DRUG ADVOCACY SYSTEM BASED ON SENTIMENT ANALYSIS FOR PHARMACEUTICAL SCRUTINY USING MACHINE LEARNING PATTERNS

Dr.M.Ajay Kumar [1]***., Sree Kamya*** [2]***., K.Kiranmai*** [3]***., K.Ujwala*** [4] ***., J.Neeshitha*** [5]

1 Professor, Department of ECE., Malla Reddy College of Engineering for Women.,
Maisammaguda.,Medchal., TS, India (ajayajju126@gmail.com)
2, 3, 4, 5 B.Tech ECE, (19RG1A0452, 19RG1A0424, 19RG1A0421, 19RG1A0418),
Malla Reddy College of Engineering for Women., Maisammaguda., Medchal., TS, India

Abstract:

Since the outbreak of coronavirus, genuine clinical resources such as doctors, nurses, diagnostic tools, and medications have been difficult to come by. Many people in the medical community perish because of the widespread sorrow. Lacking access to proper medical care, many began self- medicating, worsening their health. Recently, there has been an uptick in creative efforts aimed at automating formerly laborious tasks, and machine learning has proven useful in many settings. This paper's goal is to provide a medication recommender system with the potential to significantly lessen experts' workload. As part of this study, we developed a medicine recommendation system that predicts sentiment based on patient reviews by employing a number of vectorization processes, including Bow, TF-IDF, Word2Vec, and Manual Feature Analysis, with the goal of enabling a number of different classification algorithms to recommend the best drug for a given disease. Precision, recall, f1score, accuracy, and area under the curve (AUC) were used to rate the anticipated emotions. With an accuracy of 93%, the findings demonstrate that the Linear SVC classifier using TF-IDF vectorization performs best of all models tested.

Index Terms—Drug, Recommender System, Machine Learning, NLP, Smote, Bow, TF-IDF, Word2Vec, Sentiment analysis

INTRODUCTION

Since the number of coronavirus cases is rising rapidly, countries are experiencing a severe shortage of medical professionals, especially in rural regions, where the availability of experts is lower than in major cities. It usually takes between six and twelve years to complete medical school and become board-certified. As a result, the number of medical professionals cannot be rapidly increased in a short period of time. In this trying time, the use of a Telemedicine framework should be promoted extensively [1]. Medical mistakes occur often at the present time. Inaccurate prescriptions harm around 200 thousand

people annually in China and over 100,000 annually in the United States. Since professionals create the solution based on reference to their relatively limited expertise, specialists make errors while prescription drugs at a rate of above 40% [2][3]. Patients in need of experts with extensive knowledge of microorganisms, antibacterial drugs, and patients are in a position to make an informed decision when selecting a top-tier medicine [6]. Every day, more and more research becomes available, and with it, more medicines and diagnostic tools for medical professionals to use. Therefore, it becomes more difficult for clinicians to decide which therapy or drugs to provide based on indications, prior clinical history, and other factors.

The proliferation of the internet and e-commerce websites has made product reviews a vital part of the buying process everywhere.

People everywhere have gotten in the habit of looking at various online sources, such as reviews and websites, before making any kind of major purchase decision. The domain of medical treatment or clinical therapies has been cared for seldom compared to rating expectation and suggestions on the E-Commerce industry.

There has been a rise in the number of people seeking medical advice and diagnosis over the internet.

The results of a 2013 poll conducted by the Pew American Research Center [5] show that around 60% of adults explore for health-related topics online and that 35% of users look for diagnosing health disorders online. For the sake of both doctors and patients, a medicine recommender framework is crucial in order to increase familiarity with the effects of various medications on various diseases.

The term "recommender framework" refers to a common kind of system that makes product suggestions to the user based on their perceived benefit and need. These models use feedback from clients to categorize their feelings and provide specific recommendations. Drug recommender systems use sentiment analysis and feature engineering to determine which medications to recommend to patients with various health issues. Opinion and attitude data, as well as other forms of emotional information, may be extracted from language using a process known as sentiment analysis [7]. On the other side, Feature Extraction Engineering (FEE) is a technique for enhancing model efficiency by creating new features from current ones.

