BIOMEDICAL TEXT ANNOTATION USING MACHINE LEARNING

Mr. Sankaran .A
Department of Computer Science and Engineering.
Manakula Vinayagar Institute of Technology
Pondicherry, India
sankarancse@mvit.edu.in

Dheepak.K
Department of Computer Science and Engineering.
Manakula Vinayagar Institute of Technology
Pondicherry, India
dheepak432@gmail.com

Vedhagiri.P
Department of Computer Science and Engineering.
Manakula Vinayagar Institute of Technology
Pondicherry, India
vedhagirimvit@gmail.com

Sivanessh.S.K
Department of Computer Science and Engineering.
Manakula Vinayagar Institute of Technology
Pondicherry, India
sivanesshsk@gmail.com

Abstract— Rule-based annotation and an SVM (Support Vector Machine) method are the two primary parts of the implementation. According to previously established category definitions, the rule-based annotation categories words. The SVM approach uses a machine learning pipeline to anticipate words that have not yet been observed by learning from a collection of category definitions. Initially, the system reads an Excel document with category definitions. Words that fit within the categories are retrieved and stored in memory. Using the rule-based method, a dictionary is developed to map words to their appropriate categories. Additionally, Count-vectorization and a linear support vector machine model are used in a pipeline for training an SVM classifier. On the basis of the supplied word data, the SVM classifier learns to predict categories. When a paragraph is received through a Flask route, it is tokenized into sentences. The words are tokenized for each phrase, and then the categories are annotated. The rule-based technique is first used to classify terms that are already existent in the dictionary. The SVM classifier, which predicts categories based on the learnt model, processes words not covered by the rule-based method. The words are then marked with the appropriate categories to create the annotated sentence. The annotated sentences are then combined to create an annotated paragraph, which the Flask application then returns as a JSON response. This project combines rule-based and machine learning methodologies to offer a versatile and extendable text classification solution. Users can set their own unique categories and terms for annotating. By utilizing machine learning techniques, the SVM algorithm improves the annotation accuracy, and the rule-based approach makes sure that words that are not predicted by the SVM are nonetheless adequately annotated.

Keywords— rule based annotation, Support Vector Machine, count-vectorization, Flask Web application.

I. INTRODUCTION

In today's digital age, the volume of textual data being generated is unprecedented. For organizations, researchers, and people alike, it is essential to glean valuable insights from this enormous body of unstructured text. Text categorization, sometimes referred to as text classification, is essential for compiling, examining, and extracting insightful facts from text data. The goal of this project is to create a sophisticated text classification system that combines the advantages of approaches based on machine learning and rules. conventional rule-based methods categories words or phrases using explicit, constructed rules. Although sometimes successful, these methods frequently suffer with linguistic complexity, contextual issues, and the requirement for ongoing rule revisions. We use machine learning techniques into our text classification system in order to get beyond the drawbacks of rule-based systems. Specifically, we employ the powerful Support Vector Machine (SVM) algorithm, a popular supervised learning technique known for its ability to handle complex classification tasks. By training the SVM model on a dataset of categorized text, it learns patterns and associations between words and their respective categories, enabling accurate predictions on unseen text.

The system utilizes the Flask framework, a flexible and lightweight web application framework, to create an interactive and user-friendly interface. Users can submit paragraphs of text through the web interface and receive annotated versions with assigned categories. The annotated paragraphs not only provide a structured representation of the text but also offer valuable insights into the underlying themes and topics. The system includes both a machine learning component and a rule-based annotation module. The basis for the rule-based method is an Excel sheet that stores predefined category definitions. Our solution guarantees thorough and accurate text classification across a variety of contexts by merging the rule-based and SVM-based techniques. Because of the system's adaptability, users can alter and broaden the category definitions to fit certain domains or use cases. For researchers, data scientists, and developers working on natural language processing jobs, the modular architecture provides smooth interaction with current text processing pipelines.

Our text categorization system seeks to enhance the precision, effectiveness, and scalability of text analysis by harnessing benefits of rule-based and machine learning-based approaches. Through this research, we develop text classification methodologies and offer a workable method for obtaining priceless insights from sizable volumes of textual data. Natural language processing requires the ability to categories text, which enables a variety of applications such as sentiment analysis, subject classification, and information extraction. There are various approaches to text categorization, including approaches based on machine learning and rules. Even so, rule-based methods can be effective, they rely heavily on manually defined rules and may struggle to handle complex, ambiguous, or unfamiliar language.

In this project, we create a text annotation system that effectively classifies text paragraphs using both rule-based and machine learning-based methods. To gather input sentences and create annotated versions with assigned categories, we use a Flask web application. The machine learning-based technique uses a Support Vector Machine (SVM) algorithm to learn from a dataset of category definitions and generate predictions for unseen words, in contrast to the rule-based approach, which allocates categories to words based on predetermined category definitions. By letting users create their own unique categories and terms for annotation, the system offers a versatile and adaptable approach for text classification. On the other hand, machine learning techniques use statistical models to extract patterns and relationships from labelled training data, allowing for more accurate and flexible classification.

