Comparative Study between Traditional Machine Learning and Deep Learning Approaches for Text Classification

Cannannore Nidhi Kamath
German Research Center for Artificial Intelligence (DFKI)
University of Kaiserslautern
cannannore_nidhi.narayana_kamath@dfki.de

Syed Saqib Bukhari
German Research Center for Artificial Intelligence (DFKI)
University of Kaiserslautern
saqib.bukhari@dfki.de

Andreas Dengel
German Research Center for Artificial Intelligence (DFKI)
University of Kaiserslautern
andreas.dengel@dfki.de

ABSTRACT

In this contemporaneous world, it is an obligation for any organization working with documents to end up with the insipid task of classifying truckload of documents, which is the nascent stage of venturing into the realm of information retrieval and data mining. But classification of such humongous documents into multiple classes, calls for a lot of time and labor. Hence a system which could classify these documents with acceptable accuracy would be of an unfathomable help in document engineering. We have created multiple classifiers for document classification and compared their accuracy on raw and processed data. We have garnered data used in a corporate organization as well as publicly available data for comparison. Data is processed by removing the stop-words and stemming is implemented to produce root words. Multiple traditional machine learning techniques like Naive Bayes, Logistic Regression, Support Vector Machine, Random forest Classifier and Multi-Layer Perceptron are used for classification of documents. Classifiers are applied on raw and processed data separately and their accuracy is noted. Along with this, Deep learning technique such as Convolution Neural Network is also used to classify the data and its accuracy is compared with that of traditional machine learning techniques. We are also exploring hierarchical classifiers for classification of classes and subclasses. The system classifies the data faster and with better accuracy than if done manually. The results are discussed in the results and evaluation section.

CCS CONCEPTS

• **Information systems** → *Data mining*; *Information retrieval*; • **Computing methodologies** → *Machine learning algorithms*; • **Applied computing** → *Document analysis*;

KEYWORDS

Document Classification, Business Document Analysis, Text Mining, Deep Learning, Machine Learning

DocEng '18, August 28–31, 2018, Halifax, NS, Canada

ACM ISBN 978-1-4503-5769-2/18/08…$15.00
https://doi.org/10.1145/3209280.3209526

ACM Reference Format:
Cannannore Nidhi Kamath, Syed Saqib Bukhari, and Andreas Dengel. 2018. Comparative Study between Traditional Machine Learning and Deep Learning Approaches for Text Classification. In *DocEng '18: ACM Symposium on Document Engineering 2018, August 28–31, 2018, Halifax, NS, Canada.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3209280.3209526

1 INTRODUCTION

From time immemorial, text has been in existence in different forms. With the evolution of human race, texts and documents also saw a steady evolution. With the advent of the Internet era, usage of documents in digital format saw an exponential rise. In-depth and exploratory research in document content analysis led to organizations developing a penchant towards digitized form of documents to reduce the amount of labor, radically saving a lot of time. Gradually, document classification, clustering, categorization, Information extraction and structuring of documents became important in Document engineering. Document classification which is a process of creating a set of models that distinguish class label of the data object[4], is a combination of Natural Language Processing(NLP), Machine Learning, Pattern Recognition,and Statistical theories[1].

Classifying millions of documents into various classes manually is easier said than done. A system which can execute this in lesser time would be an absolute blessing. But, credibility of the system is always debatable. The performance of an algorithm is dependent on multiple factors like the data used, the domain and machine learning algorithm. The extracted features and their representation also play a major role in the accuracy given by a classifier[3][7]. Considering all these factors, we have compared the accuracies obtained by few traditional machine learning algorithms with a deep learning algorithm to zero in on a more reliable system.

Two datasets with multiple classes were used for the study-Proprietary and Public datasets. Study was conducted with 5 traditional machine learning techniques namely Logistic Regression, Support Vector Machine, Multinomial Naive Bayes and Random Forest Classifier. We have also used Multilayer Perceptron, which is a class of artificial neural network. We have used both raw and processed documents for the study. Accuracies obtained for each one of the algorithms for raw and processed data were noted down and the same process was repeated for a deep learning algorithm like Convolution Neural Network.

We have also explored using a hierarchical convolution neural network for the proprietary dataset which has hierarchy of classes.

Theoretical background of the techniques used are discussed in Section 3. Section 4 explains in detail the datasets being used.

Section 5 talks about the experimental setup and Section 6 sheds light on evaluation of results. The conclusion is penned down in Section 7.

2 RELATED WORK

Thornton has noted that not just humans but also intelligent systems learn through experiences[2]. Best training approaches for data classification has always been an intriguing research area. Exhaustive research has been conducted earlier comparing different text mining algorithms.

Document classification has been studied in combination with multiple algorithms and datasets derived from various fields. Naive Bayes was found to be the most effective algorithm in [5], where machine learning algorithms such as Support Vector Machine, Logistic Regression, Artificial Neural Network and Naive Bayes was used to categorize products. The data for this study was taken from three online shopping websites.

In our study, we consider only traditional machine learning techniques such as Support Vector Machine, Naive Bayes and Logistic Regression. Along with these, we are also using Random forest Classifier[11] and Multi-layer Perceptron. Each of these techniques are used for classification of two types of datasets into several classes and their accuracies are noted down.

In [8], Kim used Convolution Neural Network, a deep learning technique, which is mainly used for computer vision. Other than computer vision, Convolution Neural Network has been giving very good results in multiple Natural Language processing techniques[10]. Publicly available word2vec vectors datasets are used as inputs to the Convolution Neural Networks in Kim's study[8].

We are using the idea proposed by Kim[8] with few modifications. Instead of using pre-trained word embeddings, we create our own embeddings. In [9], it was found that constraints do not have much effect on the accuracy of the convolution neural network. Hence, we are not using L2 norm constraints on the weight vectors.

We extend our study to an Hierarchical Convolution Neural Network. One of our datasets has multiple hierarchical class structure, so we build a classifier which classifies the data to its parent class, examines the subclasses and classifies them accordingly.

3 CLASSIFICATION ALGORITHM

This section sheds light on the theoretical background of the traditional machine learning and deep learning algorithms used in this study.

Random Forest Classifier

Random forest algorithm can be used for both regression and classification but in our study we are only taking classification into account. A Random Forest is a classifier consisting many decision tree classifiers $h(x, y_k)$ where k = 1,2, . . . y_k are random vectors. Decision of the best suited x is made by each decision tree [12]. Every decision tree classifier takes part in deciding the best suited x for the system. Tree structures built on "if this than that" condition. The first step is selecting n features out of m features randomly. In the chosen n features, root node is found using best split point and it is split into child nodes. Thus the tree grows in size to a pre-decided value of Ntrees and depth of the tree is decided by node size. The same method is repeated to build the forest structure.

