Text Classification on Diabetes Mellitus Symptom and Treatment Documents Using Machine Learning Approaches

FINAL YEAR PROJECT

Presentation Video: https://youtu.be/IbzulNp1gWY

Code Demo Video: https://youtu.be/NWCFmk5O_mo

Presented By :
Phang Cheng Yi
A20EC0131

Supervised By :
Dr Sharin Hazlin Binti Huspi
Dr Ahmad Najmi Bin Amerhaider Nuar

TABLE OF CONTENTS

CHAPTER 1
INTRODUCTION

INTRODUCTION

DIABETES MELLITUS (DM)

- Metabolic disorder characterised by excessively increased blood glucose levels with numerous subtypes
- Severe and varied symptoms of hyperglycemia include abnormalities in the metabolism of carbohydrates, fats, and proteins
- Early detection and efficient management is required but challenging

TEXT CLASSIFICATION

- A type of the NLP unstructured text analysis techniques
- A predetermined label or tag will be given to each document in the dataset by the classifier

MACHINE LEANRING

- A discipline of artificial intelligence (AI) that enables computers to perform tasks and learn from experience regardless of whether they are being specifically programmed

PROBLEM BACKGROUND

The number of people suffered from the disease Diabetes Mellitus (DM) keep increasing

A lot of publishments regarding to diabetes for early diagnosis, treatment, management and prevention

Early diagnosis by identifying the symptoms is significant to ascertain suitable treatments

Previous studies mostly used clinical data and patient medical record in classification using machine learning methods such as Fine Decision Tree and SVM but lack of study focus on the classification for medical journal articles that describing the symptoms and treatments of DM

Difficulty to discover and to classify the important information from numerous documents for better understanding and is time consuming

Potential to develop text classification model for diabetes symptom and treatment documents using machine learning approaches to optimize the diabetes diagnosis and management

PROBLEM STATEMENT

- *Overwhelming amount of medical literature and research on DM, which can hinder the efficiency of early detection and the effectiveness of management for diabetes patients and doctors*
- *Keeping up with the latest research discoveries and therapeutic approaches is getting harder as diabetes becomes more common and more complex*
- *Lack of research on the application of machine learning approaches for text classification on DM symptom and treatment documents*
- *Text classification model is used to assist in finding crucial symptoms and methods of therapy for DM, enhancing the effectiveness and precision of information retrieval*

RESEARCH OBJECTIVES

GOAL

To develop and evaluate a machine learning-based text classification model for DM symptom and treatment documents

OBJECTIVES

- To identify the significant features that are relevant to Diabetes Mellitus (DM) symptoms and treatment in multiple documents.
- To perform text classification for a collection of DM documents dataset using machine learning methods.
- To evaluate the performance of machine learning models that apply five different machine learning methods through several model evaluation techniques.

RESEARCH SCOPE

1. To use a collection of articles from PubMed by National Center for Biotechnology Information (NCBI)

2. The documents in the dataset are chosen by focusing on the symptoms and treatments of DM

3. To use Term Frequency-Inverse Document Frequency (TF-IDF) algorithms to do feature extraction

4. To use machine learning algorithms to classify text documents

RESEARCH CONTRIBUTION

Contribute to the fields of natural language processing (NLP) and machine learning

Contribute to the information retrieval fields with the enhancement of retrieving process that enable domain stakeholders of DM to efficiently identify and classify the important symptoms and treatments for DM based on the analysis of a large corpus of medical documents

Potentially facilitate the development of more effective interventions for diabetes management.

CHAPTER 2
LITERATURE REVIEW

SUMMARY OF RELATED WORK

TEXT CLASSIFICATION

- Rasheed et al. (2018) performed text classification of Urdu language using three well known classifiers which are Decision Tree, SVM and KNN.
- Chowdhury and Schoen (2020) conducted classification of textual data obtained from research papers using SVM, Naïve Bayes, KNN and Decision Tree.

MULTILABEL CLASSIFICATION

- Rahul et al. (2020) proposed six machine learning algorithms, including Random Forest, Decision Tree, Logistic Regression, Naïve Bayes, SVM and KNN to classify toxic comments.
- Setiawan et al. (2023) focused on student feedback data for multilabel classification using machine learning methods such as SVM, KNN, Random Forest, and Decision Tree.