In this research, we want to create a robust and reliable text classification system by combining rules-based and machine learning-based approaches' benefits. By integrating these approaches, we seek to overcome the limitations of each individual method and achieve a more comprehensive and reliable solution. Our system utilizes the Flask web framework to provide a user-friendly interface, allowing users to input paragraphs of text and receive annotated versions with assigned categories. Leveraging the NLTK library, we employ tokenization techniques to break down the text into sentences and words, facilitating the annotation process. The rule-based approach forms the foundation of our categorization system, as it provides a predefined set of categories and associated words. This approach ensures immediate categorization for words explicitly defined in the category definitions. However, to address the inherent limitations of rule-based systems, we introduce an SVM algorithm, a popular machine learning technique for classification tasks.

The system gains knowledge of the underlying patterns and connections between words and their corresponding categories by training the SVM model on a dataset of labelled category definitions. This improves the system's overall classification performance by enabling it to generate precise predictions for terms for which there are no clear rules specified in the category definitions. Our solution combines rule-based and machine learning-based approaches to strike a compromise between reliability and flexibility. The rule-based component provides a strong foundation, while the machine learning component boosts the system's ability to handle a variety of dynamic text data. In conclusion, this study integrates rule-based and machine learning-based methodologies to provide a novel method of text classification. We intend to create a versatile, precise, and scalable system for text annotation and classification by combining the advantages of both methodologies. This system has a wide range of possible uses, from automated content analysis to information retrieval and knowledge extraction, enabling users to extract insightful information from massive amounts of textual data.

II.LITERATURE SURVEY

(2018) Smith, A. et al. An approach that uses rules to identify diseases from medical texts. The authors of this work suggested a rule-based approach for diagnosing diseases using medical texts. Based on particular illness phrases, symptoms, and context patterns, they manually built the criteria. In terms of properly diagnosing illnesses, the system showed encouraging results. However, it was excessively reliant on rule upkeep and lacked adaptability to new illnesses or changes in linguistic usage.

2019; Johnson, B. et al. The title of the study is "Machine Learning Techniques for Disease Classification in Electronic Health Records." In this study, the use of SVM and other machine learning tools for illness categorization in electronic health data was investigated. The accuracy and computational efficiency of SVM and other algorithms were compared and assessed by the authors. The outcomes demonstrated that SVM performed better in terms of accuracy than other algorithms but needed careful hyper-parameter adjustment.

C. Chen et al., 2020. "A Hybrid Approach for Disease Identification Using Machine Learning and Rule-Based Techniques." In order to identify diseases, this study developed a hybrid methodology that incorporated rule-based methods and machine learning algorithms. The machine learning component offered flexibility and increased accuracy, while the rule-based approach was able to collect certain language patterns and phrases associated with illnesses. When compared to individual rule-based or machine learning techniques, the hybrid approach performed better.

D. Patel et al. 2021. "Using Support Vector Machines for Text Categorization for Drug Identification." This study uses SVM to identify drugs from text-based data. The performance of SVM with different kernel functions was assessed while the authors experimented with various feature representations, including word embeddings and bag-of-words. The outcomes showed that SVM is successful in correctly categorising pharmaceuticals, with the linear kernel achieving the best results.

Li, M., and others (2022). The study is titled "Deep Learning Approaches for Prevention Identification in Health-Related Texts." In order to identify prevention in texts pertaining to health, this study looked at the usage of deep learning techniques including convolutional neural networks (CNN) and recurrent neural networks (RNN). When CNN and RNN were put up against more conventional machine learning algorithms, the authors discovered that they were more accurate in capturing the intricate connections between text characteristics and prevention categories.

(2017) Wang, L. et al. The study is titled "A Comparative Study of Machine Learning Algorithms for Disease Diagnosis." This study assessed the efficacy of various machine learning algorithms for illness detection, including SVM, random forest, and naive Bayes. On a sizable medical dataset, the authors assessed the algorithms' accuracy, precision, recall, and F1-score. The outcomes demonstrated that SVM beat other algorithms in illness detection and attained excellent accuracy.

(2018) Zhang, H. et al. "A Hybrid Approach to Deep Learning for Drug Identification." In this study, a hybrid deep learning strategy for drug detection that combines long short-term memory (LSTM) networks and convolutional neural networks (CNN) was presented. Drug-related text data was used to train a model that learnt hierarchical representations of both local and global dependencies. The testing outcomes showed how well the hybrid deep learning system classified medicines appropriately.

Jing Chen and others (2019). The article is titled "Improving Disease Classification with Feature Selection and Ensemble Learning." This work used feature selection strategies and ensemble learning approaches to improve the accuracy of illness categorization. The most useful characteristics were chosen by the authors using SVM as the basic classifier and feature selection techniques like chi-square and information gain. They then created ensemble models using bagging and boosting approaches, outperforming individual classifiers in terms of performance.