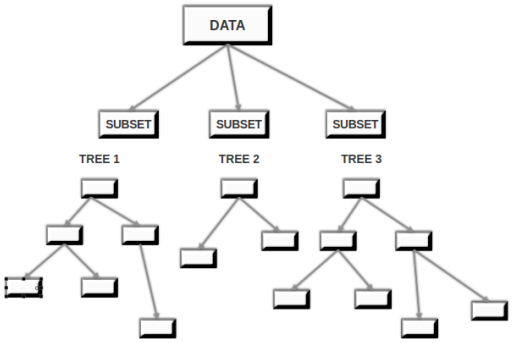

Figure 1: A Random Forest Classifier with multiple decision trees.

Figure 1 shows a random forest classifier with multiple decision trees.

All the decision trees built are trained to classify the instances. A new instance is taken into account by all decision trees, each of them gives its classification output as a vote. The instance is identified as belonging to a class on the maximum number of votes obtained by a class.

Random forest classifier has an advantage over decision trees as it eliminates the chances of overfitting.

Support Vector Machine (SVM)

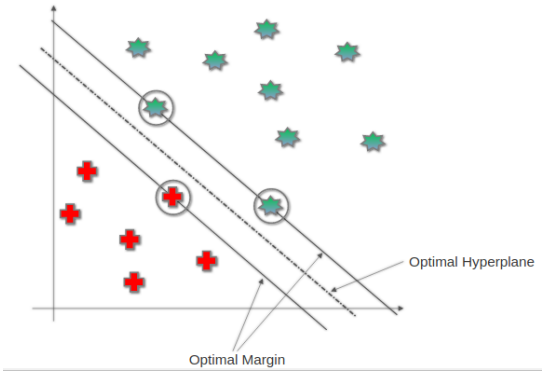

Figure 2: Illustration showing the optimal margin and optimal hyperplane. Support vectors in the circles define the margin of the largest separation between classes[13].

Support vector machine is a supervised learning algorithm which converts text documents to vector form. Number of keywords is the dimension of the vector. It is efficient in dealing with sparse and robust datasets[13].

Statistical learning theory and structural risk minimization principal form the base of Support Vector Machine. It estimates the maximum linear distance between various classes in the feature space. It determines the hyperplane which is the location of the decision boundaries that produce the best result for separating classes. This technique is called as "maximal-margin-hyper-plane"[13]. When SVM comes across the non-linearity between the classes, it maps the non-linearity of classes and the feature map[15]. This mapping is done by 'kernels'. The output of this is a hyperplane which provides direct mapping to non-linear structure in the feature space. Illustration of such hyperplane can be seen in Figure 2.

SVM was designed for binary classification. When SVM is applied for a multi-class classification problem it internally breaks the task into multiple binary classier problems and solves them using many SVMs[14].

Logistic Regression

Logistic Regression is a supervised learning algorithm used for classification. The model created by this algorithm is based on logistic function[16][17]. A logistic function, also referred to as a logistic curve(,) is a sigmoid curve(Figure 3) with the below equation

$$f(x) = \frac{M}{1 + e^{-k(x-x_0)}} \tag{3.1}$$

e is the Euler's number, x_0 is the x-value of the sigmoid's midpoint, M is the curve's maximum value and s is the steepness of the curve.

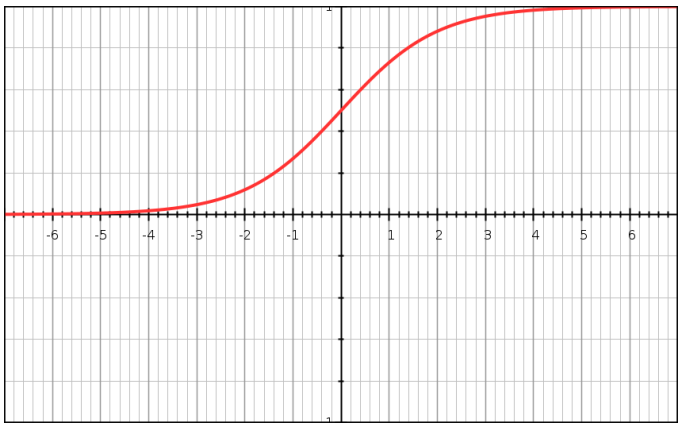

Figure 3: A Logistic Curve for the equation 3.1.

Logistic Regression uses predictor variables to classify the categorical dependent data. It converts the categorical dependent variable to probability scores, thus measuring the relationship between categorical dependent variable and a continuous independent variable. Two types of logistic regression are Binary Logistic Regression and Multinomial Logistic Regression. The types of labels used in them is the prime difference between these types. If the labels are multiple values, multinomial logistic regression is used[18]. The basic assumption for this model is that it does not consider that dependent and independent variables have a linear relationship[19]. The dependent variable usually falls into two categories and independent variables may not be interval, normally distributed and linearly related.

Naive Bayes

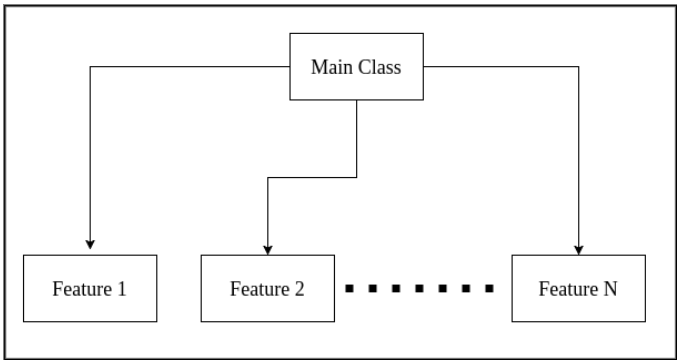

Figure 4: Conceptual illustration of Naive Bayes.

Naive Bayes classifier is a supervised learning algorithm based on Bayesian theorem[21]. Bayes theorem calculates the posterior probability P(A|B). The equation is given below

$$P(A \mid B) = \frac{P(B \mid A)\, P(A)}{P(B)} \tag{3.2}$$

P(A|B) is the posterior probability of the target class when the predictor, which is the attribute of the class is known. P(A) is the class prior probability, P(B|A) is the likelihood which is the probability of predictor of given class and P(B) is the predictor prior probability. Figure 4 shows the conceptual illustration of Naive Bayes[6].

Bayesian networks belong to probabilistic graphical models. In Bayesian networks, nodes represent a random variable and edges connecting the nodes are the probabilistic dependencies of the random variables respectively.

Bayesian networks assumes that the features are independent of each other[20]. It calculates occurrences of items or posterior probabilities and assigns document to the class with the highest posterior probability.