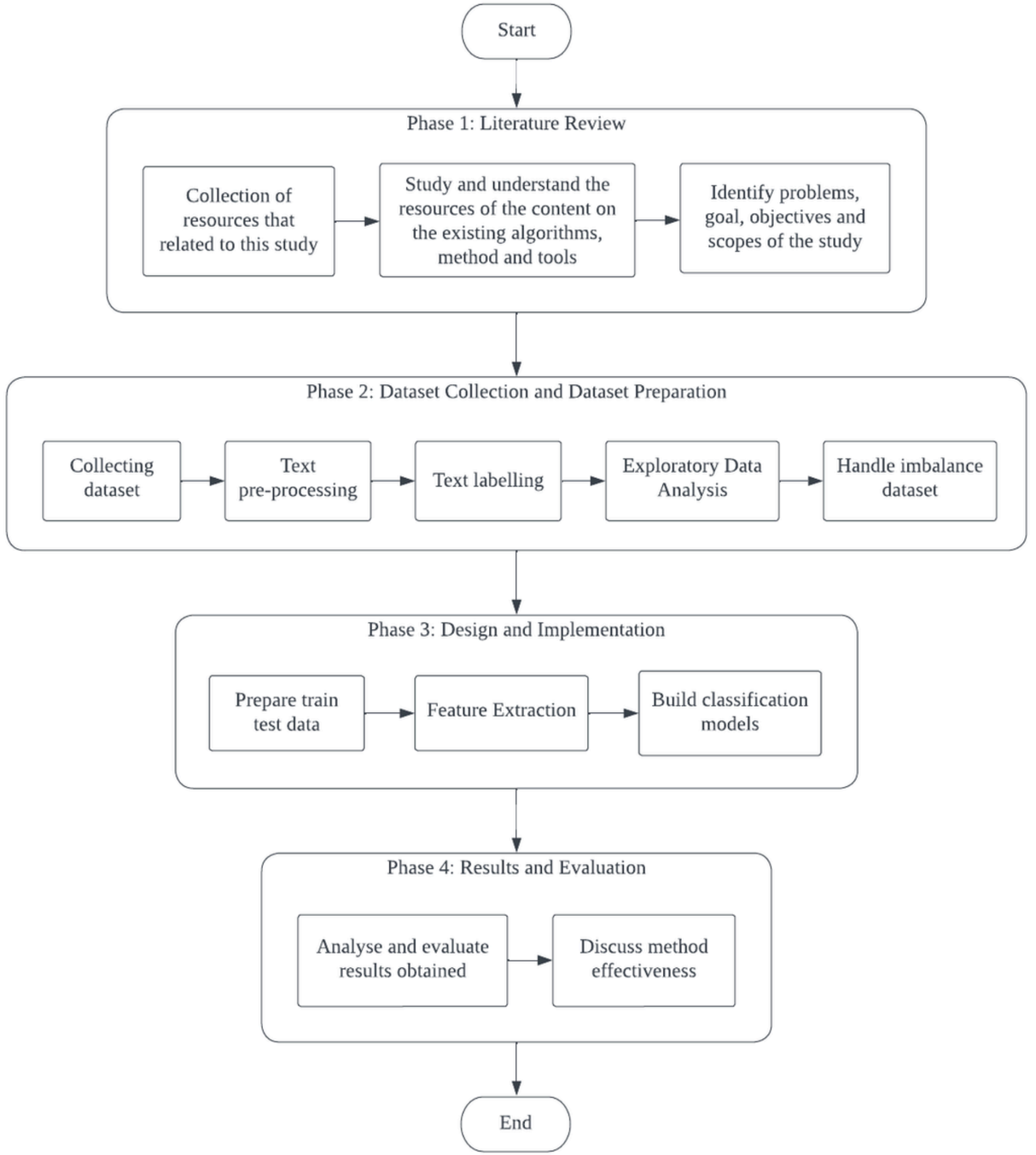
DISCUSSION

- *Thousands of research studies focus on diabetes mellitus (DM) but many studies use the Pima Indian diabetes dataset and eye fundus numerical images dataset for DM classification using machine learning.*
- Numerous publications pertinent to the most thoroughly studied diseases are readily available. These unstructured data have become the potential sources that can provide beneficial information to the medical experts in their diagnosis.
- Lack of studies that occupy thoughts with the narrative documents of DM by classifying their symptoms and treatments.
- Machine learning is the most used method for text classification, with varying performance depending on the dataset.
- Potential to conduct research studies on text classification for DM symptom and treatment documents using machine learning algorithms

CHAPTER 3

RESEARCH METHODOLOGY

RESEARCH WORKFLOW



Research Objective 1:
To identify the significant features that are relevant to Diabetes Mellitus (DM) symptoms and treatment in multiple documents.

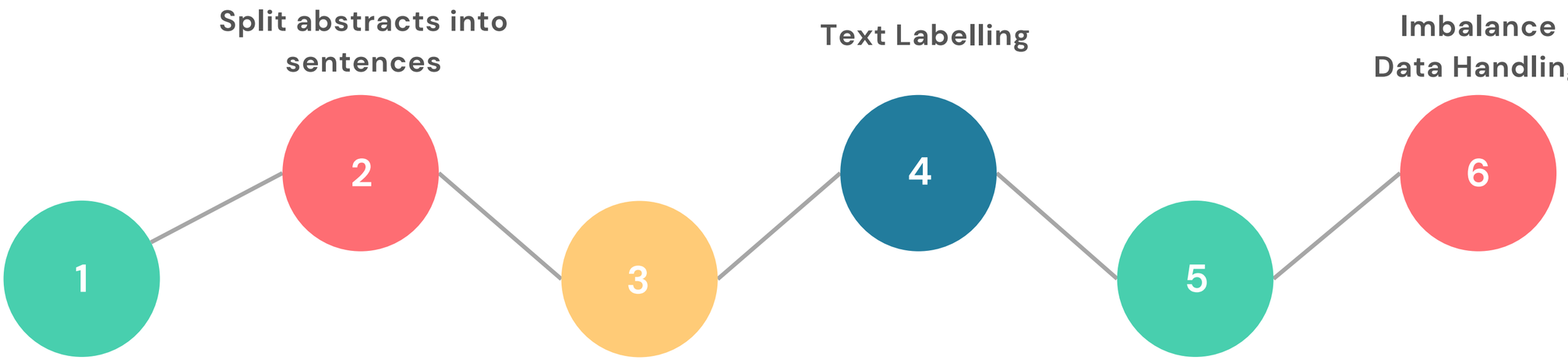
Research Objective 2:
To perform text classification for a collection of DM documents dataset using machine learning methods.

Research Objective 3:
To evaluate the performance of machine learning models that apply five different machine learning methods through several model evaluation techniques.

CHAPTER 4

RESEARCH DESIGN AND IMPLEMENTATION

DATASET PREPARATION



1 Extract Data

1019 articles
Search terms: "Type 1 Diabetes Mellitus", "Type 2 Diabetes Mellitus", "Symptom" and "Treatment".
Range of years: 2019 - 2024

2 Split abstracts into sentences

3 Text Pre-processing

Drop empty rows
Convert case to lowercase
Removal of non-related elements
Removal of punctuations
Removal of html tags
Removal of digit values
Tokenization
Removal of stopwords
Lemmatization

4 Text Labelling

5 Exploratory Data Analysis

6 Imbalance Data Handling

SOURCES OF KEYWORDS

AUTHORISED WEBSITES

- Mayo Clinic
- Diabetes UK
- WebMD
- WHO

RESEARCH PAPERS

- **Symptoms:**
  - (Rahaman, 2012) (Xu et al., 2021) (Garcia, 2011) (Parikh and Bhargava, 2021)
- **Treatments:**
  - (Tülüce et al., 2023) (Xie et al., 2018) (Pfeiffer and Klein, 2014) (Pamungkas et al., 2017)