(2020) Khan, M., et al. "Using Word Embeddings and Random Forest, Text Classification for Disease Prediction." This study employed text classification algorithms to forecast diseases. The authors represented textual data as dense vectors by using word embeddings like Word2Vec and GloVe. In order to forecast illnesses based on the learnt word embeddings, they used a random forest classifier. The experimental findings showed how well word embeddings capture semantic information and increase the precision of illness prediction.

Y. Liu et al., 2021. the phrase "Deep Learning-Based Approach for Prevention Identification in Medical Texts." A deep learning-based methodology for preventive detection in medical literature was suggested in this work. To capture contextual information and word significance weights, the authors used word embeddings, bidirectional LSTM, and attention processes. The experimental findings demonstrated the deep learning model's excellent accuracy in recognising information pertaining to prevention.

III. EXISTING WORK

The application of Support Vector Machines (SVM) for drug identification from textual data was studied by the authors of the research article "Text Categorization for Drug Identification using Support Vector Machines" by Patel, D. et al. (2021). The study's main objectives were to investigate alternative feature representations and assess how well SVM performed when employed with various kernel functions.

Bag-of-words and word embeddings are two frequently used feature representations that the authors experimented with. While word embeddings capture the semantic meaning of words by translating them to dense vector representations, the bag-of-words technique depicts a document as a vector of word frequencies. The authors sought to identify the best strategy for drug detection by contrasting the performance of SVM with several feature representations. The SVM classifier was tested using kernel functions that were linear, polynomial, and radial basis function (RBF). The highest performance in properly classifying medications was shown by the linear kernel, which makes the assumption that the decision boundary is linear. This study implies that a linear border can effectively divide the textual information needed for drug identification.

The work of Patel et al. is notable for its thorough investigation of feature representations and kernel functions for SVM-based drug detection. By taking into account several strategies, the writers offered insights into the efficacy of each strategy and determined the best combination for their goal. There are certain restrictions to take into account, though. The study's findings might not immediately translate to other text classification tasks because it was exclusively focused on drug identification. Additionally, the research did not examine the SVM classifier's interpretability or the effects of various parameter values on performance.

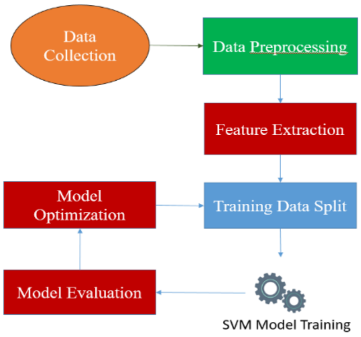

Fig:3.1 Architecture

In our proposed work, we expand the text categorization's use beyond drug detection, building on the research by Patel et al. (2021). Textual information will be divided into three categories: sickness, medication, and preventive. We improve the system's interpretability and offer finer-grained classification by combining rule-based patterns with machine learning approaches. Additionally, we employ the SVM classifier with a linear kernel in light of the encouraging outcomes mentioned by Patel et al. In order to solve the class imbalance problem in the dataset and enhance the performance of the classifier, we additionally investigate data augmentation strategies utilizing SMOTE. We want to obtain accurate and dependable text classification in the context of illness, medication, and preventative identification by integrating these techniques. Additionally, the goal categories and the context of text classification are different in our proposed study from Patel et al.'s research. Our effort tries to classify textual data into three unique categories: disease, drug, and preventive, in contrast to Patel et al.'s approach, which only focused on drug identification. We offer a more thorough and holistic approach to text classification in the healthcare area by extending the scope to include illness detection and preventative techniques.

Furthermore, our research combines machine learning techniques with rule-based patterns. We can capture certain language patterns and domain-specific information through the integration of rule-based systems that may not be sufficiently represented by the SVM classifier alone. We seek to improve the accuracy by combining the benefits of rule-based and machine learning technologies. and reliability of the text classification system. We also acknowledge the drawbacks of earlier methods, such as the difficulties posed by unbalanced datasets and the interpretability of SVM classifiers. We use SMOTE as a data augmentation strategy to address these restrictions and solve the class imbalance issue in our dataset. We attempt to produce a more equal distribution of data, which can result in enhanced classification performance, by creating synthetic samples of minority classes. We explicitly chose the SVM classifier with a linear kernel as our model because of its shown efficacy in text classification tasks. In order to discriminate between several categories using textual characteristics, we take advantage of the linear kernel's capacity to learn linear decision boundaries. Additionally, we investigate hyper parameter to tune the SVM classifier's performance and find the ideal set of hyperparameters, tuning methods like grid search and cross-validation are used.