There are multiple types of Naive Bayes classifier. In Gaussian Naive Bayes likelihood of features are assumed to be Gaussian, i.e when the data is continuous, it is assumed that the continuous values associated with the classes follow Gaussian distribution.

$$p(A_i \mid B) = \frac{1}{\sqrt{2\pi\sigma_B^2}}\, e^{\left(-\frac{(A-\mu_B)^2}{2\sigma_B^2}\right)} \tag{3.3}$$

The equation 3.3 shows the calculation of Gaussian Naive Bayes[22]. The parameters $\sigma_B$and μ_B are estimated using maximum likelihood.

Another type of Naive Bayes algorithm is the Multinomial Naive Bayes. In this method Naive Bayes algorithm is applied on multinomial distributed data. The parameters θ_y is estimated by a smoothed version of maximum likelihood, i.e. relative frequency counting:

$$\hat{\theta}_{xi} = \frac{R_{xi} + \alpha}{R_x + \alpha k} \tag{3.4}$$

where $R_{xi} = \sum_{x \in T} x_i$ is the number of times feature i appears in a sample of class y in the training set T, and $R_x = \sum_{i=1}^{|T|} R_{xi}$ is features count for class x [22].

There may be a situation when in a training data, a class and feature value never occurs together, making frequency-based probability null, which eliminates all other probabilities on multiplication. Hence a sample correction called pseudocount is taken in all probabilities so that the final value does not become zero. When the pseudocount is set to one to regularize Naive Bayes, it is called Laplace smoothing and Lidstone smoothing when pseudo count is less than 1[23][24].

Setting $\alpha = 1$ is called Laplace smoothing, while $\alpha < 1$ is called Lidstone smoothing.

Bernoulli Naive Bayes is used to implement Naive Bayes on the data that is distributed according to multivariate Bernoulli distributions[23]. In BernoulliNB features are represented as binary valued feature vectors. The decision rule for Bernoulli naive Bayes is based on

$$P(a_i \mid b) = P(i \mid b)a_i + (1 - P(i \mid b))(1 - a_i) \tag{3.5}$$

the non-occurrence of a feature i is penalized which is an indicator for class b which makes BernoulliNB different from MultinomialNB.

Multi-layer Perceptron (MLP)

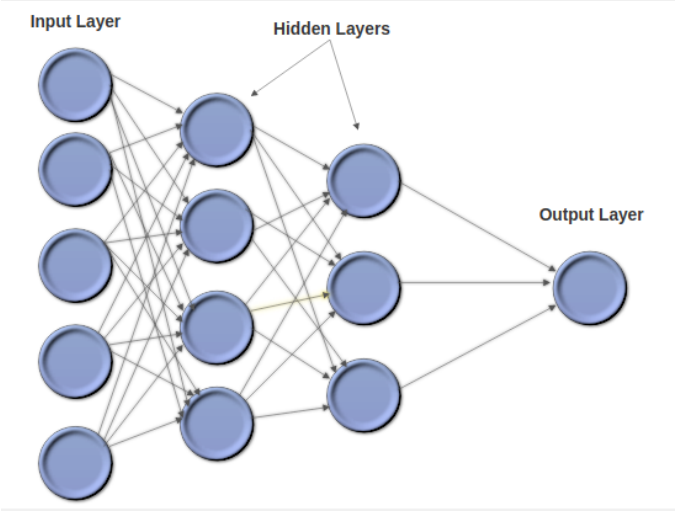

Figure 5: An Example for a Multi-Layer Perceptron which has a input layer, two hidden layers and output layer.

Multi-layer Perceptron is a class of Artificial neural networks, which uses a supervised learning method known as backpropagation for classification[25][26]. It has three or more layers as shown

in Figure 5. The input layer has neurons which represent the input features. Next comes the hidden layer, it takes the input from the first layer and applies weighted linear summation and a non-linear activation function on them[22]. Common activation functions are sigmoids shown by the below function

$$x(a_i) = \tanh(a_i) \tag{3.6}$$

Equation 3.6 is a hyperbolic tangent and 3.7 is logistic function[22].

$$x(a_i) = (1 + e^{-a_i})^{-1} \tag{3.7}$$

Activation functions are taken as input by the next layer, which is the output layer. Since MLP is fully connected, the nodes in multi-layers are connected with each other through weights. When each set of data is processed, the connection weights are changed based on the amount of error difference between the expected result and output. This learning technique is referred to as backpropagation.

Deep Learning : Convolution neural network

Convolution neural network is a deep learning technique which is widely used in computer vision. Convolution neural network is built on the basic idea of applying a sliding window over a matrix.

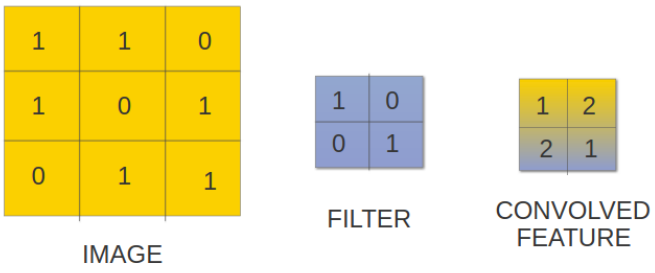

Figure 6: Convolution of a 3*3 image where 0 represent black color and 1 represent white color. A 2*2 filter is used to get the Convolved feature.

If the image on the left in Figure 6 is considered as a black and white image, with 0 for black pixels and 1 for white pixels, then when a 2*2 sliding window is moved over the image, its values are multiplied element wise and they are summed up. This technique is done for each element to get the whole convolution of the full matrix[27].

In a convolution neural network, convolutions are applied on each layer to get the output. Every layer applies not just one but multiple filters and combines their results by pooling. Pooling can be done on an entire matrix or a sliding window. *max* operation of pooling is frequently used, where maximum value from a window or matrix is taken to the next layer[10]. CNN uses non linear activation functions such as tanh or ReLU.

In text classification, documents are represented in the form of a matrix. Input layer consists of a matrix with each row in the matrix being a vector of word embeddings. In the convolution layer, sliding window or filters of multiple sizes slides over the rows of matrices. After this is the max pooling layer and final layer is a softmax classifier. Figure 7 Illustrates all the steps of a Convolution neural network.

4 DATASETS

Datasets used in any study are of utmost importance as they influence the accuracies of the study[3]. We are using two datasets in our study: a proprietary and a public dataset.

First one is dataset named Health from an insurance company. It contains images of various invoices received by the insurance company. The dataset contains 6 parent classes as outpatient, dental, medical products, cure, medical aids and inpatient. Few of these classes branch out to having subclasses. Altogether we have 18 classes in this dataset and almost 13500 documents combined. The tree structure of health dataset is illustrated in Figure 8.