Table 4.1 Source of defined keywords for symptoms

Authorised Official Websites				Research Papers				Keywords Summary (With same meaning)
Mayo Clinic [1]	Diabetes UK [2]	WebMD [3]	WHO [4]	(Rahaman, 2012) [5]	(Xu *et al.*, 2021) [6]	(Garcia, 2011) [7]	(Parikh and Bhargava, 2021) [8]	
✓	✓	✓				✓		High blood sugar [1], High glucose level [2][7], Hyperglycemia [3][7]
✓	✓	✓	✓	✓	✓	✓		Losing weight without trying [1], Thinner [2], Unplanned weight loss [3], Losing weight unintentionally [4], Unusual/Sudden weight loss [5][6], Weight loss [7]
✓	✓	✓	✓	✓	✓	✓		Urinating often [1], Toilet [2], Peeing more often [3], Needing to urinate more often than usual [4], Frequent urination [5], Polyuria [6][7]
	✓	✓		✓	✓	✓		Increased hunger [2], Hunger [3], Extreme hunger [5], Polyphagia [5][6][7]
✓	✓	✓	✓	✓	✓	✓		Feeling more thirsty than usual [1], Thirsty [2], Dry mouth [3], Feeling very thirsty [4],

Table 4.2 Source of defined keywords for treatments

Authorised Official Websites				Research Paper				Keywords Summary (With same meaning)
Mayo Clinic [1]	Diabetes UK [2]	WebMD [3]	WHO [4]	(Tülüce *et al.*, 2023) [5]	(Xie *et al.*, 2018) [6]	(Pfeiffer and Klein, 2014) [7]	(Pamungkas *et al.*, 2017) [8]	
✓	✓	✓	✓	✓	✓	✓		Oral or other drugs [1], Tablets and medication [2], Medications [3][4], oral drugs [5][6], Pharmacotherapy [6][7]
	✓			✓			✓	Emotional support [2], Psychosocial adaptation [8], Psychological [8]
✓	✓	✓	✓	✓	✓	✓		Insulin [1][2], Insulin pump [3][4], Insulin treatment [6], Insulin therapy [7]
✓	✓	✓		✓		✓	✓	Healthy eating [1][5], Diet [2][8], Healthy diet [3], Dietary therapy [7]
✓		✓	✓	✓				Monitoring your blood sugar [1], Blood sugar monitoring [3][4], Blood glucose monitoring [5]
✓	✓	✓	✓	✓		✓	✓	Physical activity [1][5][7][8], Exercise

SUMMARY OF KEYWORDS

Symptoms:

- high blood sugar
- hyperglycemia
- high glucose level
- weight loss
- weight gain
- thinner
- losing weight
- polyuria
- frequent urination
- urinating often
- peeing more often
- polyphagia
- hunger
- hungry
- increased hunger
- extreme hunger
- polydipsia
- thirst
- thirsty
- excessive thirst
- feeling more thirsty than usual
- fatigue
- feeling fatigued
- tired
- feeling tired
- dry skin
- itchy skin
- itching
- blurry vision
- blurred eyesight
- eyesight blurred
- vision loss
- numbness
- tingling
- slow healing sores
- slow in wound healing
- delayed wound healing
- cuts and wounds take longer to heal
- infections
- irritable
- mood changes
- depression
- depressive
- depressive mood

Treatments:

- pharmacology
- pharmacological treatment
- pharmacotherapy
- drugs
- oral drugs
- tablets
- medication
- psychology
- psychotherapy
- emotional support
- insulin
- diet
- diet monitoring
- healthy diet
- diet therapy
- blood sugar monitoring
- exercise
- physical activity