There were five parts to this examination: A section devoted to the introduction, which offers a brief summary of why this study was deemed necessary, Previous research in this field is summarized in the "Related Works" section, the research's methods are laid out in the "Methodology" section, the results of the applied models are evaluated using a number of different metrics, the framework's limitations are discussed in "Discussion" section, and the work's limitations are summarized in the "Conclusion" section.

Two, a Literature Review

An effort has been made to integrate machine learning and deep learning methodologies to recommender systems, thanks to the rapid growth in AI technology in recent years. Recommender frameworks are now commonplace in the hospitality, retail, and food service industries. The medication reviews are significantly more intricate to analyze as it incorporates clinical wordings like infection names, reactions, and synthetic names that were used in the production of the medication

[8], which explains why there are so few studies in the field of drug proposal framework utilizing sentiment analysis.

In order to aid professionals in their quest for information on the drugs, the research [9] introduces GalenOWL, a semantic-enabled web platform. Using the patient's infection, sensitivities, and drug interactions, the study presents a paradigm for prescribing medication. The strength of GalenOWL comes from the precise combination of clinical information with ontological words translated from clinical data using international standards like ICD-10 and UNII.

In order to find the optimal treatment plan for patients, Leilei Sun [10] analyzed massive amounts of treatment data. The goal was to estimate the degree of similarity between treatment data using a fast semantic clustering technique. To evaluate the effectiveness of the proposed therapy, the author also developed a framework. New patients may be assigned the optimal treatment plans based on their demographic characteristics, geographic location, and medical conditions by using this framework. A patient database constructed from EMRs submitted by medical facilities all around the country.

This framework was shown to increase the success percentage of treatments.

This study [11] used Naive Bayes and a Recurrent Neural Network to do a cross-lingual sentiment analysis (RNN). Tweets written in other languages were translated into English using the Google Translate API. Outperforming Naive Bayes (75.21%) by a wide margin, the findings show that RNN (95.34%) performs better.

According to the research [12], the medicine that's best for a patient depends on their ability to take it. If the patient, for instance, has a weak immune system, the doctor should prescribe effective drugs. Immunity may be determined through a risk categorization approach that was proposed. More than 60 risk variables, including hypertension, alcoholism, and so on, have been accepted as a means of determining whether or not a patient has a functional immune system. Also developed was a prototype web-based system that employs a decision support system to aid in the selection of first-line medications for patients.

Using treatment data, Xiaohong Jiang et al. [13] compared three algorithms: the decision tree approach, the support vector machine (SVM), and the backpropagation neural network. Because of its superior performance across all three criteria, support vector machine (SVM) was chosen for the medicine recommendation module.

The error-checking system was also suggested to guarantee high-quality analysis, administration, andperformance.

For example, Mohammad Mehedi Hassan et al.
[14] created a cloud-assisted medication proposal (CADRE). CADRE may recommend medications with top-N related prescriptions based on adverse effects seen by patients.

The foundation of this suggested system is collaborative filtering methods, in which drugs are first clustered according to functional description data. However, after taking into account its drawbacks such as computationally pricey, cold start, and information sparsity, the model is changed to a cloud-aided strategy using tensor decomposition for enhancing the quality of experience of medicine recommendation.

Because of the role hashtags play in sentiment analysis, Jiugang Li, et al. [15] built a hashtag recommender framework based on the skip-gram model and employed convolutional neural networks (CNN) to train semantic phrase vectors. In order to categorize hashtags using LSTM RNN, these vectors use the characteristics. Conventional models like SVM and Standard RNN are shown to be inferior to this one in the results. When conventional artificial intelligence (AI) approaches are used, including support vector machine (SVM) and collaborative filtering, the semantic characteristics are lost, which has a significant impact on obtaining a reasonable expectation.