The second dataset used is publicly available Tobacco-3482 dataset. It consists of images related to tobacco from the media. It has 9 classes and 2800 documents combined.

5 EXPERIMENTAL SETUP

In this section we discuss the implementation of the algorithms explained in the section 3. Data preparation talks about how data is being separated to raw and processed data along with all the processing steps. In Classification using Traditional Machine Learning Approaches and Classification using Convolution Neural Network, we discuss in depth about training the classifier and testing them. In addition to this, Classification using Convolution Neural Network also talks about hierarchical CNN.

Data Preparation

Preparation of data is the first and significant step in this study. The dataset under consideration has labeled data in the form of images with .png extension. Images are converted to text files using tesseract, which is a software which recognizes text characters in an image with the aid of its language models. It returns the text form of the image in a .txt file which are called OCRs.

Data files are split for train and test with the ratio 4:1. Twenty percent of the data is kept untouched and used only for testing and other eighty percent is used for training. One copy of the training data is kept as it is without any modifications to them. This data is considered as raw data. Next the data is processed by removing the punctuations. Stop words are removed and stemming is applied on them. Stopwords are identified by importing the language specific stopwords from nltk corpus. An algorithm from computational linguistics named Porter-Stemmer algorithm is used for stemming. The algorithm applies truncation rules to get the stemmed words. This is the processed data used for the classification. Figure 9 illustrates all these steps.

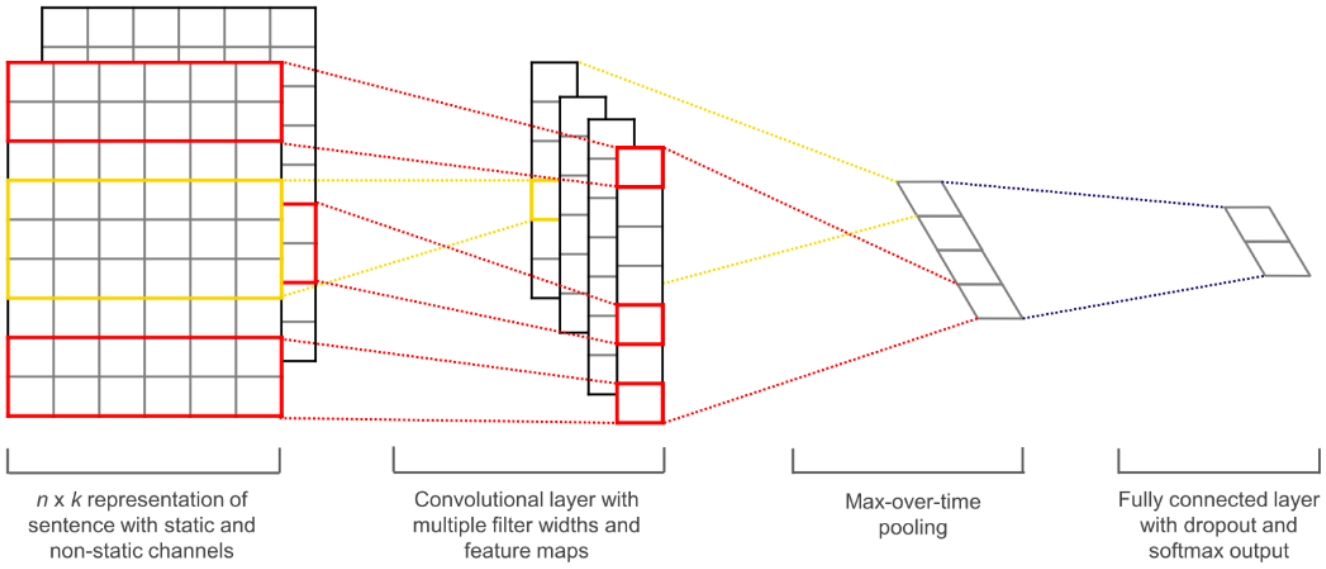

Figure 7: Illustration of a Convolutional Neural Network (CNN) architecture from [8]. n*k representation of a sentence, Convolution layer with multiple filters and feature maps, Max-pooling and Fully connected output layer with dropout and softmax are shown in detail.

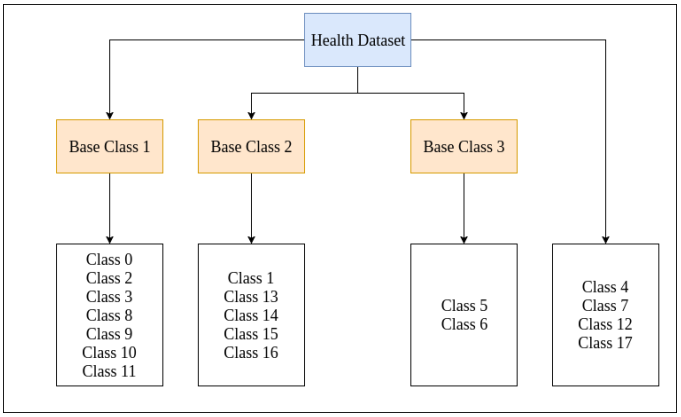

Figure 8: Structure of Health dataset. The child classes and the base class they belong to are shown.

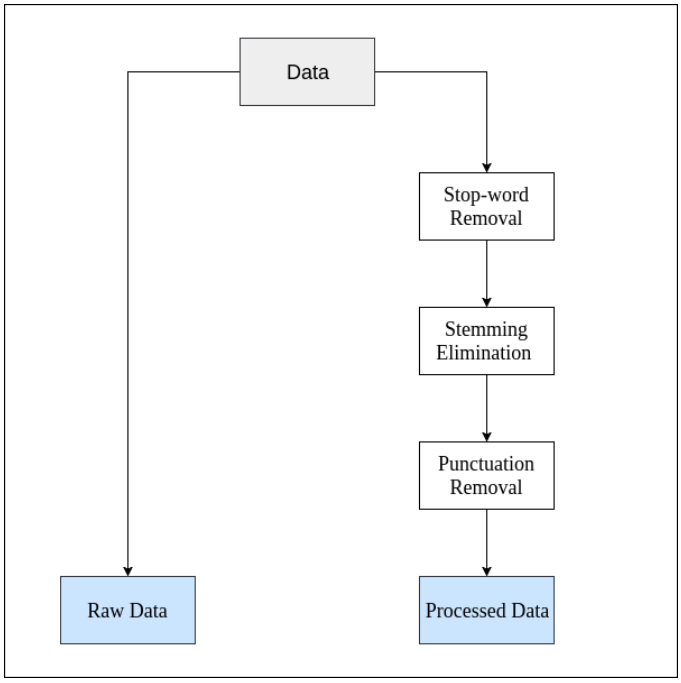

Figure 9: Datapreparation. Complete stepwise procedure of preparing the processed data is shown.