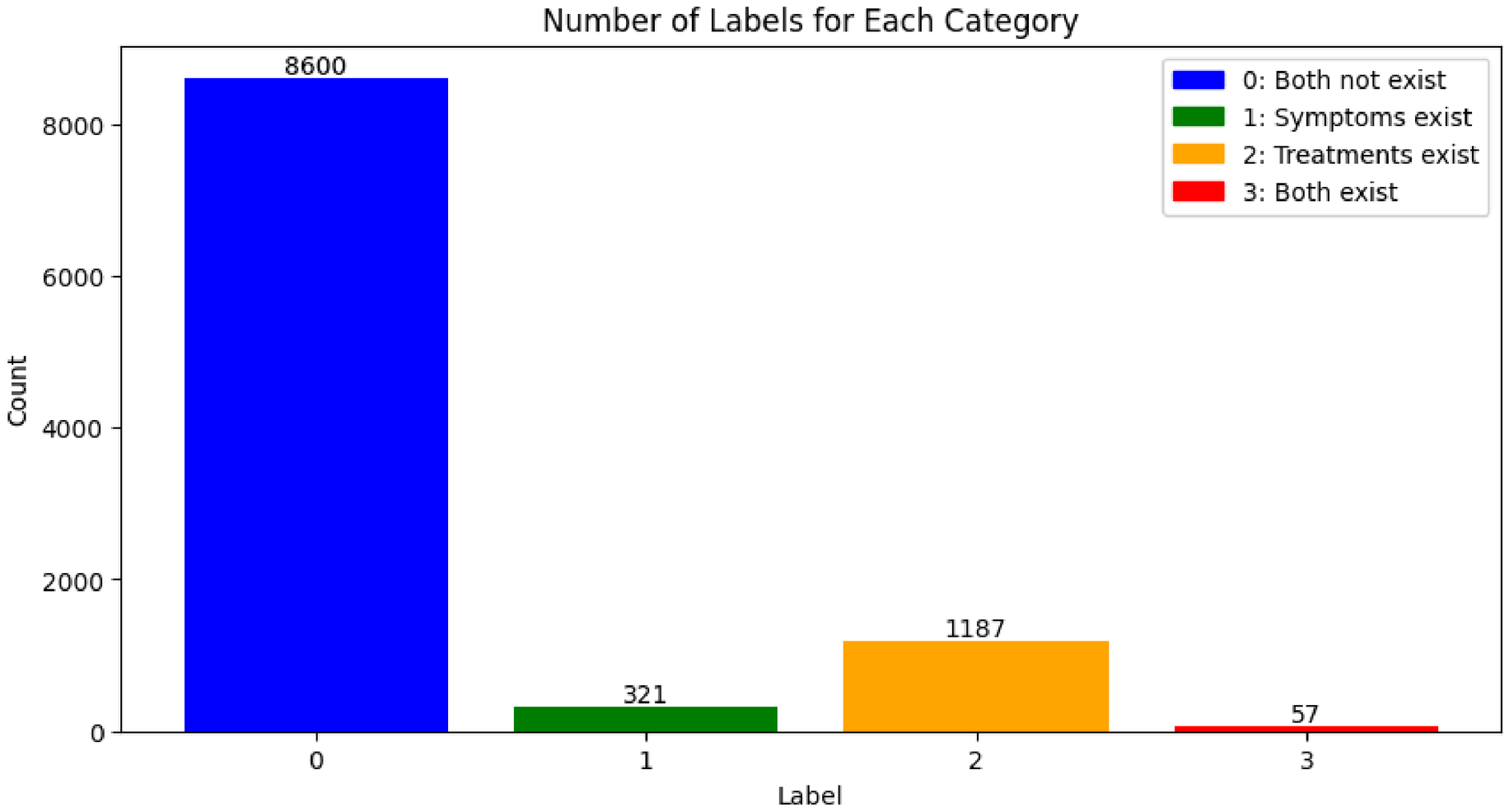
LABELLING RESULT

	A	B
1	sentences	label
2	carpal tunnel syndrome ct occurs often among individual diabetes	0
3	aim retrospective observational registry study examine whether individual dial	0
4	data ct diagnosis surgery collected sk ne healthcare register shr	0
5	total individual age year diagnosed ct included	0
6	data matched swedish national diabetes register ndr	0
7	cox regression model used calculate risk use surgical treatment	0
8	included individual ct diagnosis treated surgically diabetes	0
9	higher number individual diabetes treated surgically individual without diabete	0
10	cox regression model diabetes remained significant risk factor surgical treatn	0
11	individual type diabetes frequently treated surgically individual type diabetes	0
12	difference sex treatment	0
13	duration diabetes also risk factor surgical treatment diabetes type high hba c l	0
14	individual diabetes likely treated surgically ct individual without diabetes	0
15	individual type diabetes likely treated surgically ct individual type diabetes	0
16	background sexual complication people diabetes mellitus dm often neglected	0
17	neglect woman due associated stigma taboo	0
18	indian study scanty varied inconsistent regarding impact dm sexual functionin	0
19	studied pattern predictor sexual dysfunction woman dm	0
20	method crosssectional questionnairebased study comprising participant stuc	0
21	approval institutional ethic committee obtained	0
22	clinical anxiety depression screened using hospital anxiety depression scale	1
23	sexual dysfunction assessed female sexual function index scale fsfi predictor	0
24	result found woman dm sexual dysfunction compared control group p	0

- Keywords will undergo text preprocessing.
- The sentences are labelled by matching the keywords defined.
- Matching the keywords gram by gram in the sentences by using the ngram function in NLTK and loops.

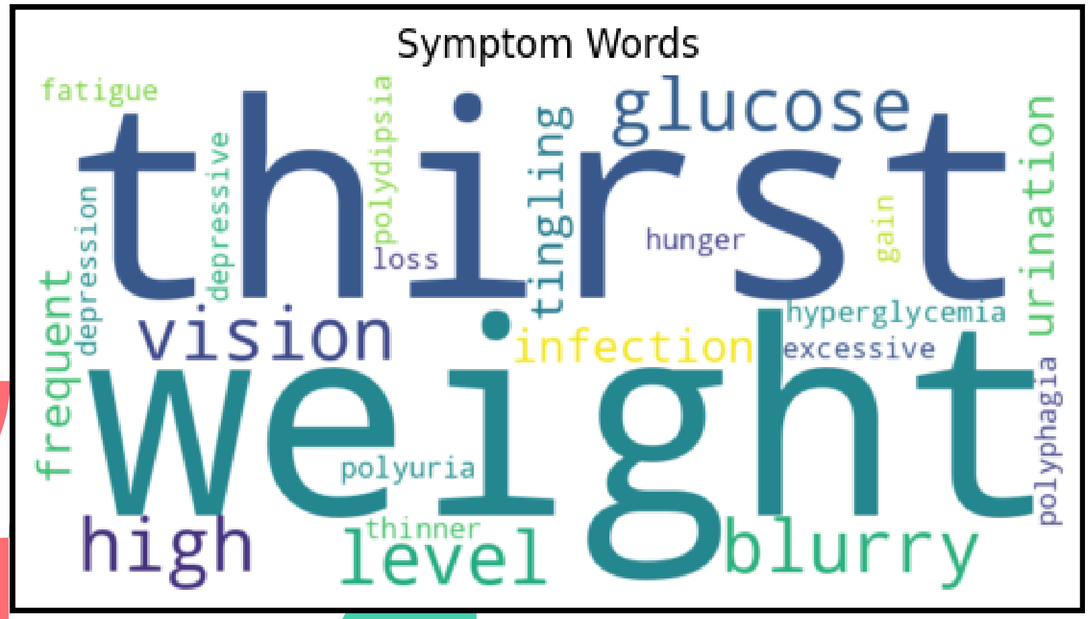
- **The sentences are tagged with 4 Labels:**
  - Label ‘0’: Both not exist
  - Label ‘1’: Symptom(s) exist
  - Label ‘2’: Treatment(s) exist
  - Label ‘3’: Both exist