Methodologies (III)

The study makes use of the Drug Review Dataset (Drugs.com) available in the UCI ML repository [4]. There are six different types of information contained in this dataset: the drug's name in text format, a patient review, the patient's condition in text format, a numerical value representing the number of people who found the review helpful, the date the review was entered, and a numerical value representing the patient's satisfaction on a scale from one to ten. There are a total of 215063 records in it.
The suggested approach for developing a medication recommender system is shown in Fig.
1. There are four steps involved: gathering relevantdata, organizing it, analysing it, and giving advice.

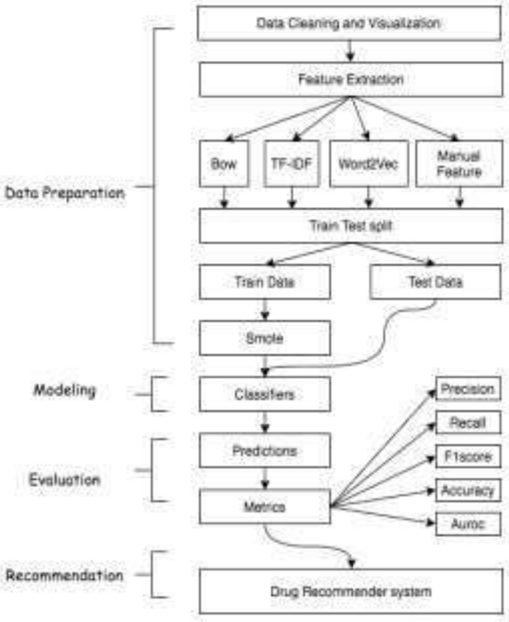


Fig. 1. Flowchart of the proposed model

A. Scrubbing and Displaying the Data

In this study, we used common Data preparation methods, such as checking for and cleaning up any duplicate rows, null values, and extraneous data or text. As can be seen in Fig. 2, all 1200 rows with null values in the conditions column were subsequently deleted. To avoid confusion, we check that each identifier is indeed one-of-a-kind.

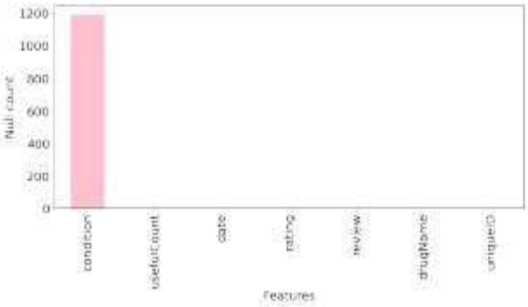


Fig. 2. Bar plot of the number of null values versus attributes

In Fig. 3 we see the top 20 diseases for which the widest range of treatments is now accessible. There are two green columns in this diagram, representing meaningless situations. By excluding these types of constraints from the final dataset, the number of rows drops to 212141.

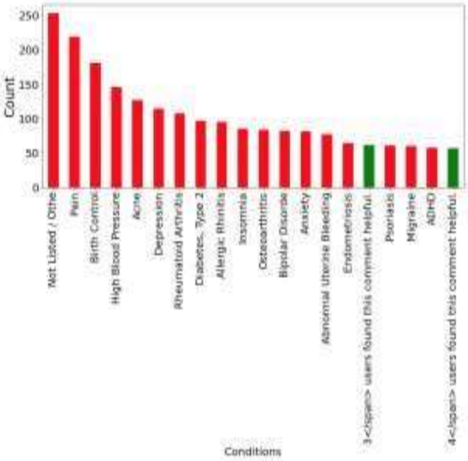


Fig. 3. Bar plot of Top 20 conditions that has a maximum number of drugs available

The ten-star rating's value counts are graphically shown in Fig. 4. There is a cyan tint for ratings lower than or equal to five, and a blue tint for ratings over five. Most people choose four characteristics, but selecting 10, 9, 1, 8, and 10 is more than twice as many. It demonstrates that people's answers are very polarized, with the positive side scoring higher than the negative side.