Classification using Traditional Machine Learning Approaches

Figure 10 illustrates the stepwise procedure for training the model using traditional machine learning algorithms. The classification of documents happens in this step. We set the ground truth of exact class which the documents belong to based on their labels. Text documents are converted into collection of vectors with the length of the entire vocabulary and an integer count for the number of times each word appeared in the document. They usually end up being sparse with the presence of lots of zeros in them. They are made dense and word features are extracted from them by a process where they learn the vocabulary dictionary and return the

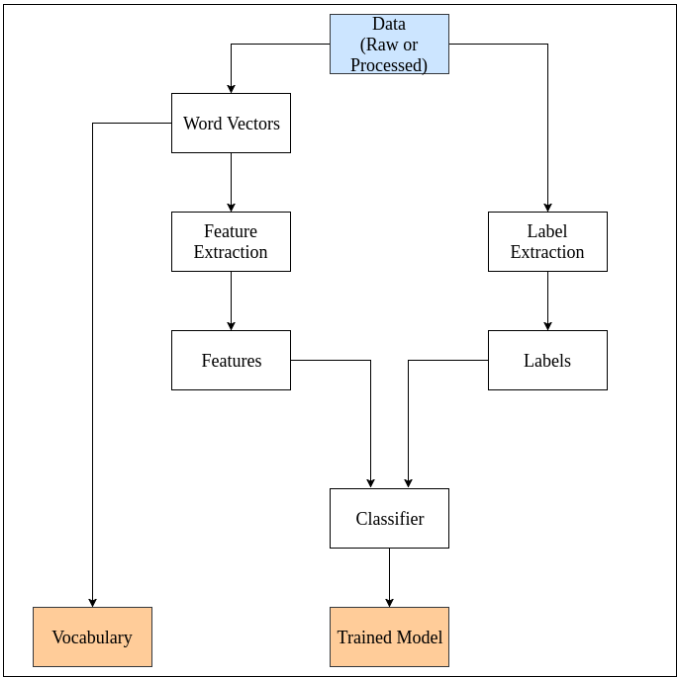

Figure 10: All the steps involved in training a model using traditional machine learning algorithms. Feature extraction from word vectors, label extraction, feeding the features and labels to the classifier for training are clearly illustrated. Output is the vocabulary and trained model.

count vectors. These feature vectors and labels are inputted to the classifier, which trains a classifier model. Building a classifier model follows a slightly different approach for every algorithm. Each of the classifiers are trained independent of each other using different algorithms and are tested separately. Each of the approaches are explained below.

Training SVM Classifier : Two flavors of Support Vector Machines were used, one being SVC with kernel as linear and other one being LinearSVC. Both of these take array of training samples and class labels as input. SVC implements "one-against-one" approach for multi-class classification where for x classes, $x*(x-1)/2$ classifiers are constructed. Each classifier trains data from 2 classes whereas LinearSVC follows "one-against-rest" approach, where for n classes, n classifiers are created[22]. Both classifiers have been trained separately and the models are saved for testing. The parameter values for SVC was kernel being linear and for linear SVC was random_state set to 0.

Training Logistic Regression Classifier : In Logistic Regression, the default "one-vs-rest" approach is implemented to train a classifier model. Word feature vectors and labels are taken as input by the algorithm which trains and fits the model according to the given training data.

Training Naive Bayes Classifier : While considering Naive Bayes, we build 3 separate classifiers which are Gaussian Naive Bayes, Multinomial Naive Bayes and Bernoulli Naive Bayes. Gaussian Naive Bayes implements the Gaussian Naive Bayes algorithm, Multinomial Naive Bayes implements the naive Bayes algorithm for multinomial distributed data and Bernoulli Naive Bayes implements the naive Bayes algorithm for data which is distributed according to multivariate Bernoulli distributions. All of them take word feature vectors and labels as input and train the classifier model.

Training Random Tree Classifier : Random tree classifier takes word feature vectors and labels as inputs and creates multiple decision tree classifiers on the samples of dataset. It improves the accuracy and controls overfitting by averaging the decision tree classifiers. Thus it trains an optimal model. The parameter random_state was set to 0.

Training Multi-layer Perceptron Classifier : Multi-layer perceptron also takes word feature vectors and labels as inputs but it is different from other supervised learning algorithms from the fact that it has multiple layers and uses backpropagation. For a multi-class classification problem, it applies softmax as output function. The training is done by a gradient descent which is calculated by backpropagation. In this study ReLU has been used as activation function for the layers. The final model is saved for training. The parameters used are as follows: activation function was relu, solver was set to lbfgs, alpha had the value of 1e-5, hidden_layer_sizes were 15, and random_state was set to 1.

Testing : Text documents in the test data sets are also converted into collection of vectors by the same strategy applied on the training dataset. The features extracted from them are tested against the trained model, which outputs the labels of the documents. The output is compared with the ground truth to get the accuracy for each model. The complete flow of testing is shown in Figure 11.

Classification using Convolution Neural Network

Training : The training approach is same for both raw and processed data sets. All sentences are padded to maximum sentence length. All sentences are converted to vectors of integers by mapping each word in the vocabulary index of the dataset to an integer.

The data is divided into training and evaluation data in the ratio specified by dev-sample-percentage parameter. Words are embedded into low-dimensional vectors in the first layer. Word embeddings are input to the next layer which performs convolutions over them using multiple filters. In this study, filters with size 3,4 and 5 were used. In the next layer a long feature vector is generated by max-pooling the output from the convolution layer. Pooling is required because we use multiple filters of different sizes which result in matrices of different shapes. During pooling, they are all iterated and layers for each one of them are created and merged to form a feature vector. Dropout regularization is applied to the feature vector and the results are classified by softmaxing.

