National College of Ireland

Classification of Cancer Gene Variants Stages using Ensemble and Deep Learning Approaches

MSc Research Project
Data Analytics

Ramyaa Rajasekar
Student ID: x21122881

School of Computing
National College of Ireland

Supervisor: Qurrat Ul Ain

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Ramyaa Rajasekar
Student ID:	x21122881
Programme:	Data Analytics
Year:	2022 - 2023
Module:	MSc Research Project
Supervisor:	Qurrat Ul Ain
Submission Due Date:	01/02/2023
Project Title:	Classification of Cancer Gene Variants Stages using Ensemble and Deep Learning Approaches
Word Count:	7127
Page Count:	22

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Ramyaa Rajasekar
Date:	1st February 2023

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Classification of Cancer Gene Variants Stages using Ensemble and Deep Learning Approaches

Ramyaa Rajasekar
x21122881

Abstract

The advancement of technology and its integration with healthcare have had a positive impact on the world. Among several diseases, Cancer has had a significant impact on society in recent years, but it is also found that the threats can be reduced by implementing artificial intelligence to help medical professionals make an early diagnosis using cutting-edge technology in classifying the cancer stages on a patient's genetic history based on the clinical evidence to provide individualized treatments in a time-efficient manner. Therefore, to automate the manual process handled by clinical experts in classifying the genetic mutations to a specific cancer class using MSKCC gene data chosen from Kaggle, traditional machine learning models - Logistic Regression, K nearest neighbor (KNN), Random Forest (RF), Support Vector Machine (SVM), Gradient Boosting (GB), Majority Voting Ensemble Classifier, and in deep learning model - Long Short Term Memory(LSTM) was built and the model performances are compared. Different techniques such as Natural Language Processing (NLP), Word2Vec word embedding technique, and hyperparameter tuning were implemented to increase the classification prediction. Finally, the models built are evaluated using metrics such as Accuracy, Recall, F1 score, and Log loss. Based on the evaluation metrics Voting Ensemble Classifier attained better accuracy of 69% with minimal log loss of 0.89.

1 Introduction

Cancer cells in a tumor have the potential to travel to other parts of the body. Considering, the large impact on the number of cancer patients, early detection and diagnosis is a crucial part of patient care and clinical research management. Early detection and prognosis of cancer play an important role in providing cancer treatment. Examining genes is crucial for prevention, therapy, and associated recovery. Therefore, individualized medication is crucial in the treatment of cancer. The two main causes of cancer in humans are genetics and diet. Thousands of genetic mutations are utilized to classify cancer into subtypes (Sruthi et al.; 2022). Most cancer tumors, once sequenced, can have hundreds of genetic alterations. It is challenging for the medical community to determine the mutation and recommend a treatment for patients in a short amount of time because every person's genetic makeup affects their risk of developing a certain type of cancer. The adoption of customized medicine in the treatment of cancer is progressing slowly, though, a significant amount of manual effort to classify the gene data based on unstructured text is still needed.

On the web, there is a huge amount of unstructured medical text that includes valuable information. People find it challenging to digest, read, and remember this text since it is evolving and multiplying. To create novel automation methods to process unstructured text, data mining, and information extraction algorithms are required to be applied. Automatic classification of unstructured text allows useful information management independent of classification's subjective standards (Al-Doulat et al.; 2019) which reduces the human effort involved. Hence, to reduce manual effort implemented by pathologists and oncologists in classifying the genetic variants into different cancer type machine learning plays a vital role in finding patterns in clinical text and categorizing genes into nine different cancer stages.

The motivation for this research is the understanding, that not every person's body reacts the same way to an illness with cancer throughout the pandemic. As tumor growth varies genetically for individuals, the challenge faced by clinical experts is to manually differentiate the genetic variants of each cancer category having the clinical literature as base knowledge which is considered a time-consuming process. To overcome this manual effort, machine learning is implemented to build a classifier model, which automates the classification of genes to cancer classes. Through this time-efficient approach, the diagnostic and recovery rates would eventually rise in favor of the clinical management and cancer patients.

The research question addressed in this research is: *To what extent the cancer gene variants stages can be classified into nine different cancer classes having the clinical literature as a knowledge base using traditional machine learning, Ensemble Learning, and Deep Learning modeling techniques?*

The main objective of the research is to classify the genetic variants into nine different cancer types based on the annotated knowledge base using different traditional machine learning models -Logistic Regression, Support Vector Machine, Random Forest, Gradient Boosting, an ensemble model - Maximum Voting Classifier, and deep learning model – Long Short-Term Memory (LSTM) with hyper-parameter tuning. As a novel approach word embedding technique, Word2Vec is applied for all the features and an ensemble classifier and deep learning model were utilized for model building by passing the best parameters.This research is implemented using open-source gene variants data provided by Memorial Sloan Kettering Cancer Center (MSKCC) available in public repository Kaggle. Also, A comparative analysis of machine learning and deep learning models are evaluated using various evaluation metrics such as confusion matrix, Accuracy, Precision, F1 Score, and Log loss.

The research contributes to clinical pathologists and medical professionals by minimizing the time and effort required to manually classify a patient's cancer stage using clinical text and gene information. An automated machine-learning model is developed to replace this tedious effort. In the end, the medication procedure improves in treating cancer patients since reports can be prepared and given to the doctor more efficiently based on model results.

Following this Introduction section, Section 2 discusses the Related Works proposed by researchers and experts, Section 3 comprises the Methods utilized in this research, Section 4 depicts the Design Specification of this research, Section 5, discusses the Implementation, Section 6 describes the Evaluation and Discussion followed by Conclusion and Future Work in Section 7.

2 Related Works

Numerous biological and machine learning-based scientific studies have focused on classifying cancer gene variants as it performs better than existing technologies in every imaginable way.To perform text classification and build different machine learning models for any genetic data, this section explains the research done by experts in various machine learning and Artificial intelligence induced approaches.The findings, comparison, and novelty of this research project are derived from the knowledge acquired in this section.

2.1 Machine Learning Approaches to Classify Genetic Data

To predict cancer classifications based on data from the genome sequence, several machine learning algorithms have been used by experts. Similarly, in this research (Biswas et al.; 2021) experts utilized a textual dataset (MSKCC) from the public repository Kaggle which comprises gene variants information to classify them into different cancer types having the clinical text as the base knowledge. In the pre-processing stage, the text is cleaned and processed using ntlk libraries such as the removal of stop words, numerical values, and punctuation using regex, one hot, and target encoding implemented for gene and variation features which eventually helps the machine to build the model. The train, test split was done in an 80:20 ratio. A Random model was built using logistic regression, Support Vector Machine (SVM), K nearest Neighbours (KNN), Decision Tree, Random Forest, and Naïve Bayes by performing hyperparameter tuning. The evaluation metrics such as Log loss, Confusion Matrix are calculated. It is proved that implementing One hot encoding as the text processing Random Forest resulted in the least log loss, but logistic regression resulted in attaining 69% of precision and F1 score whereas KNN was the least performed model.

In this research paper, experts performed the classification of genetic mutation to a cancer type based on the clinical literature data whereas in this research (Waykole and Thakare; 2018) one hot encoding was implemented to extract features from the gene and variation features and Term Frequency - Inverse Document Frequency (TF-IDF) was used to extract features from the clinical literature. After implementing the word embedding techniques, the data is converted into vector form then, a logistic regression model was built with cross-validation and attained an accuracy of 64%. It was also stated that different text feature extraction methods, such as word2vec, can be used in further experiments, and other classification algorithms can be used to improve accuracy.