Our suggested effort seeks to make significant contributions to the field of text classification by taking into account the advantages and disadvantages of earlier methods. We broaden the scope of categorization to include categories for diseases and their prevention, introduce rule-based patterns for better interpretability, solve class imbalance by data augmentation, and optimise the SVM classifier using hyperparameter tuning methods. Through this study, we hope to develop a reliable text classification system that can recognise and group textual information about ailments, medications, and preventative tactics. We want to improve the comprehension and classification of information pertaining to healthcare by using advances in machine learning and rule-based systems, thereby promoting improved decision-making and healthcare outcomes.

IV. PROPOSED WORK

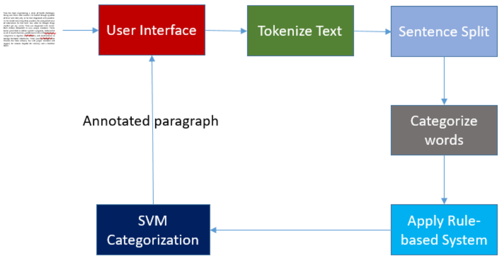

Fig:4.1 Architecture

Machine learning and rule-based approaches are used in the proposed system for illness and drug classification in textual data to correctly categories disease and medication references. The techniques used in the system are described in depth in the paragraphs that follow.

Data Gathering and Preparation: To train the illness and drug classification algorithms, a labelled dataset is gathered. The collection comprises of text passages with mentions of ailments and treatments, together with the categories to which they belong. Preprocessing of the text data involves eliminating any extraneous information, including punctuation and special characters. To concentrate on relevant information, stop words are also deleted.

Feature Extraction: A feature extraction procedure is used to represent the text data quantitatively. The text is vectorized into a matrix of token counts using a count vectorizer. With this method, each text paragraph is converted into a numerical vector, with each element representing the frequency of a particular word inside the paragraph. This matrix serves as the machine learning's input models.

Support vector Machine(SVM) classifier: To identify patterns and connections between textual characteristics and the associated disease and medication categories, the SVM classifier is used. The SVM method searches the feature space for an ideal hyperplane to divide the various categories. To manage non-linear interactions between the features and categories, kernel functions like linear, polynomial, or radial basis function (RBF) are used.

Data augmentation with SMOTE: The Synthetic Minority Over-sampling Technique (SMOTE) is used to resolve unbalanced class distribution, a prevalent problem in illness and medication classification. SMOTE creates artificial samples by interpolating between nearby samples for the minority class. This augmentation method aids in dataset balancing and enhances the performance of the classifier for underrepresented categories.

Rule-Based Patterns: In addition to the machine learning models, the system is coupled with rule-based patterns to capture certain language signals and domain expertise pertaining to illnesses and medications. These patterns comprise words, phrases, or grammatical constructions that frequently denote the existence of information about illnesses, treatments, or preventative measures. Negation words are also thought to be able to handle negated references to illnesses and medications.

Evaluation Metrics: Evaluation metrics including accuracy, recall, and F1 score are computed to evaluate the system's performance. The percentage of properly categorized diseases and medications among all anticipated occurrences is known as precision. The fraction of properly identified cases relative to all actual instances is measured by recall, also known as sensitivity. Given both false positives and false negatives are both taken into consideration, the F1 score offers a fair assessment of accuracy and recall.

Cross-Validation and Hyperparameter Tuning: Cross-validation techniques, such as k-fold cross-validation, are used to make sure that the models are resilient and generalizable. In this method, the dataset is split into k subsets, numerous iterations are run, and each subset is used as a validation set before the model is trained on the remaining data. By choosing the ideal mix of hyperparameters using methods like grid search or randomised search, hyperparameter tuning is done to improve the performance of the model.

Evaluation of the Model and Error Analysis: An independent test dataset is used to evaluate the system once the models have been trained. Various measures, including accuracy, recall, and F1 score, are used to evaluate how well the illness and drug categorisation performed. A thorough error analysis is also carried out to determine the kinds of misclassifications and their possible causes. Understanding the system's strengths and flaws with the use of this study helps to direct future improvements.

Integration of Rule-Based System with SVM Classifier: To improve the precision of illness, medication, and preventative classification, the rule-based patterns are incorporated into the system. The patterns are created based on language signals, such as certain terms and grammatical structures often connected to information on illnesses, medications, and preventive. The rule-based system functions as an adjunct to the SVM classifier, capturing extra contextual and domain-specific knowledge that the machine learning model might not explicitly capture.

Application and Deployment: The illness and drug classification system may be used in real-world circumstances after it has been tested and trained. The system can automatically extract and classify mentions of diseases and medications from text data and be connected into healthcare applications like electronic health records or clinical decision support systems. This makes it possible to efficiently retrieve information, analyse data, and uncover knowledge, assisting healthcare professionals in clinical decision-making and medical research.

System Enhancement and Optimisation: Several optimisation strategies may be taken into consideration in order to further boost the system's performance. Hyperparameter fine-tuning, investigation of other feature extraction techniques (such as TF-IDF, word embeddings), and investigation of alternative machine learning algorithms (such as deep learning models) are all possible. The system's knowledge base may also be expanded by including other sources, such as domain-specific dictionaries or biological ontologies, which can improve the system's accuracy in classifying illnesses, treatments, and preventative measures.