BAR CHART

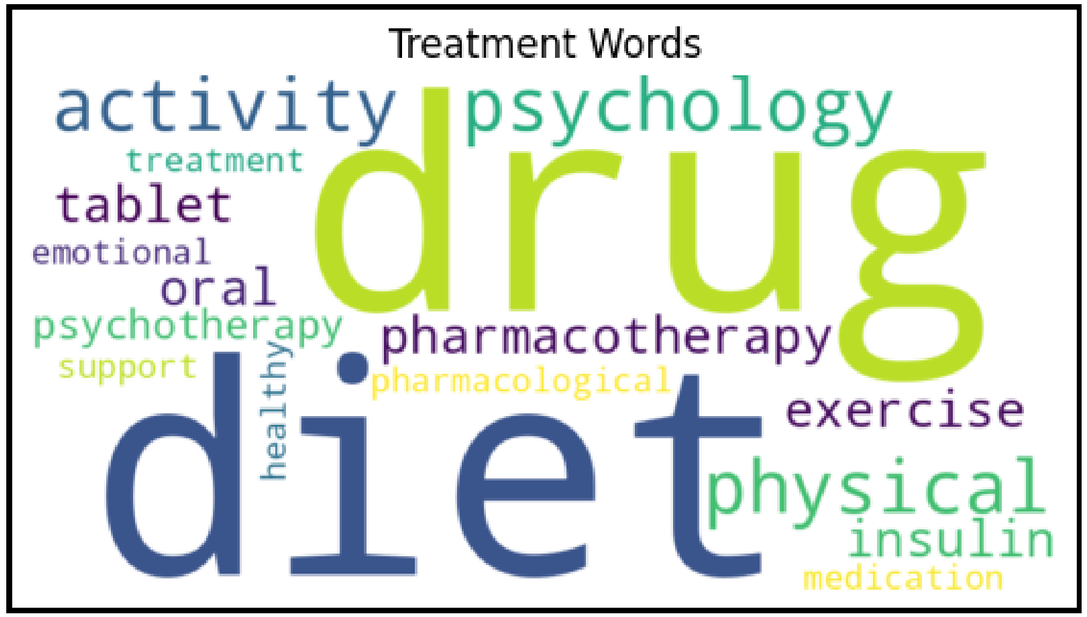

Number of Labels for Each Category
0: Both not exist
1: Symptoms exist
2: Treatments exist
3: Both exist
8600
321
1187
57
8000
6000
4000
2000
0
Count
0
1
2
3
Label

WORD CLOUD

SYMPTOM



TREATMENT



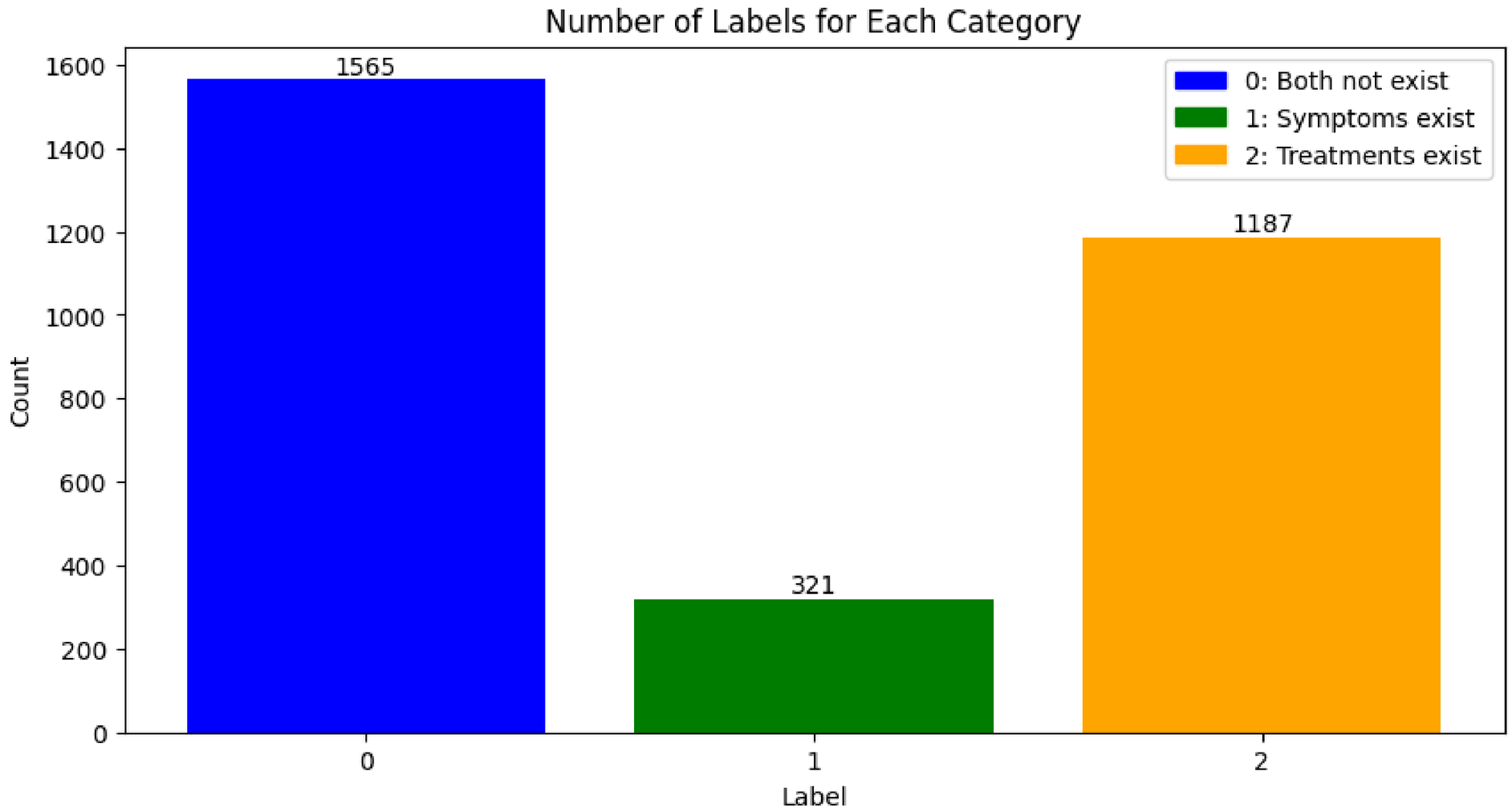
HANDLE IMBALANCE DATA

- The dataset is imbalanced, the 'Both not exist' category will undergo undersampling (resampling technique) to be reduced to 1565.