Multiple hyper-parameters are used while training the convolution neural network. Number and sizes of convolution filters, pooling strategies (max, average), and activation functions (ReLU, tanh) are few of them. Fine tuning of these parameters helps in getting better accuracy[10]. Number of epochs, batch sizes for each batch of training, evaluate-every and checkpoint-every are the training parameters. The model is evaluated on evaluation data

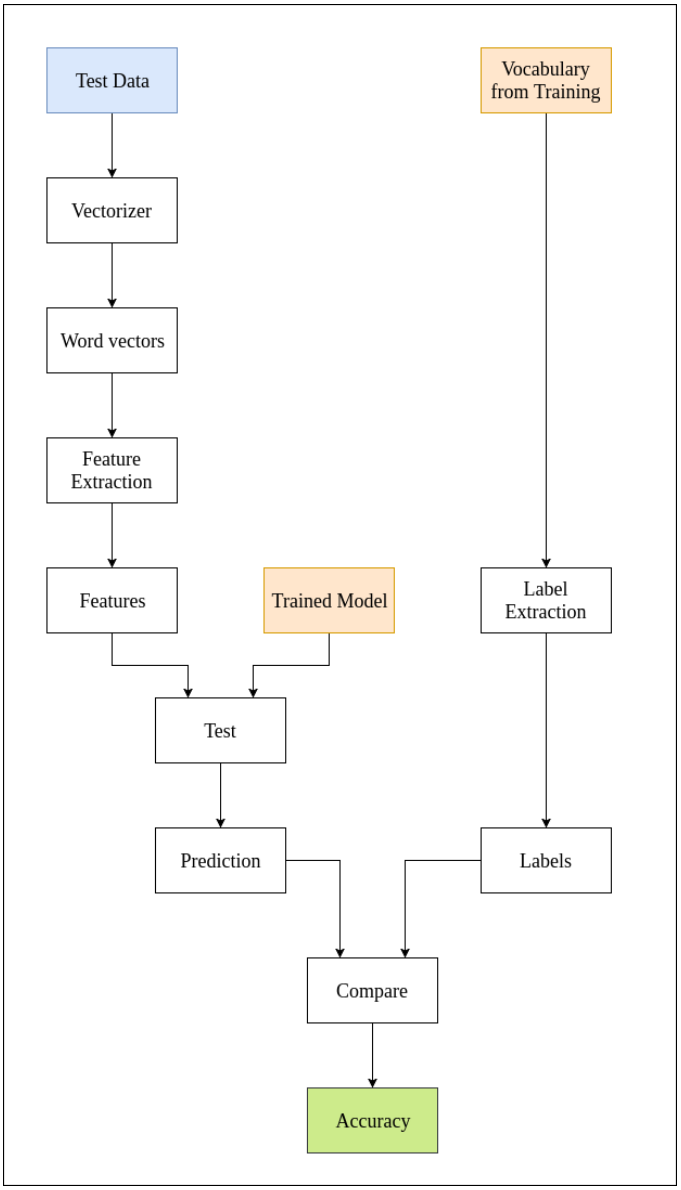

Figure 11: The flowchart illustrates all the steps involved in testing the trained model with test data. Method used to predict the accuracy is also shown.

after number of steps specified by evaluate-every parameter. The model is saved after number of steps specified in checkpoint-every parameter and the training is continued. These saved models are used in testing.

Training is done by initially loading the data sets and constructing the ground truth by extracting their labels. After which the loaded data is used to build vocabulary vectors. The data is then shuffled and split according to dev-sample-percentage parameter into training and evaluation data. Batches of data are generated for training and evaluation. Training the model starts by considering all the training parameters and hyper parameters. The training accuracy and loss are saved, also the model is saved after number of steps specified in checkpoint-every parameter. Figure 12 illustrates a CNN architecture for sentence classification[8]. The parameter values used in this study are as follows: dev_sample_percentage was set to .10, embedding_dim was 128, filter_sizes were 3,4 and 5, num_filters was 178, dropout_keep_prob was 0.5, l2_reg_lambda was 0.0, batch_size was 63, evaluate_every was 100, checkpoint_every was 100, num_checkpoints was 5, allow_soft_placement was set to True and log_device_placement was False.

Hierarchical CNN : In the earlier section, the convolution neural network built was taking all the classes as input and building the model. Even though the health dataset had hierarchical classes, hierarchy was not taken into consideration. All the classes were considered at the same level of hierarchy.

In the new system for convolution neural network for hierarchical classifiers, we are building multiple classifiers to classify the data of hierarchical classes in health dataset as shown in Figure 13. A classifier is trained to classify data into three base classes and other four classes which do not have any child classes. Three separate classifiers are trained to classify data to subclasses for each of these base classes.

While testing, the base classifier model is first invoked which branches out to other classifiers. The accuracy of the systems depends primarily on the base classifier.

Advantage of this system is that it takes lesser time to train the classifiers for child classes as the load balancing happens.

Testing : Test data is loaded and their labels are extracted to set the ground truth. Loaded data is used to build vocabulary vectors. The saved model checkpoints during training are loaded and the test data is validated with them and the ground truth. The new predictions are obtained as output along with the accuracy of predictions.

6 RESULTS AND EVALUATION

In this section, accuracies obtained by all the algorithms for both datasets are discussed.

Initially, We applied two flavors of Support vector Machine, on Tobacco-3482 and Health dataset for both raw and processed data. The accuracies obtained by them are shown in Table 1. All accuracies are in percentages. It was observed that raw data gave better accuracies than processed data. It was observed that LinearSVC had an upper hand over SVC with its accuracies being higher with a slight margin.

Table 1: Comparison of accuracies in percentage obtained by two flavors of SVM, namely SVC and LinearSVC for health and tobacco datasets.

	Tobacco Dataset		Health Dataset	
	Raw	Processed	Raw	Processed
SVC	69%	57%	77.1%	69.94%
LinearSVC	69.86%	64%	78.11%	73.11%