In this research, an Electronic medical record was considered as a source of information to determine and classify, the patient's diagnosis and treatment. As the data chosen in this research (Jamaluddin and Wibawa; 2021) was an unstructured text format, therefore, to extract the features and read the hidden information TF-IDF technique is implemented, and SVM was chosen as a classifier model as it is one of the best models to classify text data. The kernel function and the decision to apply preprocessing techniques are considered while building the model. The outcome shows that the TF-IDF and SVM methods can be utilized to predict diagnosis with stop word removal in an efficient manner. All SVM kernels' classification performance improved when stop words were removed, with accuracy reaching 89.91% for the linear kernel, 90.58% for the polynomial kernel, 90.75% for the RBF kernel, and 91.03% for the sigmoid kernel.

2.2 Deep Learning and Ensemble Approaches to Classify Genetic Data

Convolutional neural networks (CNN) have recently shown that they perform well in a variety of tasks involving natural language processing. In this study (Yoon et al.; 2018) experts aim to perform text classification on clinical pathology reports. Hence, a new word-based CNN model was mentioned as a state of the art method evaluating the convolution filters with the variance of the max pooling outputs in the train set. This method was used to extract information from a very unbalanced dataset of cancer pathology reports and classify the cancer types within it. One of the innovative techniques used was to build a model with approximately a third fewer network weights than traditional CNN training and produced gains in the micro-averaged F1-score of about.07 and the macro-averaged F1-score of about 22. The present focus of the investigation was only word-based Using CNN models, only one network layer's worth of the convolution filters was evaluated. However, further study will be done to create more useful metrics that can assess several convolution filter layers or different layer types.

This research was performed to predict different stages of lung cancer from a textual data by building integrated machine learning approaches. The data was initially analysed for missing values and linear transformation was performed. Later the normalised data is split into 80 and 20 ratios of the Train and Test set. In this study (Reddy et al.; 2019), an Ensemble model comprised of KNN, Neural Networks, and Decision Tree along with Bagging as meta learner was built. The output from individual models is then fed into the integrated models and then the prediction is evaluated. The comparison of model performance is evaluated with and without bagging associated with the models built. It was observed that the bootstrap technique enhances the model performance and accuracy by attaining 0.97 (Decision Tree), 0.94 (KNN), and 0.96 (Neural Networks) and the ensemble models result with 0.98 accuracy.

In this research, the goal was to apply deep learning approaches to perform the classification of the medical text data which is in an unstructured format using different linguistic techniques. In this study (Al-Doulat et al.; 2019), multi-class classification is performed rather than binary classification. Different NLP techniques were used during the pre-processing steps such as Sanitisation, sentence, and word tokenization, stop words and punctuation removal and word lemmatization. Later, in the data cleaning step, feature extraction was performed using content-based and domain-specific approaches. To understand the better correlation between the text features and the target class Recursive feature elimination technique was used along with cross-validation. Finally, the deep neural network model is built with 2,3,4,5, and layers with a hidden layer including 50 to 100 neurons and it has been evaluated using metrices such as accuracy, precision, and recall. It was noticed that the technique implemented in this study resulted in better accuracy of 82% compared to the baseline model attained by TF-IDF (62%) approach.

This study focused on the classification of medical publications primarily on prevalent cancer kinds. Because most cancer kinds have a similar literature content, experts in this research (Kolukisa et al.; 2021) purposefully focused on MEDLINE papers about common cancer types. As a result, this circumstance makes the classification somewhat more difficult. To perform this classification, numerous machine learning models use both conventional and deep learning architectures. The LSTM model attained the best performance at 82% F1 score. The textual data pre-processing was performed using Natural Language Processing Techniques (NLP). TF-IDF was utilized to understand

the correlation of each feature with the entire document and get the frequency counts following this approach Logistic regression and a Dense neural network model was built. Secondly, using a word embedding techniques Unigram, CNN, Recurrent Neural Network (RNN), and LSTM models are built, and the models are evaluated in a comparative manner. Overall, the findings in this study show a significant benefit of using both text mining and machine learning techniques to separate medical literature on prevalent cancer kinds and resulted in LSTM attaining the best accuracy. On further analysis, Word2vec could have increased the classification accuracy compared to TF-IDF.

2.3 Medical Text Data Preprocessing techniques

In this study (Sadman et al.; 2020), an NLP based framework is built and different machine learning and deep learning algorithms have been built to compare the categorization performance of medical records. Initially, the framework model is split into training and testing phases. In the training phase, the train set is fed as input, and data cleaning is performed using different NLP techniques such as stop words, lemmatization, and tokenization. Later, feature engineering is implemented using Principal Component Analysis (PCA) and a Bag of Words (BOW). Finally, in the model-building stage, 7 different NLP classification algorithms model is built to train the model. Experts stated the dataset is split into 90-10 ratios when fed into the training phase by default. The evaluation metrics utilized are the confusion matrix and F1 scores for the model performance. The main advantage of this research is the NLP framework built can be utilized for different datasets. The accuracy was observed to be resulting around 92% which is considered the best-performed ensemble model.

Four medical text datasets, including two datasets of medical records and two datasets of medical literature, are the subject of experimental verifications chosen by experts in this study (Qing et al.; 2019). It is stated that the classification of the medical text deals with complicated medical vocabularies and languages. Hence, to overcome this difficulty, a novel approach hierarchical attention neural network was introduced which comprises document and sentence representation sections. In sentence representation word embedding techniques such as bag of words and sentence, the decoder was utilized.

Whereas in a similar study, experts proposed CNN, Bidirectional LSTM,Multi-Head Attention a unique deep learning model for classifying medical literature. A comparison was performed in this study (Shen and Zhang; 2020) on different word embedding techniques such as TF-IDF implemented dataset is built with machine learning models such as NB, Decision Tree, Random Forest, KNN and SNLP-based as word2vec technique is implemented to build deep learning models such as CNN and LSTM. With an accuracy of 89.79% and an F1-score of 89.85%, the SVM based on TF-IDF performs best. The performance of deep learning-based methods is superior to machine learning-based methods. The suggested strategy in this research performs better than other methods, with an accuracy of 91.99% and an F1-score of 92.03%.

2.4 Literature Review Highlights and Summary

From the knowledge acquired from the literature review, it is observed that unstructured text classification is a complex process therefore different techniques such as Natural Language Processing(NLP), and Word embedding techniques play a major role in building a machine learning model. The techniques utilized by researchers are to implement one

hot encoding, tokenizer, and TF-IDF to convert the text into vectors or sequences. Also, there are many other techniques such as Word2Vec or Glove pre-trained model that can be utilized which would give better classification results. When comes to model building, it was proved ensemble classifiers and deep learning model gives satisfactory results. From the insights acquired, As a novel approach in this research word2vec embedding technique was implemented to the data chosen and an ensemble classifier and deep learning model were built and compared to classify the gene variants.

