5-8**, **2017**

Authors Profile

Varun Saproo is pursuing B.Tech in Computer Engineering from Indus University, Ahmedabad. His area of interests are Machine Learning, NLP, Deep Learning

Rujuta Upadhyay is pursuing B.Tech in Computer Engineering from Indus University, Ahmedabad. Her area of interests are Machine Learning, Cryptography and Network Security.

Manisha Valera has M.E. in Computer Engineering. She is working as an Asst. Professor in Department of Computer Engineering, Indus University. Her area of interests are Big Data, Machine Learning.