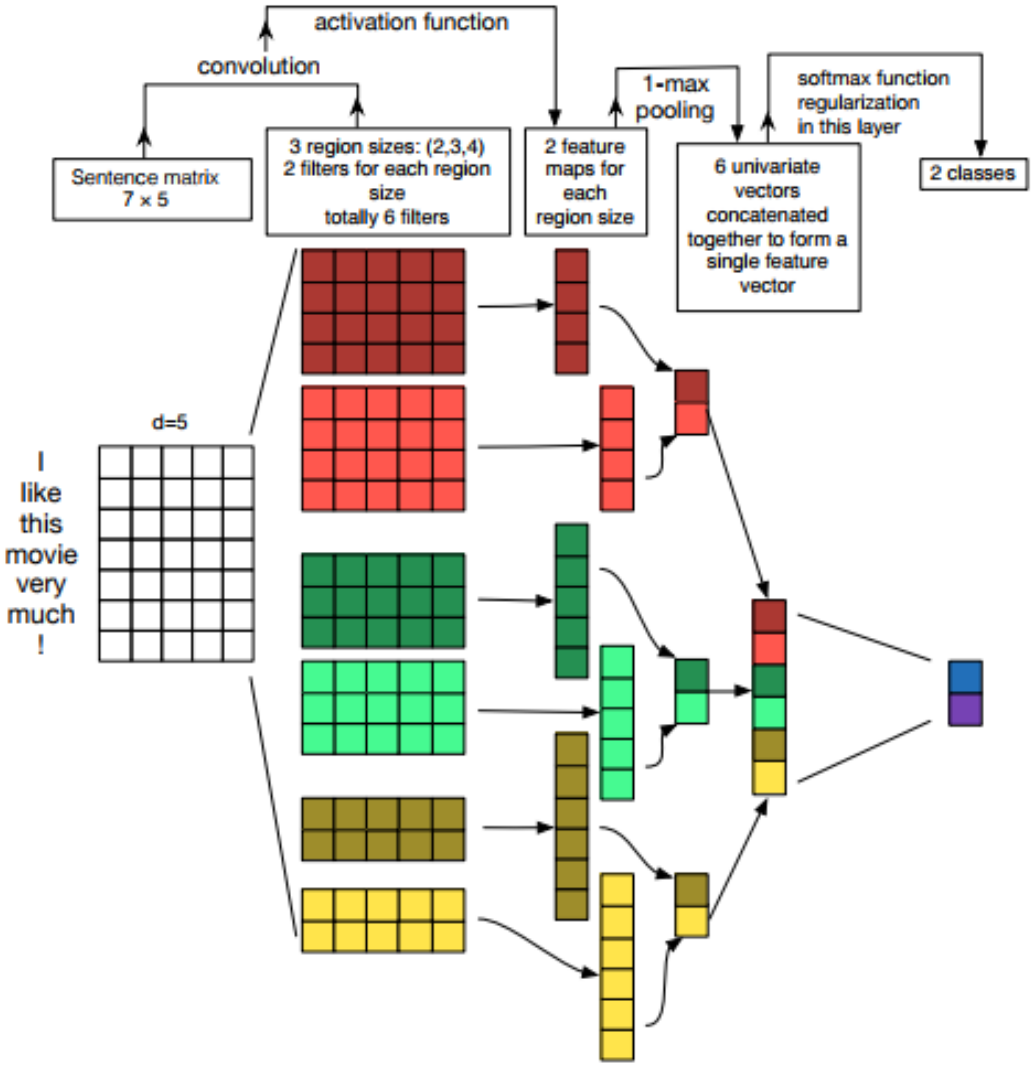

Figure 12: Illustration of a CNN architecture for sentence classification[8].

In Naive Bayes, We tried all three flavors of Naive Bayes where Gaussian Naive Bayes, Multinomial Naive Bayes and Bernoulli Naive Bayes were trained against raw and processed data for both Health and tobacco datasets. The accuracies obtained by them are shown in Table 2. Again, classifiers for raw data had higher accuracies than that for processed data. Gaussian Naive Bayes classifier had the least accuracy for both the datasets. Multinomial Naive Bayes and Bernoulli Naive Bayes being very good classifiers for text classification produced good accuracies but Multinomial Naive Bayes had upper hand over Bernoulli Naive Bayes in both the datasets.

Table 2: Comparison of accuracies in percentage obtained by three flavors of Naive Bayes, namely Gaussian Naive Bayes, Multinomial Naive Bayes and Bernoulli Naive Bayes for health and tobacco datasets.

	Tobacco Dataset		Health Dataset	
	Raw	Processed	Raw	Processed
Gaussian NB	42.28%	38%	66.53%	60.5%
Multinomial NB	68.73%	58%	77.27%	75%
Bernoulli NB	58.64%	49.2%	75.22%	73.5%

As LinearSVC and Multinomial Naive Bayes have better frequencies than their other flavors, We compare these with other traditional machine learning algorithms. Table 3 shows the comparison of all traditional machine learning algorithms with convolution

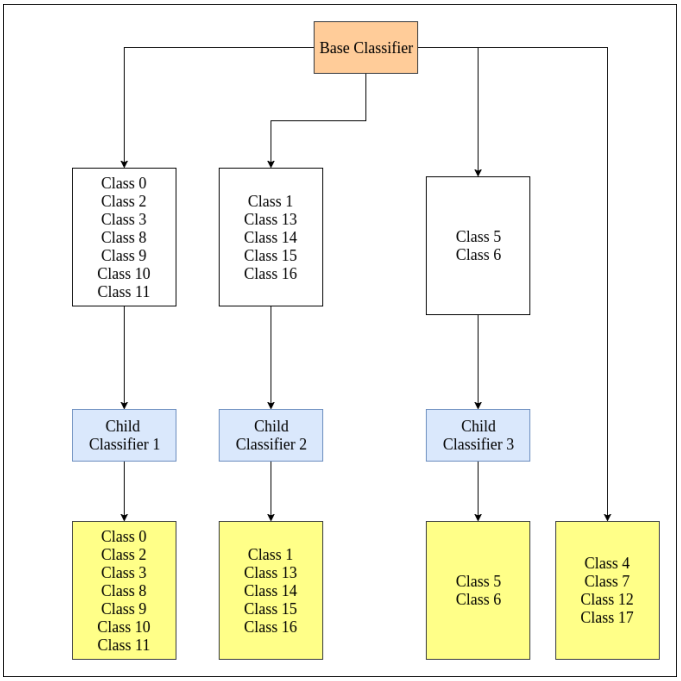

Figure 13: Hierarchical Convolution neural network. The base class passes few classes to the child classifier and the child classifier classifies them accordingly.

neural network. It can be noted that random forest classifier has the least accuracy percentage with both processed and raw data, followed by Naive Bayes, SVM and MLP almost having their accuracies in the same range. It can be clearly seen that Logistic Regression has the best accuracy amongst all the traditional machine learning algorithm.

When the accuracies of Logistic Regression are compared with that of convolution neural network, it can be noted that convolution neural network is a clear winner with accuracies being way higher than all other traditional machine learning algorithms. Another observation which can me made from the table is that the accuracies of traditional machine learning algorithms are better for raw data than that of processed data in both the sets whereas convolution neural network has better accuracies for processed data than that of raw data. This is due to the presence of deeper layers and also the feature vectors used in the convolution neural networks took context of the text into consideration and they were also continuous vectors.

As convolution neural network proved the best suite for these datasets we tried the convolution neural network by preserving the class hierarchies. Four classifiers were built and tested against the Health dataset(Figure 13), as tobacco dataset did not have any hierarchy in its classes. All the classifiers for child classes gave very high accuracies but the base classifier produced an accuracy which was way lower than that of the convolution neural network with all classes at the same level of hierarchy. The accuracies are compared in Table 3. As the base class accuracy is the deciding

Table 3: Comparison of accuracies in percentage obtained by traditional machine learning algorithms with convolution neural network for health and tobacco datasets.