Author	Dataset Used	Algorithm and Techniques	Results
(Biswas et al.; 2021)	MSKCC Genetic data	LR,KNN,NB,RF,DT, One hot encoding	Logistic Regression - 69%
(Waykole and Thakare; 2018)	MSKCC Genetic data	Logistic Regression, One hot encoding and TF-IDF	Accuracy 64%
(Jamaluddin and Wibawa; 2021)	Medical Record	SVM and TF-IDF	Accuracy-91%
(Yoon et al.; 2018)	Pathology Report	CNN and NLP	MSE-0.15 and R Score-0.9864%
(Reddy et al.; 2019)	Lung Cancer Text data	Ensemble Approach Bagging KNN, DT, Neural Network	DT-0.97,KNN-0.94, NN - 0.96, Ensemble - 0.98
(Kolukisa et al.; 2021)	Medical Publications data	LSTM, CNN, NLP, TF-IDF, Unigram	LSTM F1 score 82%
(Shen and Zhang; 2020)	Medical Text Data data	RF, KNN, SVM, TF-IDF, Word2Vec	SVM(TF-IDF) Accuracy 91.99%

Table 1: Summary of Different Approaches from Literature Review

3 Methodology

In this research, the methodology followed is Knowledge Discovery in Database (KDD), which comprise a set of predetermined steps to process the data before applying various data mining algorithms, and acquire insights from the data, analyze the patterns, then provide a meaningful result. It comprises various steps as shown in figure 1. As an initial stage, In Data Acquisition - the dataset selection details are discussed, followed by data pre-processing, feature extraction, and transformation phases - the dataset chosen was in a textual format, different NLP and word embedding techniques implemented are discussed and In the Model and Evaluation stages – different machine learning and deep learning models utilized on the classification of genes variants to different cancer categories are discussed. Later, Knowledge and results are acquired by comparing and evaluating the models using metrics.

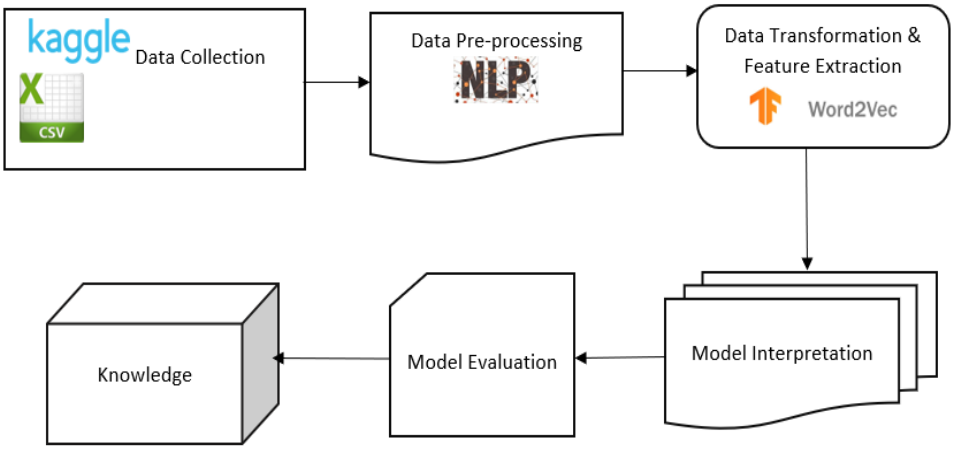

Figure 1: KDD Methodology

3.1 Data Acquisition

Open-source datasets related to cancer gene variants are widely available and can be utilized for the classification and detection of cancer categories to relevant gene variants. Similarly, a public dataset which is available in Kaggle[1] provided by MSKCC for the research to classify gene variants into nine different cancer categories by referring clinical literature as a knowledge base. Different sets of files are available for training (with rows

	ID	Gene	Variation	Class	TEXT
0	0	FAM58A	Truncating Mutations	1	Cyclin-dependent kinases (CDKs) regulate a var...
1	1	CBL	W802*	2	Abstract Background Non-small cell lung canc...
2	2	CBL	Q249E	2	Abstract Background Non-small cell lung canc...
3	3	CBL	N454D	3	Recent evidence has demonstrated that acquired...
4	4	CBL	L399V	4	Oncogenic mutations in the monomeric Casitas B...

Figure 2: Overview Of Dataset

3322) and testing (with rows 5669) sets. The files are categorized into gene variants (comprises of information about genetic mutations and variations) and text (the text file comprises clinical information from which experts used to classify the genes manually) files having the ID column as the primary key for both the files. As shown in figure 2, the variants file comprises Gene, Variants, and Class columns which depict the gene in which the mutation is located and the variations represent the corresponding amino acids for the mutation whereas the text file comprises Text column which has the clinical text information of cancer affected genes which helps to classify the genetic mutations. The class column has categorical data ranges from class 1 to class 9 available only in the training set and the classes denote generalized cancer categories such as Carcinoma,

[1]https://www.kaggle.com/competitions/msk-redefining-cancer-treatment/data

Leukemia, lymphoma, Mixed Types, and so on in which the genetic variants are classified whereas in the testing set the trained model will be used to perform the prediction and generate the probabilities of class for that gene variants.

3.2 Exploratory Data Analysis

In this stage, the data were analyzed using different visualization techniques to attain insights and meaningful information about the gene variants and clinical text correlations. Visualization was done to understand the distribution of text for every class, to analyze the gene column data distribution, and the class distribution, and to analyze the highest number of gene occurrences for each class using python visualization libraries such as pyplot, patches, markers, and seaborn. The number of unique values in the gene, variants, and class columns are also analyzed.

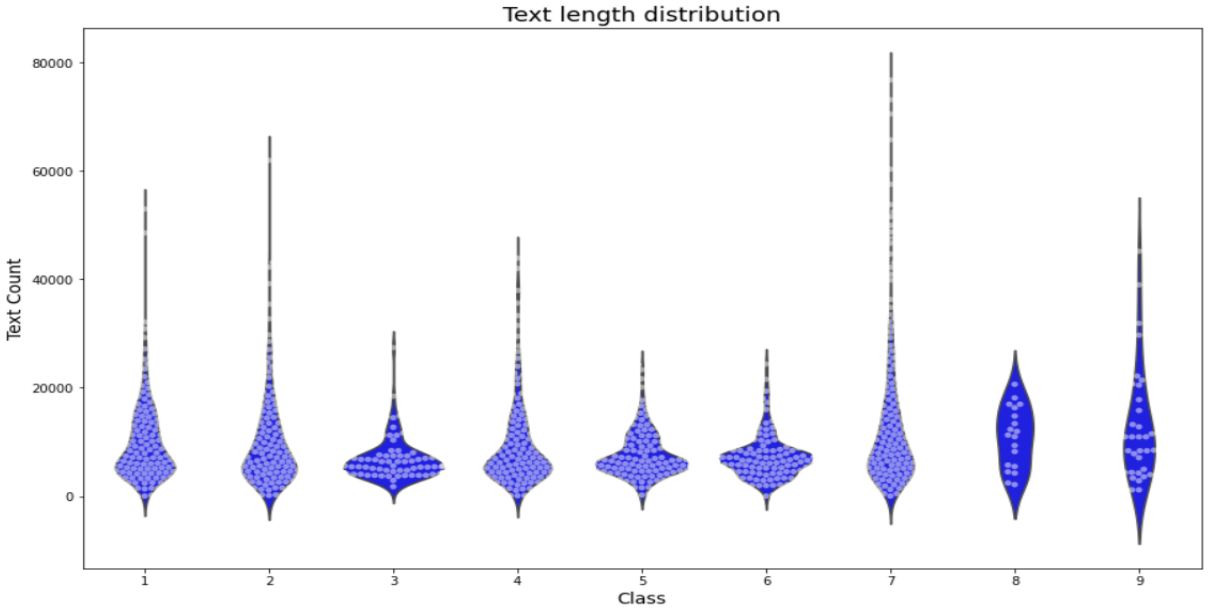

Figure 3: Text Length Distribution

To understand the text distribution, text count was calculated for the number of words available in the text column, and it's evident from the violin plot as in the figure 3 that the text distribution is more for class 7 as it ranges from 0 to 80000, the rest of the classes have the distribution from 0 to 20000.