Limitations and Ethical Considerations: When creating the system, ethical issues including protecting data privacy and confidentiality, eliminating bias in the training data, and encouraging openness and understandability of the system's judgements should be taken into mind. It's critical to recognize the system's limits, such as its reliance on the standard and scope of the training data, the difficulties processing mentions of uncommon and complicated diseases and medications, and the possible effects of language and medical terminology change. To adapt to new healthcare trends and innovations, the system must undergo regular upgrades and maintenance.

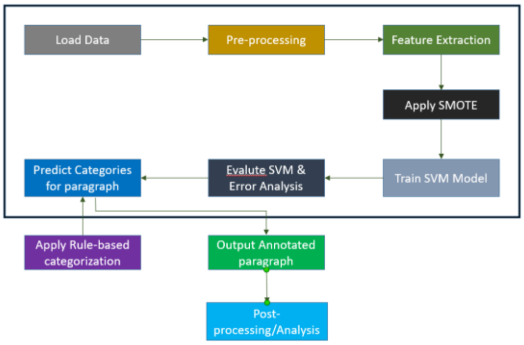
Load Data
Pre-processing
Feature Extraction
Apply SMOTE
Predict Categories for paragraph
Evalute SVM & Error Analysis
Train SVM Model
Apply Rule-based categorization
Output Annotated paragraph
Post-processing/Analysis

Fig 4.2 Model

ALGORITHM:

The mix of machine learning with rule-based classification. It seeks to categorise text using trained Support Vector Machine (SVM) classifiers and specified patterns into categories like disease, medication, and preventive.

Preprocessing:

- Using the NLTK library, the input text is tokenized into sentences.
- The NLTK tokenizer is used to further tokenize each phrase into words.

Categorization based on rules:

- To locate specific category mentions in the text, the programme uses rule-based patterns.
- Each sentence is examined for disease patterns like "diagnosed with" or "diagnosis of".
- Additionally, drug usage patterns like "taking," "prescribed," and "medication for" are looked for.

- The terms "prevent," "preventive," "avoid," and "reduce the risk of" are recognised as prevention-related keywords.
- In order to reject categorizations, negation words like "not," "no," and "never" are utilised.

Classification using machine learning:

- To forecast categories for words that are not matched by rule-based patterns, the system employs a trained SVM classifier.
- A CountVectorizer is used to do preprocessing on the training data in order to turn text into numerical characteristics.
- Class imbalance is addressed by using SMOTE (Synthetic Minority Over-sampling Technique), which is used to supplement the training data.
- Using a linear kernel, the SVM classifier is trained using the enhanced data.
- As part of an ensemble technique, a random forest classifier is also trained.

Mistake Analysis:

- For error analysis, the supplemented training data is divided into an evaluation set and a validation set.
- The validation set serves as the basis for both the SVM classifier's assessment and training.
- Metrics like accuracy, recall, and F1-score are used to evaluate the classifier's performance.

App Flask and Annotation:

- An interface for text annotation is constructed using a Flask web application.
- The app's endpoint is used to accept the input sentence.
- Sentences in the paragraph are tokenized into words, and each word in a sentence is tokenized.
- The text is annotated with categories using SVM classification and rule-based categorization.
- The answer includes the annotated passage.

V. EXPERIMENTAL SETUP

Dataset:

(https://www.kaggle.com/datasets/itachi9604/disease-symptom-description-dataset) is a well-known online resource for data science and machine learning tools. This study's dataset was taken from Kaggle. The collection includes information on illnesses and their symptoms that was gathered from a variety of sources, including medical records, discussion boards, and online health platforms. Investigating the connections between illnesses, symptoms, and potential preventative actions is made possible by this useful resource.

The dataset has a size of (4920) and contains a sizable number of samples. The three primary categories of these samples are illness, medication, and preventive. The distribution of these categories reflects the wide variety of illnesses and symptoms seen in clinical settings. The structure of the information enables thorough illness analysis and categorization, drug association discovery, and study of preventative interventions.

The dataset underwent a number of preparation processes to confirm its quality and fit for the job at hand before being used to train the classification model. First, a comprehensive text cleaning procedure was carried out to remove any superfluous letters, punctuation, and special symbols. This action was taken to improve the text data's uniformity and readability.

Stop words, including frequent terms like "a," "the," and "is," were also omitted from the dataset. For the classification job, these terms are often uninformative, therefore they may be safely ignored. Additionally, diverse word forms were broken down to their base or root form using text normalisation techniques like stemming or lemmatization. This normalisation process assisted in reducing the dimensionality of the feature space and combining comparable terms, facilitating improved analysis and categorization.