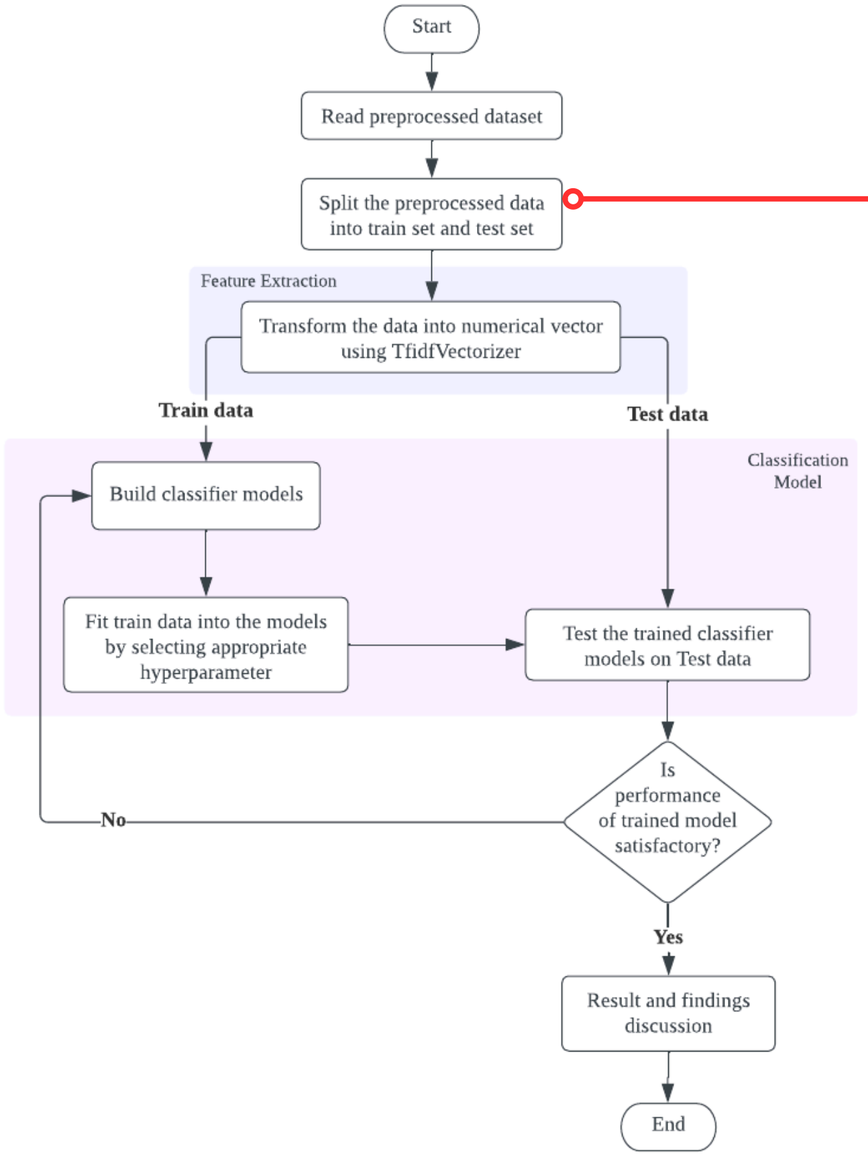
- Values of 1565 is obtained from the calculation of total number of other categories.

- For the 'Both exist' category, it will be removed since the amount of values is too small and it does not have significant impact for the model to achieve the research's objectives.

FINAL DATASET



MODEL BUILDING



- **Split dataset into three different training and testing ratio**

SLPIT 1

- Training: 70%
- Testing: 30%

SPLIT 2

- Training: 80%
- Testing: 20%

SPLIT 3

- Training: 90%
- Testing: 10%

CHAPTER 5

RESULTS, ANALYSIS AND DISCUSSION

SUPPORT VECTOR MACHINE (SVM)

Train Test Ratio	Accuracy (%)	Best Parameter
70:30	95.01	C = 1.1
80:20	96.75	C = 0.9
90:10	96.43	C = 1.1

HYPERPARAMETER TUNING

- The research uses a 'linear' kernel in SVM because it is more suitable for text data represented as high-dimensional TF-IDF vectors, while non-linear kernels may cause overfitting or underfitting.
- The gamma parameter is set to 'auto' for completeness but does not affect the linear kernel.
- The C parameter, crucial for balancing bias, is tested with values from 0.8 to 1.3, as the model's accuracy stabilizes from 0.8 and optimizes at C=1.3.

PERFORMANCE

- *For 70:30 split, the best performance is at C=1.1 with an accuracy of 95.01%. Precision, recall, and F1-score are consistently high for class 0, with class 1 having high precision but lower recall, and high scores overall for class 2.*
- *For 80:20 split, the best performance is at C=0.9 with an accuracy of 96.75%. Precision, recall, and F1-score are high for classes 0 and 2, with class 1 showing high precision at C=0.8 and C=0.9, and stable F1-scores starting from C=0.9.*
- *For 90:10 split, the best performance is at C=0.9 with an accuracy of 96.75%, but C=1.1 is chosen as optimal with 96.43% accuracy due to balanced precision, recall, and F1-score, avoiding overfitting in class 1.*
- *Across all train-test splits, C values from 0.9 to 1.1 provide the best balance between precision, recall, and F1-score.*
- *The SVM model shows the best overall performance at C=0.9 with a train-test split of 80:20.*

LOGISTIC REGRESSION (LR)

Train Test Ratio	Accuracy (%)	Best Parameter
70:30	93.82	C=5.0 solver='liblinear'
80:20	95.12	C=5.0 solver='liblinear'
90:10	95.45	C=5.0 solver='liblinear'

HYPERPARAMETER TUNING

- The C parameter controls regularization strength, and values 1, 3, and 5 are chosen to find an optimal balance.
- The 'class_weight' is set to 'balanced' to handle imbalanced classes by adjusting weights inversely proportional to class frequencies.
- The 'solver' parameter is tested with 'lbfgs' and 'liblinear' to determine the best optimization algorithm, with 'lbfgs' being efficient for multiclass problems and 'liblinear' suitable for smaller datasets and less computationally intense.