Genes with the greatest number of instances in each class are visualized in figure 4 from which it is understood that the gene BRCA1 is dominating in the class 5 category, SF3B1 is in class 9, PTEN is dominating in class 4 and BRCA1 and BRCA2 dominating in class 6.

Class distribution was analyzed by plotting the class column using a bar plot as shown in figure 5, from which it is observed that there is not much data available for class 8 and class 9 whereas class 7 comprises more samples.

3.3 Data Pre-processing

The train set variants and text files are of different CSV files, the variants and text files are merged using the ID column as the primary key. There were missing values found in

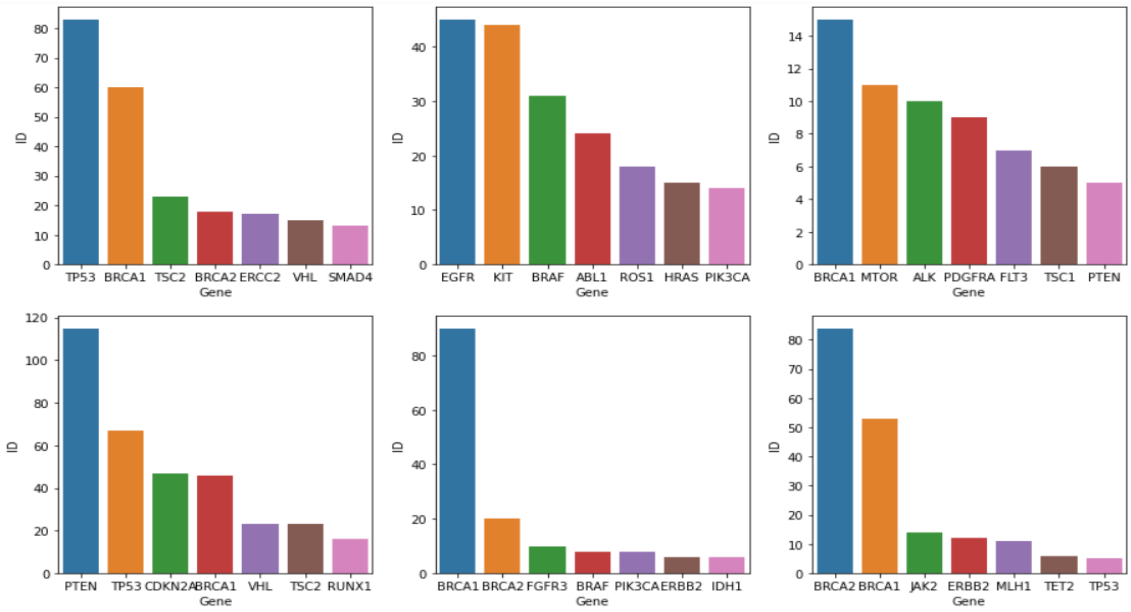

Figure 4: Gene occurrences for each class

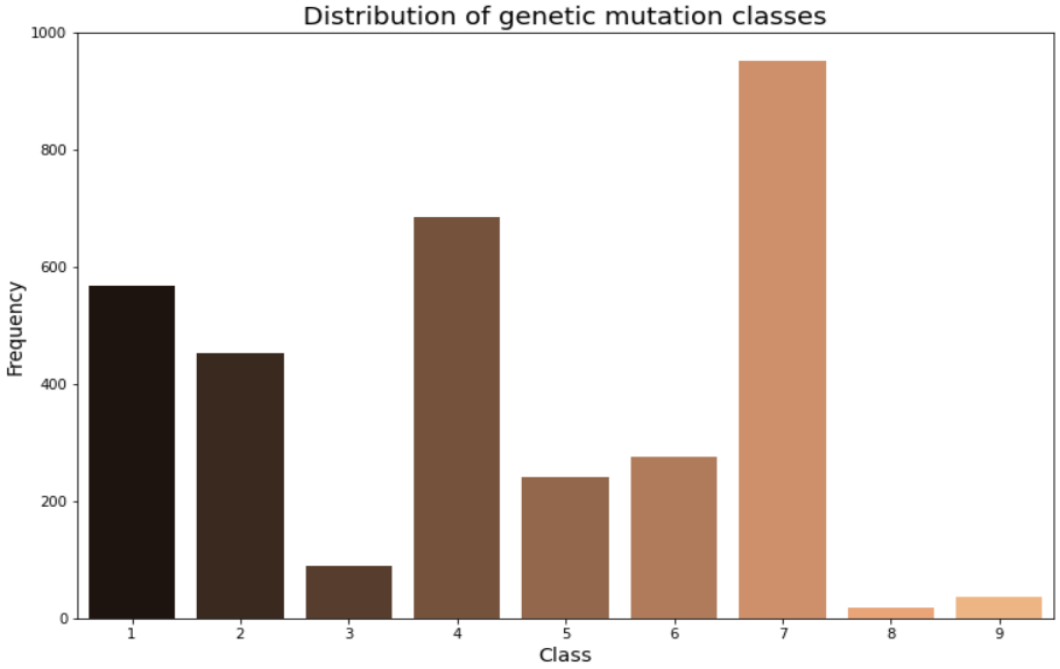

Figure 5: Class distribution

the text file which were replaced with the corresponding Gene and variation data. The data chosen for this research is of textual format, to build a machine learning model the text data should be converted into a machine-understandable format. Natural Language Processing (NLP) techniques were utilized for the removal of stop words, numeric values, punctuations, and special characters also the characters are converted to lowercase to maintain consistency by importing integrated ntlk libraries in python such as regex, stopwords, and lemmatize.

3.4 Data Transformation and Feature Extraction

The data chosen are in continuous text format, hence, it is essential to understand the frequency of the words and convert the text to numerical or vector format to enable machine learning models to fully comprehend the data and perform accurate classification.

Therefore, in this research to understand the word frequency and to convert the word into the form of vectors Word2Vec word embedding technique google's pre-trained model has been implemented using the genism library in python. By using word embedding, it is possible to convert strings into sequences of vectors that may then be used as training data for a model that predicts the future. Words with similar traits can be grouped together using word embeddings, whereas words with different traits can be dispersed widely apart in the vector space. It also creates semantics that is helpful for text-based classification.

Initially, a unique token was assigned to each sentence. The tokenized sentences were generated using a class named mysentences, where each distinct token represents that entire sentence. To train a word2vec model 100 was used as the embedding size by default, and each word was represented by a numeric vector with 100 dimensions, min_count as 1 as the words which occur at least once will be taken into consideration and workers as 4(which depicts the number of threads used to reduce the training time). The average or mean of all the numerical vectors were then calculated to produce a single vector for each entry. For example, the embedding achieved for the word 'mutation' is 0.8196579.

For deep learning model LSTM, the input data was converted to a tensor format data by transforming the text into sequences using a tokenizer as it breaks the sentence into a list of words or strings using a function texts_to_sequences to pass it as a sequence of words. Later, the sequenced data is padded to ensure the length of the input data is all of the same dimension using pad_sequence.

3.4.1 Word Embedding Exploration

Additionally, t-Distributed Stochastic Neighbor Embedding (t-SNE) was used for data exploration of high-dimensional data, to understand the cosine similarity of the vectors. For the t-SNE, a sample of 100 words was considered from the overall vocab size and it is reduced to 2 dimensional and converted to vectors to understand the similarity among the words plotted in the vector space. Principal Component Analysis (PCA) and K mean clustering were used to visualize the correlation of embedding with the classes to understand how the individual values of variables are placed across each cluster and how it resembles. But K means clustering gave better results, It was visualized for the text column against the corresponding class, The clusters determined are 9 and from the figure 6 it is observed that the text distribution against the class appears to be distinct.