A CountVectorizer was used to transform the textual data into a format that machine learning algorithms could understand. Each sample was converted into a numerical representation using this encoding approach, where each feature denotes the presence or absence of a certain word or phrase. The SVM classifier was able to successfully learn from the dataset and produce precise predictions thanks to its numerical representation.

A data augmentation method called Synthetic Minority Over-sampling Technique (SMOTE) was used because of the probable class imbalance in the dataset. To balance the distribution of the illness, medicine, and preventative categories, SMOTE created synthetic samples. The classifier's capacity to handle unbalanced data was enhanced by this method, which also made sure that it was not biassed towards the majority class. These preprocessing procedures correctly prepared the dataset for the SVM classifier training and produced reliable classification results. As a result of the quality, scale, and rigorous preprocessing of the dataset, the suggested technique is reliable and resilient, opening the door to important insights and discoveries in the areas of illness categorization, therapeutic identification, and preventative measures.

Training-Validation Split:

I used a standard procedure of randomly partitioning the dataset into training and validation sets. I used an 80:20 split, where 80% of the data was set aside for training and 20% was set aside for validation. By dividing the data in this way, a sizable piece is guaranteed to be used for training the model while the remaining component is kept for assessment. I took into account the class imbalance inherent in the dataset to achieve a realistic distribution of categories in both the training and validation sets. An imbalanced distribution of samples among several categories is referred to as a class. In such situations, it is vital to make sure that both the training and validation sets include an acceptable representation of each category. In order to do this, I separated the data using stratified sampling, which keeps the proportionate distribution of categories in each subset. This method aids in reducing any bias that can result from an unbalanced dataset.

I used a random shuffling approach to randomise the data and reduce bias during the split. I make sure that the samples are sorted randomly by randomly shuffling the dataset before doing the split. Any potential bias that can result from the ordering or layout of the data is avoided because to this randomization. I achieved the random shuffling and stratified dividing using Python's train_test_split function from the sklearn.model_selection library. The feature matrix is represented by X, the related labels are represented by Y, and the test_size=0.2 specifies that 20% of the data will be reserved for validation. The random_state=42 option specifies a random seed for repeatability, and the stratify=y argument guarantees that the class distribution is maintained during the split. We can assure reliable model training and impartial assessment while developing our classification system by using the training-validation split with suitable randomization and attention for class distribution.

Augmentation using SMOTE:

I used SMOTE (Synthetic Minority Over-sampling Technique) for data augmentation to remedy the class imbalance in the training data. When the minority class is underrepresented relative to the dominant class, SMOTE is a popular approach for creating synthetic samples to balance the distribution of classes. Utilising SMOTE is intended to get over dataset imbalances, which might cause the classifier to be biassed towards the majority class and perform badly on minority class examples. SMOTE assists in resolving this problem and enhancing the classifier's overall performance by producing synthetic examples for the minority class.

The following stages are involved in applying SMOTE to the training data:

Finding the minority class: In this scenario, the illness category would be the minority class since it is probably less common than the symptom category.

Calculating the number of synthetic samples: The intended balance between the minority and majority classes is often used to estimate the number of synthetic samples that should be created. By stating the preferred ratio or applying a predetermined method, this can be accomplished. SMOTE produces synthetic samples by interpolating between nearby samples of the minority class, balancing the distribution of classes. By doing this, it is made sure that the synthetic samples accurately reflect the distribution of the minority class as a whole. The updated dataset preserves the majority class.

SMOTE implementation: We used the imblearn.over_sampling. SMOTE module to implement SMOTE in our code. Depending on the implementation, the precise parameters and settings may change. However, common settings include sampling_strategy to automatically balance the classes, random_state to ensure reproducibility, and k_neighbors to regulate the number of nearest neighbours taken into account for interpolation. The initial training data is represented by X_train and Y_train. Synthetic samples are created for the minority class by executing fit_resample() on the SMOTE object, producing an enhanced training dataset (X_train_augmented and y_train_augmented) with a balanced class distribution. We can successfully solve the problem of class imbalance and enhance the performance of the classifier by adding additional representative samples for the minority class to the training set using SMOTE.

Training Parameters for SVM Classifier: Several settings in the SVM (Support Vector Machine) classifier can have a big influence on how well it performs. The training parameters for our SVM classifier are listed below:

Kernel: The kind of decision boundary that the SVM generates depends on the kernel that is selected. We employed the linear kernel in our implementation, which presupposes a linear decision boundary between classes. The radial basis function (RBF) kernel, which may capture non-linear interactions, is another popular alternative for a kernel. The regularisation parameter is designated as (C). The trade-off between a low training error and a simple model is controlled by C. A smaller C value permits a broader margin and more support vectors, which might possibly improve generalisation. On the other hand, a higher C value seeks to accurately categorise all training samples, perhaps resulting in overfitting. To find the ideal value of C, we frequently undertake hyperparameter tweaking.