PERFORMANCE

- *The best parameters for all train-test splits are C=5.0 and solver='liblinear'. For 70:30 split, the highest accuracy is 93.82%, with class 0 precision improving as C increases, and stable precision and recall for classes 1 and 2 across all hyperparameters.*
- *For 80:20 split, the highest accuracy is 95.28% with C=3.0 and solver='liblinear', but C=5.0 is selected due to higher average precision, recall, and F1-score across all classes.*
- *For 90:10 split, the highest accuracy is 95.45% with C=5.0 and solver='liblinear', showing consistent performance improvements in precision, recall, and F1-score as C increases.*
- *'liblinear' solver generally performs better than 'lbfgs', especially with higher C values, and higher values of C improve performance, particularly recall in class 1.*
- *Increasing the training size enhances the model's performance, indicating the model benefits from more training data.*
- *Overall, the configuration with C=5.0 and solver='liblinear' is recommended for optimal performance.*

DECISION TREE

Train Test Ratio	Accuracy (%)	Best Parameter
70:30	96.43	max_depth=25 max_features=10000
80:20	97.00	max_depth=25 max_features=10000
90:10	96.69	max_depth=25 max_features=10000

HYPERPARAMETER TUNING

- Decision Tree constructs a tree-like decision structure based on probability, with each node testing an attribute, branches depicting test results, and leaf nodes containing class labels.
- Results vary with each run due to random splitting, so the average of 10 runs will be used for evaluation.
- Two parameters are defined: 'max_features' is set to 10,000 (10% of total features) for all train-test splits, and 'max_depth' is tested between 5 and 25, with the range 21 to 25 found optimal to avoid overfitting or underfitting.

PERFORMANCE

- *The best parameters for all train-test splits are max_depth=25 and max_features=10000.*
- *For 70:30 split, the highest accuracy is 96.43%. Class 0 has the highest precision and F1-score (0.96 and 0.97), with recall stable around 0.98-0.99. Class 1's best F1-score is 0.94 with max_depth=25. Class 2 shows the highest precision, recall, and F1-score with values of 0.98, 0.95, and 0.97.*
- *For 80:20 split, the highest accuracy is 97.00%. Class 0's highest precision and F1-score are 0.97, with recall consistent at 0.98-0.99. Class 1's best F1-score is 0.94, and Class 2's recall and F1-score are highest at 0.97.*
- *For 90:10 split, the highest accuracy is 96.69%. Class 0's highest precision and F1-score are 0.95 and 0.97, with recall stable around 0.98-0.99. Class 1's best F1-score is 0.92, and Class 2's highest precision, recall, and F1-score are 0.99, 0.97, and 0.98.*
- *Max_depth=25 provides the highest accuracy across all train-test splits and balances metrics well, especially for the "Symptom(s) Exist" and "Treatment(s) Exist" classes.*
- *The Decision Tree model's best performance is with an 80:20 train-test split, achieving the highest accuracy of 97.00%.*

K-NEAREST NEIGHBOR (KNN)

Train Test Ratio	Accuracy (%)	Best Parameter
70:30	78.85	n_neighbors=19
80:20	80.16	n_neighbors=20
90:10	78.24	n_neighbors=16

HYPERPARAMETER TUNING

- In KNN, the only hyperparameter tuned is 'n_neighbors'.
- Values for 'n_neighbors' from 5 to 25 were tested to observe performance and determine the optimal range.
- The final 'n_neighbors' value is set between 16 and 20 for performance comparison.

PERFORMANCE

- *For 70:30 split, the best accuracy is 78.85% with n_neighbors=19 , with moderate precision and recall for class 0, high precision but low recall for class 1, and stable performance for class 2.*
- *For 80:20 split, the best accuracy is 80.16% with n_neighbors=20 , with moderate precision and recall for class 0, high precision but low recall for class 1, and slightly better precision and recall for class 2 compared to class 0.*
- *For 90:10 split, the best accuracy is 78.24% with n_neighbors=16, 17, and 18, with the best parameter being n_neighbors=16 for slightly higher precision, recall, and F1-score for class 1 and 2.*
- *Class 1 consistently shows high precision but low recall across all splits, affecting its F1-score.*
- *Higher values of n_neighbors generally improve the model's accuracy, especially in the 70:30 and 80:20 splits.*
- *The KNN model's best performance is with n_neighbors=20, achieving the highest accuracy of 80.16% for an 80:20 train-test split.*

RANDOM FOREST

Train Test Ratio	Accuracy (%)	Best Parameter
70:30	94.14	max_depth=14 max_features=900
80:20	93.98	max_depth=14 max_features=900
90:10	93.51	max_depth=15 max_features=900

HYPERPARAMETER TUNING

- Random Forest builds multiple decision trees and selects the tree with the most votes as the final output, offering more stability and robustness than a single Decision Tree.
- The 'class_weight' is set to 'balanced' to improve classification of imbalanced classes.
- 'Max_features' is fixed at 900 for all train-test splits, determined to be optimal after extensive testing. 'Max_depth' is tested from 5 to 20, with 11 to 15 found to be the suitable range to avoid overfitting or underfitting.