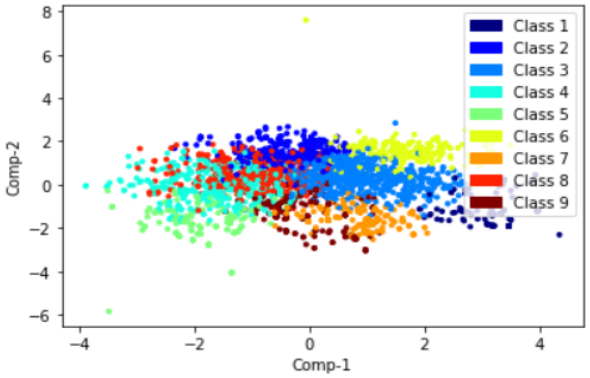

Figure 6: K means Clustering for Text and Class Correlation

3.5 Modelling

The main aim of the research is to classify the gene variants into different cancer classes, to perform this classification, different machine learning and deep learning classification models were built. Initially, Logistic Regression, K nearest neighbor, Random Forest, Support Vector Machine (SVM), and Gradient Boosting classification models were built along with hyperparameter tuning. Secondly, an ensemble classification model with Random Forest, Gradient Boosting classifier, and Support Vector Machine (SVM) with Majority voting classifier that works as a meta classifier was built and a deep learning model was implemented using Long Short-term Memory (LSTM) and a comparative analysis of the different models built was performed.

3.5.1 Ensemble Classifier

An ensemble classifier model was built by combining the predictions of Random Forest, Gradient Boosting classifier, Support Vector Machine (SVM), and Majority Voting work as a meta classifier. As shown in figure 7, it combines the predictions from each classifier model and calculates an average predictions accuracy of class predicted calculates voting of the number of correctly predicted classes and provides final predictions.

Support Vector Machine (SVM),(Hameetha Begum and Nisha Rani; 2021) is one of the supervised machine learning algorithms which plots a hyperplane through which the data is divided into two sets, the points closest to the plane is considered the support vectors. When it comes to hyperparameter tuning, different parameters such as kernels - Linear, rbf, sigmoid and, C = 3,5,20 can be passed and the best parameters against the training set are found and the model is built.

Random Forest is one of the effective solutions for the classification and regression problem. As part of the training, multiple decision trees are built and combined as part of the model building to produce accurate predictions. Different parameter values such as max depth, min sample leaf, min sample split, and criterion are passed to perform hyperparameter tuning.

Gradient Boosting is a supervised learning one of the tree-based algorithms similar to a decision tree classifier which can be used for continuous and categorical target variables. It basically aggregates the predictions of each decision tree and helps to calculate the final results. Also, boosting plays an important role in boosting the accuracy or prediction by supporting the meta-learner.

Majority Voting Classifier works as a meta classifier that gets the input from multiple

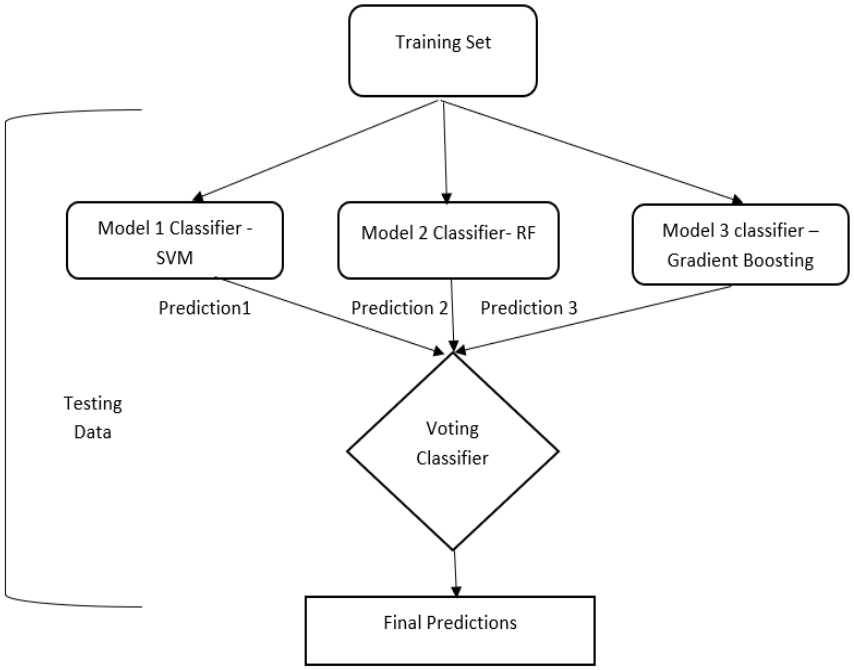

Figure 7: Ensemble - Voting Classifier

classifiers in this research the classifiers utilized are SVM, Random Forest, and Gradient boosting. The predictions from individual classifiers are taken into consideration and it votes for a specific class. The highly voted class is considered the final prediction and the accuracy is calculated.

3.5.2 Long Short Term Memory (LSTM)

LSTM is one of the effective model for sequential data and is meant as a part of the Recurrent Neural Network (RNN) model. In this research, a four-layered sequential model is implemented, in the embedding layer one vector of each word is stored. When invoked, it transforms word index sequences into vector sequences. Words with comparable meanings generally have similar vectors after training(Zhang; 2021). This layer comprises the input data information such as maximum number of words to be taken, and word embedding dimensions, second layer is a spatial dropout1D layer with value of 0.2,then the LSTM layer with 100 units with recurrent dropout as 0.2 to avoid over fitting. Finally, the last layer is the dense layer with 9 units.As the research is of multi class classification, the activation value was passed as "Softmax" as it converts the output layers in a probability predictions and loss as "Categorical Crossentropy", epoch as 10 and batch size as 64. Figure 8 shows the architecture of the LSTM [2] for text classification.

3.6 Evaluation

To perform the comparative analysis of the classification models built, various evaluation metrics were utilized such as Accuracy, Precision, Recall, F1 Score, Log loss, and a plot

[2]https://colah.github.io/posts/2015-08-Understanding-LSTMs/

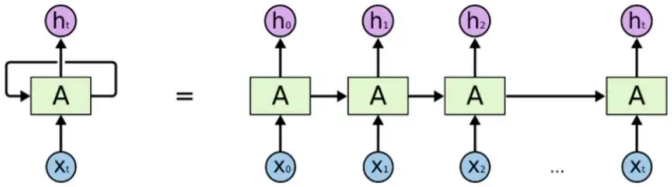

Figure 8: LSTM - Multi-class architecture

of confusion matrix to understand the probability of the prediction of classes with the actual or true cancer classes. According to the research article (Hameetha Begum and Nisha Rani; 2021), the metrics frequently employed for categorization prediction are listed below:

Confusion Matrix:

As the research is based on multi-class classification, a classification matrix is built for the nine cancer classes where N = 9 (9x9 matrix), and the probability prediction of the cancer classes to the actual or true values is depicted. In other words, the true positives, true negatives, false positives, and false negatives are plotted to understand the prediction through visualization.