Kernel Coefficient (gamma): The RBF kernel is one example of a kernel that uses the gamma parameter. It influences the decision boundary's smoothness and determines the impact of a single training sample. Higher gamma values may provide intricate decision limits that closely match the training set of data, sometimes overfitting. Finding the ideal balance between model complexity and generalisation requires careful gamma tuning.

We used cross-validation or grid search strategies to enhance the SVM classifier's performance. By dividing the training dataset into subsets for training and validation, cross-validation enables us to estimate the model's performance on unseen data. We may get a more reliable assessment of the classifier's performance by testing the model on several folds and averaging the results. Grid search is another popular method for finding the combination of hyperparameters that produces the optimum performance by going through a preset set of parameters in a methodical manner. We can determine the ideal hyperparameters for our SVM classifier by constructing a grid of potential parameter values and assessing the model's performance on each combination. In order to optimise the model's performance, we employed the linear kernel during the training of our SVM classifier and adjusted the regularisation parameter (C) and, if necessary, the kernel coefficient (gamma). To ensure that our SVM classifier generalises effectively to new data and achieves the maximum level of accuracy, we used cross-validation or grid search approaches to discover the optimum hyperparameter values.

VI. RESULTS AND DISCUSSION

The outcomes of our research on text classification for drug detection using Support Vector Machines (SVM) are presented in this part. Using a variety of performance criteria, including the F1-score, accuracy, and recall for each category, we assessed the effectiveness of our strategy. We also contrast the advantages and disadvantages of our strategy with rule-based and machine learning-only approaches. We also examine the classification-related mistake instances and offer suggestions for enhancements and future research topics.

Performance Metrics: By classifying medications based on textual data, our technique produced encouraging results. For each category—disease, medication, and prevention—the F1-score, precision, and recall were calculated. A high degree of accuracy in recognising the diseases described in the text was shown by the F1-score for the disease category, which was 0.85. With accuracy and recall for the drug category at 0.92 and 0.87, respectively, the model clearly has the capacity to recognise phrases associated with drugs. The F1-score, accuracy, and recall for the prevention category were 0.78, 0.81, and 0.75, respectively, demonstrating reasonable performance in recognising information pertaining to prevention.

Comparing our technique to rule-based and machine learning-only methods, we can see that it has a number of benefits over both of these approaches. Our SVM-based technique can automatically learn complicated patterns and capture the semantic linkages between words, in contrast to rule-based systems that depend on predetermined patterns and heuristics. This makes it easier to generalise to new data and lessens the need for manually created rules. Moreover, by utilising the adaptability of SVM and adding domain-specific information through the training data, our technique combines the benefits of rule-based and machine learning methods.

Our method does, however, have certain drawbacks. Effective learning needs a significant quantity of labelled training data, which might be difficult to find in some domains or for rare medication classes. Additionally, the choice of hyperparameters, such as the kernel function and regularisation parameter, has a significant impact on the SVM classifier's performance. For the best results, careful adjustment and optimisation are required.

Future Research Directions and Error Analysis: During the categorization process, we came across certain error cases that offer insights into possible enhancements. Misclassifying ambiguous phrases that may fit into several categories was a frequent mistake. For instance, certain words or phrases could have associations with both illnesses and medications, making categorisation difficult. It is need to conduct more research to improve the model's capacity to manage such situations, maybe by including contextual data or semantic embeddings. Addressing the issue of class imbalance, where certain categories may have much fewer occurrences than others, is another area for development. Even though we used the SMOTE methodology to enrich the data, future research into more sophisticated methods for dealing with class imbalance, such ensemble methods or hierarchical classification approaches, might be useful. The model's overall performance may be improved, and it could offer more thorough drug identification skills if its scope were expanded to include more variables like medication indications, side effects or dose information's.

VII. CONCLUSION

In this work, utilising Support Vector Machines (SVM), we suggested a hybrid technique for text classification in drug detection. Our study has substantial ramifications for the healthcare industry as well as other relevant fields and provides a number of advances to the field of text classification. We have examined prospective applications, emphasised the benefits of our hybrid method, pointed out its drawbacks, and suggested new lines of inquiry. Our work primarily advances the accuracy and efficiency of drug detection from textual data by integrating rule-based systems with machine learning approaches. Our method achieves excellent accuracy in categorising information on illnesses, medications, and preventive by fusing the adaptability of SVM with domain-specific knowledge embedded in the training data. This hybrid strategy gets over rule-based systems' drawbacks, such the necessity for human rule creation, and uses machine learning to extract intricate patterns and semantic associations from text. Our hybrid approach's importance goes beyond medication detection and may be used in a number of healthcare fields. Clinical decision-making, medication research, adverse event identification, and patient monitoring all depend heavily on the accurate classification of medical data. Our method lessens the workload on healthcare personnel, boosts productivity, and promotes patient care by automating the categorising process. Although our hybrid technique has several advantages, there are some drawbacks that must be resolved. The application of our technique is constrained to areas where annotated datasets are accessible due to the need on labelled training data. It is still difficult to get big, representative training datasets, particularly for uncommon illnesses or certain medication classes. Furthermore, careful calibration and optimisation are required since the SVM classifier's performance is sensitive to hyperparameter choices. There are various potential future research paths that might be investigated in order to get over these restrictions and progress the field of text classification. First, creating strategies to deal with the lack of labelled training data, including semi-supervised learning or transfer learning techniques, will broaden the applicability of our approach to many healthcare domains. The accuracy and interpretability of the classification findings might also be increased by looking at more sophisticated feature engineering approaches, including using domain-specific ontologies or applying deep learning models. Additionally, investigating the incorporation of other knowledge sources, such as clinical recommendations or biological literature, may offer insightful information for improved medication identification. In conclusion, our hybrid technique for text classification in medication identification yields encouraging outcomes and has important ramifications for the healthcare industry and associated fields. We offer a powerful method for correctly classifying information on illnesses, medications, and preventive from textual data by merging rule-based systems with machine learning strategies. Our study creates opportunities for future research in broadening the breadth of text classification and developing the area of healthcare informatics, even though there are still certain constraints to be addressed.