	Tobacco Dataset		Health Dataset	
	Raw	Processed	Raw	Processed
LinearSVC	69.86%	64%	78.11%	73.11%
Multinomial NB	68.73%	58%	77.27%	75%
Logistic Regression	73%	67%	81%	77%
Random Forest Classifier	60%	50%	73%	67%
Multi-layer Perceptron	68.73%	61%	76%	68%
Deep Learning : CNN	82%	96%	84%	89.27%

factor for hierarchical system, a lesser accuracy at this level would effect the accuracies for other levels adversely. The factor which we assume could be effecting the base class accuracy would be that the subclasses under a base class would not have similar features, which made learning difficult. Multiple convolution neural network classifiers preserving the class hierarchies would be best suited for a system where subclasses are clubbed under a base class based on the similarity of their features and not from the organizational factors.

7 CONCLUSION

In this paper we have compared the performance of multiple flavors of different traditional machine learning algorithms with that of the accuracy produced by the convolution neural network(a deep learning technique). Two types of datasets were used in the study. One is public tobacco-3482 dataset and other one is the health dataset from corporate sector. It has been noted that Logistic Regression produced the best accuracy among the five traditional machine learning techniques. But the deep learning technique produced better accuracy than Logistic Regression on both datasets. Another point which the paper threw light on was that the traditional machine learning algorithms produced better accuracies with raw data whereas deep learning algorithm produced better accuracies with processed data for both datasets. It can also be noted that the deep learning technique for hierarchical classes produce better accuracies when the subclasses have similar features.

REFERENCES

[1] T. M. Mitchell: Machine learning.Burr Ridge, IL: McGraw Hill, 1997.
[2] J. Thornton, "Techniques In Computational Learning", Chapman and Hall, London, 1992.
[3] Jurafsky D, Martin JH: Speech and Language Processing. Pearson Education India, 2000.
[4] Archana Chaudhary,Savita Kolhe, Rajkamal, "Machine Learning Techniques for Mobile Intelligent Systems: A Study", IEEE International conference on Wireless and Optical Communications Networks, ISBN 978-1- 4673-1988-1, 2012
[5] Chanawee Chavaltada, Kitsuchart Pasupa, David R. Hardoon, "A Comparative Study of Machine Learning Techniques in Automatic Product Categorisation", In Proceeding of the 14th International Symposium on Neural Networks (ISNN 2017), 21-23 June 2017, Hokkaido, Japan (Fengyu Cong, Andrew Leung, Qinglai Wei, eds.), vol. 10261, pp. 10-17, 2017.
[6] A Comparative Study of Machine Learning Methods for Verbal Autopsy Text Classification. / Danso, Samuel; Atwell, Eric; Johnson, Owen. In: International Journal of Computer Science, 18.02.2014.

[7] P. Strecht, L. Cruz, C. Soares, J. Mendes-Moreira, and R. Abreu, "A Comparative Study of Classification and Regression Algorithms for Modelling Students' Academic Performance," in International Educational Data Mining Society, 2015, pp. 392-395.

[8] Kim, Y. (2014). Convolutional Neural Networks for Sentence Classification. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP 2014), 1746-1751.

[9] Yih, Wen-tau et al. "Semantic Parsing for Single-Relation Question Answering." ACL (2014).

[10] Zhang, Ye and Byron C. Wallace. "A Sensitivity Analysis of (and Practitioners' Guide to) Convolutional Neural Networks for Sentence Classification." CoRR abs/1510.03820 (2015): n. pag.

[11] Xu, Baoxun et al. "An Improved Random Forest Classifier for Text Categorization." JCP 7 (2012): 2913-2920.

[12] Breiman L, Random Forests, Machine Learning, 45, 5-32, (2001).

[13] VN Vapnik: An overview of statistical learning theory. Neural Networks, IEEE Transactions on, 1999, 10:988-999.

[14] G Madzarov, D. Gjorgievikj and I. Chorbev, " A Multi-class SVM Classifier Utilizing Binary Decision Tree", Informatica, pp. 233-241 (2009).

[15] Isabelle Guyon, Bernhard E. Boser, and Vladimir Vapnik. 1992. Automatic Capacity Tuning of Very Large VC-Dimension Classifiers. In Advances in Neural Information Processing Systems 5, [NIPS Conference], Stephen Jose Hanson, Jack D. Cowan, and C. Lee Giles (Eds.). Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 147-155.

[16] https://hal.inria.fr/hal-00860051/document

[17] Aaron Defazio, Francis Bach, Simon Lacoste-Julien. SAGA: A Fast Incremental Gradient Method With Support for Non-Strongly Convex Composite Objectives. Advances In Neural Information Processing Systems, Nov 2014, Montreal, Canada. <hal-01016843v3>

[18] Yuth, K.: Principle and using logistic regression analysis for research. RMUTSV Res. J. 4(1), 1-12 (2012)

[19] Qingshan Ni, Zheng-Zhi Wang, Qingjuan Han,Gangguo Li, Xiaomin Wang, Guangyun Wang, "Using Logistic Regression Method to Predict Protein Function from Protein-Protein Interaction Data", ICBBE 2009. 3rd International Conference on Bioinformatics and Biomedical Engineering, E-ISBN 978-1-4244-2902-8,2009

[20] Aurangzeb Khan, Baharum Baharudin, Lam Hong Lee,"A Review of Machine Learning Algorithms for Text-Documents Classification", Journal Of Advances In Information Technology, February 2010.

[21] Lewis, D.D.: Naive (Bayes) at forty: the independence assumption in information retrieval. In: NÃĺdellec, C., Rouveirol, C. (eds.) ECML 1998. LNCS, vol. 1398, pp. 4-15. Springer, Heidelberg (1998). doi:10.1007/BFb0026666

[22] Scikit-learn: Machine Learning in Python, Pedregosa et al., JMLR 12, pp. 2825-2830, 2011.

[23] McCallum, Andrew; Nigam, Kamal (1998). A comparison of event models for Naive Bayes text classification (PDF). AAAI-98 workshop on learning for text categorization. 752.

[24] Rennie, J.; Shih, L.; Teevan, J.; Karger, D. (2003). Tackling the poor assumptions of Naive Bayes classifiers

[25] Rosenblatt, Frank. x. Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms. Spartan Books, Washington DC, 1961

[26] Rumelhart, David E., Geoffrey E. Hinton, and R. J. Williams. "Learning Internal Representations by Error Propagation". David E. Rumelhart, James L. McClelland, and the PDP research group. (editors), Parallel distributed processing: Explorations in the microstructure of cognition, Volume 1: Foundation. MIT Press, 1986.

[27] Sessions, Valerie and Marco G. Valtorta. "The Effects of Data Quality on Machine Learning Algorithms." ICIQ (2006).