Accuracy is evaluated by calculating the number of samples correctly predicted. It depicts how accurately the prediction is performed with respect to the actual values. The percentage attained shows the model performance from which the classification can be evaluated.

$$Accuracy = \frac{TruePositiveclasses + TrueNegativeclasses}{TotalPredictionsclasses} \tag{1}$$

Precision is calculated as the ratio of correctly predicted positive classes to the total and falsely predicted positive classes.

$$Precision = \frac{TruePositiveclasses}{TruePositiveclasses + FalsePositiveclasses} \tag{2}$$

Recall is calculated as the ratio of correctly predicted positive classes to all observations in actual class. As there are two different ways to calculate, macro averaged recall is taken into consideration, to calculate recall for individual classes and average the same.

F1 score is an evaluation metric which is combination of precision and recall. Therefore, both false positives and negatives are considered while calculating this score. Although F1 is generally more beneficial than accuracy, especially if there is an uneven class distribution.

$$F1Score = \frac{2PrecisionXRecall}{Precision + Recall} \tag{3}$$

Log loss is considered as an ideal evaluation metrics when it comes to classification modelling analysis. The log loss will be better when closer to zero when the prediction probability is good and vice versa. A low Log Loss indicates the model's uncertainty or entropy is low as it is a measure of both.

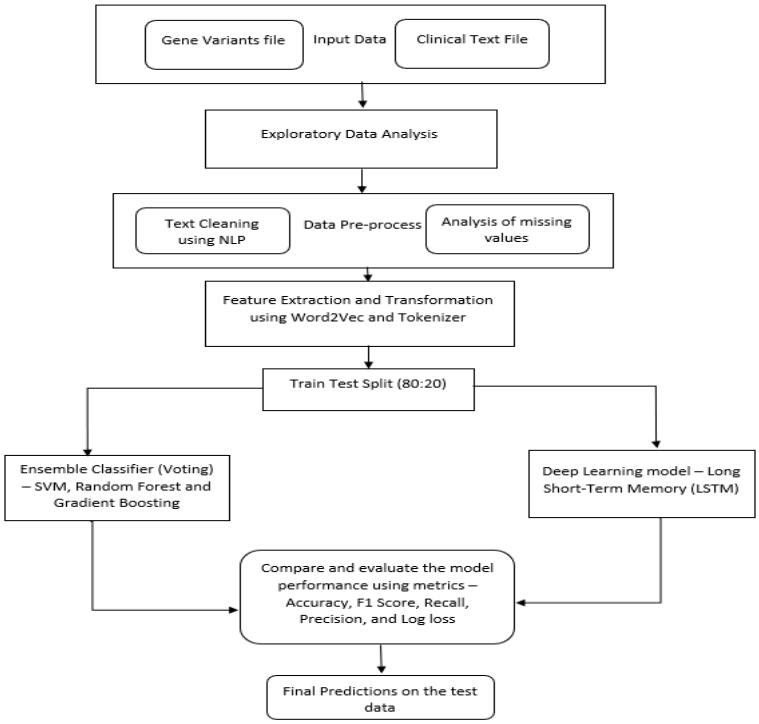

Figure 9: Design Flow Process

4 Design Specification

From figure 9, the different process stages of the research are depicted in a sequential manner. Initially, the input data is loaded, and exploratory data analysis is performed with the raw data. Then, the textual preprocessing techniques using NLP and missing values were handled. As part of the feature transformation, Word embedding techniques are utilized and the data is split into 80 and 20 ratio. Finally, a traditional ensemble classifier is built and a deep learning model LSTM is built. The model performance is evaluated using different classification metrics.

5 Implementation

The dataset chosen for this research was taken from the public repository Kaggle. It was provided in different files for the train and test set. Initially, the training data was taken, and the train test split ratio was done for 80:20 ratio, the model building and evaluation were done on the data split and the prediction from the best-performed model was applied to the test dataset provided.

The entire model-building process and implementation are performed on Jupyter notebook hosted by Google Colaboratory, which enables the coding and execution of machine learning and deep learning models through the browser.

The programming language used to implement the classification approach was Python, as it is free and open source and has various integrated libraries which were utilized for implementing NLP techniques, Word Embedding techniques, and Tenser flow Keras versions for deep learning models.

As part of textual pre-processing, the data was cleaned by removing special characters,

stop words, and punctuations using NLP techniques by importing ntlk libraries.

As a novel approach, the text was transformed into vectors for Gene, Variation, and Text features using google's pretrained Word2vec model by importing the genism library,To transform a set of sentences into its appropriate vector, a transformer was defined using the Sklearn interface. and the visualization of the vectors is performed to understand the words plotted on the vector space using the Principal Component Analysis (PCA), K means clustering, and t-SNE.

Different machine learning and deep learning models are built and they can be found in the Sklearn Python library utilized in different packages.

5.1 Logistic Regression With and Without hyperparameter tuning

Initially, a model was built by implementing a logistic regression algorithm with the default parameters, and by feeding the best parameters C = 1.0, solver = 'newton-cg', penalty = 'l2', max_iter=100. After the training, the test set is predicted and evaluated using classification evaluation metrics.

5.2 KNN With and Without hyper parameter tuning

Secondly, the KNN model was built with the default parameters and by finding out the best parameters the model was again trained with the train split, from which the best parameters found are algorithm = 'auto', n_neighbors= 3 using GridSearchCV with 10 times cross-validated. After the training, the test set is predicted and evaluated using classification evaluation metrics.

5.3 SVM With and Without hyperparameter tuning

SVM model was built with the default parameters and by finding out the best parameters the model is again trained with the train split, from which the best parameters found kernel = 'rbf', C = 20, probability = True using RandomizedSearchCV. After the training, the test set is predicted and evaluated using classification evaluation metrics.

5.4 Random Forest

Random Forest model was built with the default parameters and by finding out the best parameters the model is again trained with the train split, using RandomizedSearchCV from which the best parameters found bootstrap = True, criterion= 'entropy', max_depth= None, max_features= 5, min_samples_leaf=4, min_samples_split=6. After the training, the test set is predicted and evaluated using classification evaluation metrics.

5.5 Ensemble Learning Classifier – Majority Voting Classifier

As a novel approach, an ensemble model is built by combining the predictions of Gradient Boosting, Random Forest, and SVM with a hyperparameter. The Voting classifier acted as a meta learner by passing the parameter vote = "Soft", which takes the predictions of different classifier models as input and calculates the majority voting for the classes

predicted across all classifiers. The final predictions are made by calculating the majority of classes predicted. This model can be imported from sklearn.ensemble python package.

5.6 LSTM model

To build an LSTM model, the four-layered sequential model is implemented, in the embedding layer one vector of each word is stored. This layer comprises the input data information such as the maximum number of words to be taken, and word embedding dimensions, second layer is a spatial dropout1D layer with a value of 0.2, then the LSTM layer with 100 units with recurrent dropout as 0.2 to avoid overfitting. Finally, the last layer is the dense layer with 9 units. As the research is of multi-class classification, the activation value was passed as "Softmax" and loss as "Categorical Crossentropy", epoch as 10, and batch size as 64. The packages utilized to perform this model building are Keras.preprocessing and keras.layers from which the tokenizer, pad sequences, and different layers such as embedding, LSTM, Dense and Spatial Dropout1D are imported.

6 Evaluation

The results attained after implementing nine different models are evaluated by comparing the Accuracy, Precision, Recall, F1 score, and logloss values.

6.1 Logistic Regression evaluation

From the results, it was observed that the logistic regression model before and after passing the best parameters attained the same accuracy of 0.61, precision of 0.59, F1 score of 0.58, and log loss of 1.11. The results attained were not very interesting when compared to the results acquired by experts in previous research using this algorithm.