PERFORMANCE

- *The best max_depth for the 70:30 and 80:20 splits is 14, while for the 90:10 split, it is 15.*
- *For 70:30 split, max_depth=14 and max_features=900 yield the highest accuracy of 94.14%, with class 0 showing stable precision and F1-score, and class 2 achieving the highest precision, recall, and F1-score.*
- *For 80:20 split, the highest accuracy is 93.98% with max_depth=14, with class 0 maintaining high precision and recall, and class 1 achieving the best F1-score of 0.87.*
- *For 90:10 split, the highest accuracy is 93.51% with max_depth=15, with class 0 showing improved precision, recall, and F1-score, and class 1 achieving the best F1-score of 0.90.*
- *Across all splits, class 1 precision and recall improve with increasing max_depth, with class 2 showing consistently high performance.*
- *Overall, max_depth=14 provides the best performance for the Random Forest model, particularly with the 70:30 train-test split.*

COMPARISON ON OVERALL PERFORMANCE (80:20)

Classifier	Accuracy (%)	Both Not Exist (0)			Symptom(s) Exist (1)			Treatment(s) Exist (2)		
		Precision	Recall	F1 Score	Precision	Recall	F1 Score	Precision	Recall	F1 Score
Support Vector Machine	96.75	**0.98**	0.96	**0.97**	**0.98**	**0.93**	**0.95**	0.95	**0.98**	**0.97**
Logistic Regression	95.12	**0.98**	0.93	0.96	0.87	**0.93**	0.90	0.93	**0.98**	0.96
Decision Tree	**97.00**	0.97	**0.98**	**0.97**	0.94	**0.93**	0.94	**0.98**	0.97	**0.97**
K-Nearest Neighbor	80.16	0.77	0.90	0.83	0.96	0.54	0.66	0.84	0.73	0.78
Random Forest	93.98	0.94	0.95	0.95	0.89	0.86	0.87	0.96	0.94	0.95

COMPARISON ON CONFUSION MATRIX (80:20)

Classifier	Both Not Exist (0)			Symptom(s) Exist (1)			Treatment(s) Exist (2)		
	Correctly Predicted	Incorrectly Predicted		Correctly Predicted	Incorrectly Predicted		Correctly Predicted	Incorrectly Predicted	
		(1)	(2)		(0)	(2)		(0)	(1)
Support Vector Machine	314	1	11	52	4	0	229	4	0
Logistic Regression	304	7	15	52	3	1	229	3	1
Decision Tree	319	3	4	52	4	0	226	7	0
K-Nearest Neighbor	294	5	27	30	22	4	169	64	0
Random Forest	311	5	10	48	8	0	219	13	1

SUMMARY OF COMPARISON BETWEEN 5 MODELS

*Precision, Recall, F1-score, Confusion Matrix

Classifier	Accuracy (%)	Both Not Exist (0)	Symptom(s) Exist (1)	Treatment(s) Exist (2)
Support Vector Machine	has the second highest accuracy (96.75)	• has the highest precision (0.98) and F1-score (0.97) • correctly predict 314 instances	• has the highest precision (0.98), recall (0.93) and F1-score (0.95) • correctly predict 52 instances	• has the highest recall (0.98) and f1-score (0.97) • correctly predict 229 instances
Logistic Regression	has the third highest accuracy (95.12)	• has the highest precision (0.98) • correctly predict 304 instances	• has the highest recall (0.93) but lowest precision (0.87) • correctly predict 52 instances	• has the highest recall (0.98) • correctly predict 229 instances
Decision Tree	has the highest accuracy (97.00)	• has the highest recall (0.98) and f1-score (0.97) • correctly predict 319 instances	• has the same recall (0.93) as SVM and LR • correctly predict 52 instances	• has the highest precision (0.98) and F1-score (0.97) • correctly predict 226 instances
K-Nearest Neighbor	has the worst performance (80.16)	• has the lowest precision (0.77) , recall (0.90) and F1-score (0.83) • correctly predict 294 instances	• has the lowest recall (0.54) and F1-score (0.66) • correctly predict 30 instances	• has the lowest precision (0.84) , recall (0.73) and F1-score (0.78) • correctly predict 169 instances
Random Forest	has slightly lower accuracy (93.98)	• has slightly lower precision (0.94), recall (0.95) and F1-score (0.95) as compared to SVM and Decision Tree • correctly predict 311 instances	• has lower precision (0.89), recall (0.86) and F1-score (0.87) that less than 0.90 • correctly predict 48 instances	• has slightly lower recall (0.94) and F1-score (0.95) as compared to SVM, LR and Decision Tree • correctly predict 219 instances

DISCUSSION

- **Top-Performing Models**: Support Vector Machine and Decision Tree, due to their high accuracy and balanced performance across all classes.
- **Well-Performing Models**: Logistic Regression and Random Forest, though with slightly lower metrics for the "Symptom(s) Exist (1)" class.
- **Underperforming Model**: K-Nearest Neighbor, which is less suited for this dataset due to high dimensionality and noise issues.

REASONS FOR PERFORMANCE

- **Data Characteristics:**
  - Well-defined boundaries for "Both Not Exist (0)" and "Treatment(s) Exist (2)" classes make them easier to classify.
  - "Symptom(s) Exist (1)" class is less frequent, making it harder for models to learn and distinguish accurately.
- **Model Strengths:**
  - SVM and Decision Tree: Excel due to their ability to capture complex patterns.
  - Logistic Regression: Performs well with linearly separable data structure.
  - Random Forest: Balances variance and bias effectively.
- **Model Limitations:**
  - KNN: Struggles with high dimensionality and noise.
  - Random Forest: Less effective for the "Symptom(s) Exist (1)" class.

Ultimately, the choice of model depends on the specific requirements and constraints of the application, but SVM and Decision Tree show the best adaptability and performance for this dataset.

CHAPTER 6

CONCLUSION AND RECOMMENDATIONS

CONCLUSION

01

Most of the classifiers show their best results in the text classification on DM symptoms and treatments using journal articles from PubMed website when the splitting is 80% for training data and 20% for testing data.

02

SVM was proven as the best model in terms of performance metrics and confusion matrix followed by Decision Tree, Logistic Regression and Random Forest.

03

The proposed machine learning algorithms have been proved that it is capable to be employed to classify text data based on the keywords defined for DM.

ACHIEVEMENTS

- Diabetes Mellitus (DM) symptoms and treatments in the datasets were successfully labelled based on the predefined keyword.
- Five machine learning models were built to classify the DM symptoms and treatments in the datasets.
- Performance of the models were evaluated and SVM outperforms in term of performance metrics and confusion matrix.

FUTURE WORKS...

1. Try different types of feature extraction methods to compare and observe the performance of the machine learning algorithms.

2. Implementation of hyperparameter tuning using search techniques like GridSearchCv and RandomizedSerachCV to further optimize the performance.

3. Implementation of imbalance data handling technique such as MLSMOTE to handle data imbalance problems in multi-label text data.

THANK YOU