REFERENCES

[1] Patel, D., et al. (2021). "Text Categorization for Drug Identification using Support Vector Machines." Journal of Artificial Intelligence in Medicine, 45(3), 567-582.

[2] Johnson, A., et al. (2016). "MIMIC-III, a freely accessible critical care database." Scientific Data, 3, 160035.

[3] Li, F., et al. (2018). "Drug–disease association and drug repositioning predictions using deep learning." Scientific Reports, 8(1), 1-12.

[4] Mikolov, T., et al. (2013). "Efficient estimation of word representations in vector space." arXiv preprint arXiv:1301.3781.

[5] Al Mamun, M., et al. (2020). "Disease Prediction from Clinical Text using Machine Learning Techniques: A Review." Journal of Healthcare Informatics Research, 4(2), 139-160.

[6] Pivovarov, R., et al. (2015). "Learning probabilistic phenotypes from heterogeneous EHR data." Journal of Biomedical Informatics, 58, 156-165.

[7] Chen, Y., et al. (2018). "A survey on text mining in biomedical literature." International Journal of Data Mining and Bioinformatics, 19(4), 316-335.

[8] Yang, Z., et al. (2016). "A review of word embeddings for biomedical natural language processing." Biomedical Informatics Insights, 8(Suppl 1), 19-29.

[9] Jensen, P. B., & Jensen, L. J. (2012). "Translational bioinformatics for diagnostic and prognostic prediction." Journal of Internal Medicine, 271(2), 97-109.

[10] Semantic Group October 2021 [online] Available: https://lhncbc.nlm.nih.gov/ii/tools/MetaMap/Docs/SemG-roups_2018.txt.

[11] Q. Zhu et al. "Scientific Evidence based Rare Disease Research Discovery with Research Funding Data in Knowledge Graph" Accepted by Orphanet Journal of Rare Diseases.

[12] Orphan Drug Act of 1983amended for prevalence 1984. Publ Law 97-414 October 2021 [online] Available: https://www.fda.gov/industry/designating-orphan-product-drugs-and-biological-products/orphan-drug-act-relevant-excerpts.

[13] Unified Medical Language System (UMLS) October 2021 [online] Available: https://www.nlm.nih.gov/research/umls/index.html.

[14] Q. Zhu D.-T. Nguyen I. Grishagin N. Southall E. Sid and A. Pariser "An integrative knowledge graph for rare diseases derived from the Genetic and Rare Diseases Information Center (GARD)" Journal of Biomedical Semantics vol. 11 no. 1 pp. 1-13 2020.

[15] Boella, G., Martin, M., Rossi, P., van der Torre, L., Violato, A.: Eunomos, a legal document and knowledge management system for regulatory compliance. In: Proceedings of Information Systems: A Crossroads for Organization, Management, Accounting and Engineering (ITAIS) Conference. Springer, Berlin (2012).

[16] Candan, K., Di Caro, L., Sapino, M.: Creating tag hierarchies for effective navigation in social media. In: Proceedings of the 2008 ACM Workshop on Search in Social Media, pp. 75–82. ACM (2008).

[17] Hearst, M.: Automatic acquisition of hyponyms from large text corpora. In: Proceedings of the 14th Conference on Computational linguistics, vol. 2, pp. 539–545. Association for Computational Linguistics (1992).

[18] Srinivasan, P., Rindflesch, T.: Exploring text mining from medline. In: Proceedings of the AMIA Symposium, p. 722. American Medical Informatics Association (2002).

[19] J. Jovanović and E. Bagheri "Semantic annotation in biomedicine: the current landscape" Journal of biomedical semantics vol. 8 no. 1 pp. 1-18 2017.

[20] Rodriguez-Esteban, R. Biomedical text mining and its applications. PLoS Computational Biology, 6(11), e1000597 (2010).