Models	Accuracy	Precision	Recall	F1 Score	Log loss
Logistic Regression	0.61	0.59	0.61	0.58	1.11
KNN	0.57	0.58	0.57	0.58	4.49
KNN with Hyp	0.59	0.60	0.58	0.59	6.69
SVM	0.60	0.58	0.60	0.55	1.08
SVM with Hyp	0.64	0.65	0.64	0.63	1.01
Random Forest	0.69	0.69	0.69	0.68	1.07
Gradient Boosting	0.67	0.66	0.67	0.66	0.96
Voting Classifier	0.69	0.69	0.69	0.68	0.89

Table 2: Result Comparison of Different Machine Learning Approaches

Models	Training Accuracy	Validation Accuracy	Training Loss	Validation loss
LSTM	0.78	0.54	0.63	1.36

Table 3: LSTM Results Summary

6.2 KNN Evaluation

For the KNN model, the results observed were a little better than the previous model logistic regression where the accuracy attained was 0.57 after passing the best parameters the accuracy got increased by 2%, but the log loss acquired seems to be very high 6.69 from which it can be concluded this model can be rejected as the loss appears to be far from 0. Though the k value is changed from 1 to 3, it is observed that the change in accuracy and precision is 2% and other output parameters such as Recall and F1 score is 1%, and log loss from 4.49 to 6.69 which is approximately 2%.

6.3 Support Vector Machine Evaluation

Support vector Machine, with default parameters the accuracy attained, was 0.60, which was similar to the previous models but when built by feeding the best parameters it is evident the results came out well with an Accuracy is 0.64, Precision of 0.64, the F1 score of 0.63 and log loss resulted to be low 1.01 compared to the above models.

6.4 Random Forest Evaluation

Random forest classifier before and after passing the best parameters attained the same accuracy with accuracy and precision of 0.69, F1 score of 0.68, and loss of 1.07. When compared to other models, Random Forest attained better accuracy.

6.5 Gradient Boosting

Gradient Boosting classifier attained results better than SVM with an accuracy of 0.67, and one of the least losses of 0.96. When compared to other models, GB attained the second least log loss.

6.6 Ensemble Voting Classifier Evaluation

Majority Voting ensemble model - SVM,RF, and GB with the best parameters observed as a good classifier with accuracy and precision of 0.69 and F1 score of 0.68 also the log loss was reduced to 0.89 which is better than Random forest and other models built.

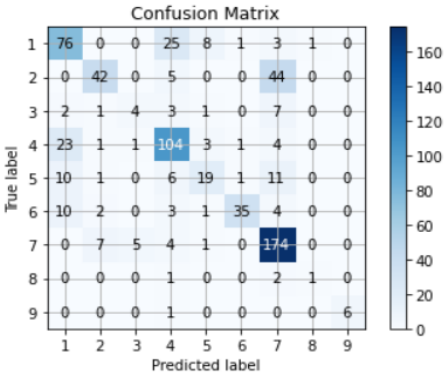

Figure 10: Ensemble Classifier- Confusion Matrix

6.7 LSTM Evaluation

In order to perform a comparative analysis with machine and deep learning model, LSTM model was built as it's an effective text classification model.For epoch 10 and batch size 64, the training accuracy attained was 0.78 with a loss of 0.63 but the validation accuracy of 0.54 and log loss of 1.36 appears to be low compared to the training. From this model, it is observed that the training evaluation is observed to be good but the validation accuracy is less compared to the previously built models.

From the graph 11, it is observed there is a huge difference in the results between the training and the validation set. For each epoch, there is variation in the accuracy and log loss.For the training set the log loss is reduced and accuracy is increased after the epoch value 4.

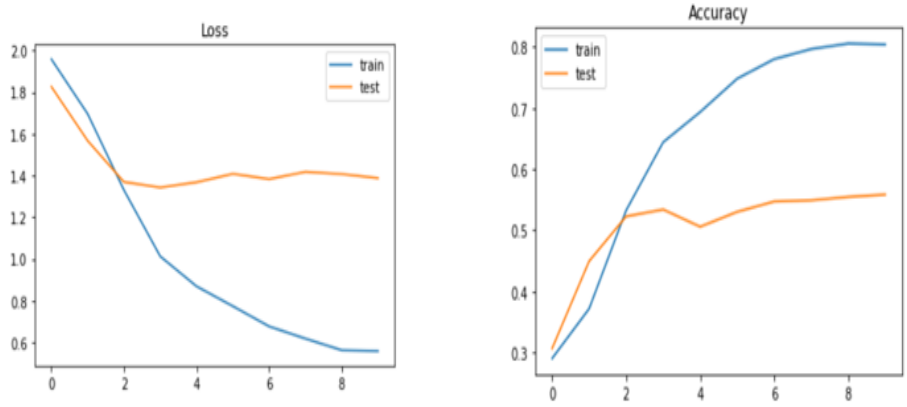

Figure 11: LSTM - Accuracy and Log loss

The limitations observed in this classification process were, the data set size chosen for this research was considered to be small and to overcome the imbalance issue, more samples can be added as per clinical experts' advice to improve the classification accuracy and to analyze if there is any influence in the prediction results. However, the LSTM model didn't favor well in the results, hybrid deep learning models can be implemented. Additionally, this research is limited to the word2Vec embedding technique, where it is observed that different embedding techniques such as Doc2Vec can also be implemented.

6.8 Comparison of Random Forest, Ensemble Classifier and LSTM

By comparing the performance of the models, the Voting Classifier and Random forest gave similar accuracy, but when the log loss was compared the voting classifier attained a minimum log loss of 0.89. As Random Forest is well-known as an ensemble technique where it combines multiple decision trees, the framework of Random Forest and Voting classifier is quite similar as both take the average voting for prediction. When the predictions of the Random Forest and Ensemble classifier were compared, it was observed from the confusion matrix that the predicted classes were nearly similar whereas the number of misclassified classes was more in Random Forest. The results from the LSTM model

were not so promising when compared to the previous research, the validation accuracy attained is the least, and this could be increased by implementing hybrid models with different transformation techniques.

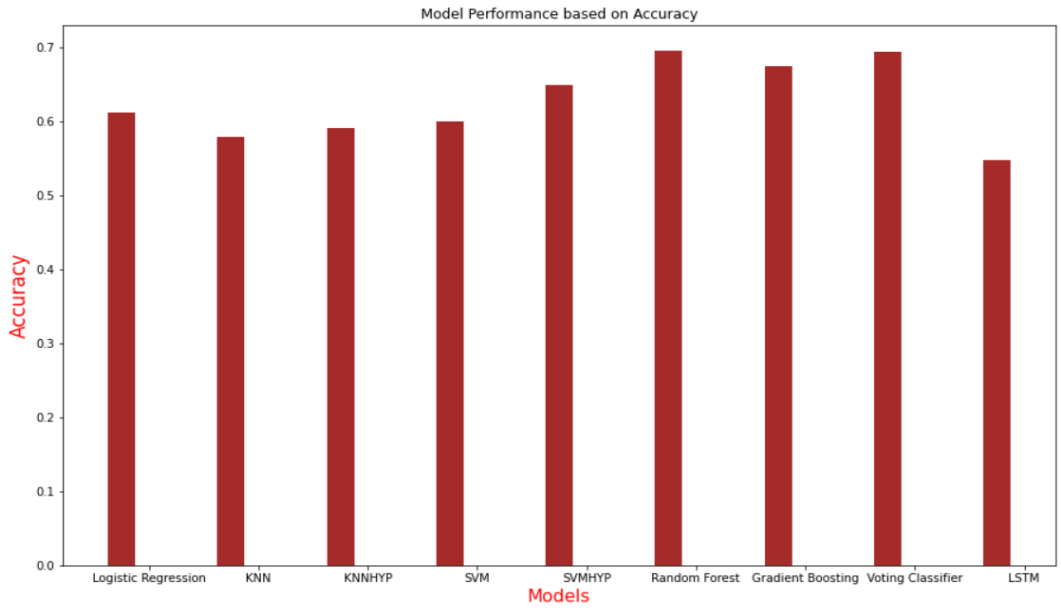

Figure 12: Model Comparison Based on Accuracy

The final analysis was based on the evaluation performed on the test or validation set. By comparing the models as shown in figure 12, accuracy, precision, F1 score, and log loss the voting classifier performed well and gave minimal log loss of 0.89 and a true predicted class of 69, and KNN was considered as the least performed model with the highest log loss of 6.69.

6.9 Discussion

As the main goal of this research is to have the clinical literature as a knowledge base and classify the gene variants into different cancer classes. Models are designed to assign a probability to a particular cancer class. This was achieved from the predictions performed by the best model - Ensemble Majority Voting classifier, it is evident that the majority of the correctly classified (174) genes EGFR, ROS1, ERBB2, and ERBB3 are correctly predicted as class 7 category which is related to carcinoma cancer categories and this mutation primarily affects the lung and organs such as stomach, bladder, and ovaries. Secondly, the genes BRCA1, BRCA2, TP53, and NF3 are the Breast Cancer mutations accurately classified as Class 1 (76), and genes TSC1 and TSC2 mutation of Kidney cancer classified to Class 4 (104) are observed to have a moderate prediction. Apparently, it appears that the incorrect predictions for the genes BCOR and IDH2 associated with Class 8 and genes SF3B1 and IDH1 associated with Class 9 belong to the categories of leukemia and lymphoma cancer types. Class 3, 8, and class 9 forecasts were incorrect because there were insufficient samples available to train the model also the highest predicted class is 7, different sampling approaches could have supported and improved these predictions.

The challenges faced in this classification process was, the whole data is in a textual format, to convert the text into vectors Word2Vec vector model was applied for all fea-

tures, the words had less similarity and to obtain the single fixed vector to the mean of the embedded words are taken. To build a deep learning model, the issues faced were the dataset input dimension and format, Hence, to build an LSTM model the words are converted to sequences and padded to a specific length to have a consistent length across the input layer.

By comparing with the prior study (Biswas et al.; 2021), for the MSKCC dataset in this research KNN and Logistic Regression model gave the higher log loss and least accuracy compared to the previous study it attained the least log loss and better accuracy. An improvement in the research was to implement a deep learning model it is found that the training accuracy attained by the model is 0.77 but the validation accuracy was low. Hence, the predictions made by these models didn't meet the expectations and need improvement by implementing different feature extraction and transformation techniques.

Overall, in this research, it is observed that the K nearest Neighbour is not considered a well-performing model due to the highest log loss. Whereas the well-performed model which classified a greater number of classes to the actual values with minimal loss is the majority voting classifier.

7 Conclusion and Future Work

The main objective of this research is to classify the genetic mutations into corresponding cancer classes which are of 9 categories. This classification was performed by clinical experts manually by referring to the clinical evidence as a reference to overcome this manual effort it was suggested that the machine learning model can be built, trained to understand the clinical information, and based on the knowledge acquired the gene and variations can be classified to relevant cancer classes. In assistance to this, in this research the dataset was taken from the public repository Kaggle, several pre-processing and word embedding techniques were performed and various machine learning models were built from which Voting Ensemble Classifier considered the best model to classify the gene variants.

When compared to the research performed by other experts in this context, the Word2vec embedding technique was implemented in this research as data transformation gave better accuracy for the ensemble classifier and Random Forest with minimal log loss of 0.89 and 1.07 and accuracy of 0.69.

As part of future work, as the data is of textual format different word embedding techniques like paragraph2vec or Glove method can be utilized for better classification rather than TF-IDF or Word2Vec. Also, in this research LSTM didn't perform well, different hybrid models such as Convolutional Neural Networks with LSTM can also be implemented. The class distribution was observed to be imbalanced as only a few samples are available for classes 8 and 9 for which class balancing can be applied and model building and evaluation can be performed. Additionally, it is advised to collect a more number of samples in addition to the existing set to attain better and avoid biased results.

References

Al-Doulat, A., Obaidat, I. and Lee, M. (2019). Unstructured medical text classification using linguistic analysis: A supervised deep learning approach, *2019 IEEE/ACS 16th*

International Conference on Computer Systems and Applications (AICCSA), pp. 1–7.

Biswas, S., Kumar, V. and Das, S. (2021). Multiclass classification models for personalized medicine prediction based on patients genetic variants, *2021 IEEE International Conference on Technology, Research, and Innovation for Betterment of Society (TRIBES)*, pp. 1–6.

Hameetha Begum, S. and Nisha Rani, S. N. (2021). Model evaluation of various supervised machine learning algorithm for heart disease prediction, *2021 International Conference on Software Engineering Computer Systems and 4th International Conference on Computational Science and Information Management (ICSECS-ICOCSIM)*, pp. 119–123.

Jamaluddin, M. and Wibawa, A. D. (2021). Patient diagnosis classification based on electronic medical record using text mining and support vector machine, *2021 International Seminar on Application for Technology of Information and Communication (iSemantic)*, pp. 243–248.

Kolukisa, B., Dedeturk, B. K., Dedeturk, B. A., Gulsen, A. and Bakal, G. (2021). A comparative analysis on medical article classification using text mining machine learning algorithms, *2021 6th International Conference on Computer Science and Engineering (UBMK)*, pp. 360–365.

Qing, L., Linhong, W. and Xuehai, D. (2019). A novel neural network-based method for medical text classification, *Future Internet* **11**.
URL: *https://www.mdpi.com/1999-5903/11/12/255*

Reddy, D., Hemanth Kumar, E. N., Reddy, D. and P, M. (2019). Integrated machine learning model for prediction of lung cancer stages from textual data using ensemble method, *2019 1st International Conference on Advances in Information Technology (ICAIT)*, pp. 353–357.

Sadman, N., Tasneem, S., Haque, A., Islam, M. M., Ahsan, M. M. and Gupta, K. D. (2020). "can nlp techniques be utilized as a reliable tool for medical science?" - building a nlp framework to classify medical reports, *2020 11th IEEE Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*, pp. 0159–0166.

Shen, Z. and Zhang, S. (2020). A novel deep-learning-based model for medical text classification, *Proceedings of the 2020 9th International Conference on Computing and Pattern Recognition* .

Sruthi, G., Ram, C. L., Sai, M. K., Singh, B. P., Majhotra, N. and Sharma, N. (2022). Cancer prediction using machine learning, *2022 2nd International Conference on Innovative Practices in Technology and Management (ICIPTM)*, Vol. 2, pp. 217–221.

Waykole, R. N. and Thakare, A. D. (2018). Intelligent classification of clinically actionable genetic mutations based on clinical evidences, *2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*, pp. 1–4.

Yoon, H.-J., Robinson, S., Christian, J. B., Qiu, J. X. and Tourassi, G. D. (2018). Filter pruning of convolutional neural networks for text classification: A case study of cancer pathology report comprehension, *2018 IEEE EMBS International Conference on Biomedical Health Informatics (BHI)*, pp. 345–348.

Zhang, Y. (2021). Research on text classification method based on lstm neural network model, *2021 IEEE Asia-Pacific Conference on Image Processing, Electronics and Computers (IPEC)*, pp. 1019–1022.