Review text was included with the condition and drug column since the terms for both conditions and drugs are highly predictive. Cleaning up the review text prior to vectorization is a necessary step before moving on to feature extraction. Text preparation is another name for this step. As a first step, we cleaned the reviews by removing any traces of HTML, as well as quotations, punctuation, URLs, etc. To prevent unnecessary repetition, the case of each word in the cleaned reviews was changed to lowercase, and the texts were tokenized to make them easier to process. Stop words such as "a," "to," "all," "we," "with," and so on were also deleted from the corpus. Lemmatization was applied to each token, bringing them back to their base forms. We classified every review into positive and negative categories based on the aggregated ratings from our users and used thisinformation to do a full sentiment analysis.

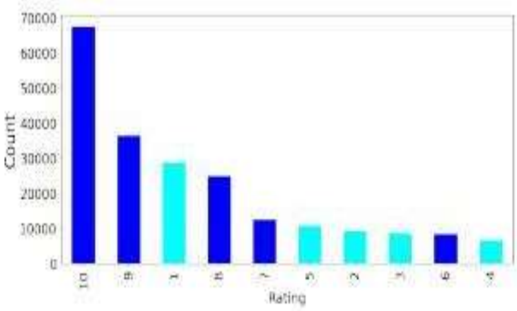


Fig. 4. Bar plot of count of rating values versus 10 rating number

Any review that receives a score between 6 and 10 from users is considered favourable.

Extracting Features, Part B

Sentiment analysis classifiers can't be built without first preparing the text for analysis via pre- processing. Text must be converted to numbers before machine learning algorithms can use them effectively. For example, numerical vectors. The bag of words (Bow) [16], TF-IDF [17], and Word2Vec [18] are all well-known and simple methods for feature extraction from textual data employed in this study. In addition to Bow, TF- IDF, and Word2Vec, we also employed certain feature engineering approaches to extract features by hand from the review column, resulting in amodel we call manual feature.

1) Bow, also known as Bag of Words [16], is a method used in natural language processing for token frequency counts. A phrase or token may be labelled as a single word (unigram) or an arbitrary number of words (n-grams). The (1,2) n-gram range was used for this analysis.

Figure 5 shows the structure of a phrase broken down into unigrams, diagrams, and trigrams. Since it takes into account all the words without taking into account how certain terms are extremely consecutive in the corpus, the Bow model results ina big matrix that is computationally costly to train.

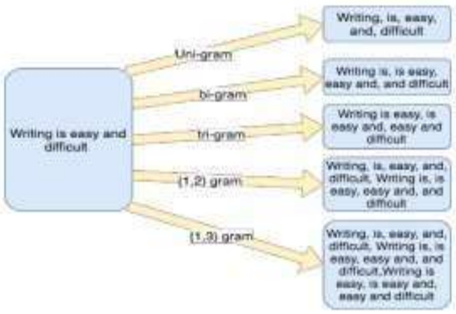

Writing is easy and difficult
Uni-gram
bi-gram
tri-gram
(1,2) gram
(1,3) gram
Writing, is, easy, and, difficult
Writing is, is easy, easy and, and difficult
Writing is easy, is easy and, easy and difficult
Writing, is, easy, and, difficult, Writing is, is easy, easy and, and difficult
Writing, is, easy, and, difficult, Writing is, is easy, easy and, and difficult,Writing is easy, is easy and, easy and difficult

Fig. 5. Comparison of various types of grams framed from a sentence

Two-Factor Inverse-Dependence Analysis (TF- IDF) [17] is a well-known weighing approach in which words are provided with weight rather than count. TF-IDF evaluates relevance, not a recurrence, which follows from the notion of giving low priority to the phrases that occur often in the dataset. The probability of a word appearing in a given text is its term frequency (TF).

$$tf(t,d) = log(1 + freq(t,d)) \qquad (1)$$

Inverse document frequency (IDF) is the opposite of the number of times a specific term showed up in the whole corpus. It catches how a specific term is document specific.

$$idf(t,d) = log(\frac{N}{count(d\epsilon D : t\epsilon d)}) \qquad (2)$$

TF-IDF is the multiplication of TF with IDF, suggesting how vital and relevant a word is in the document.

$$tfidf(t,d,D) = tf(t,d).idf(t,D) \qquad (3)$$

Similarly to Bow, the n-gram selection for TF-IDF in this study is (1,2).

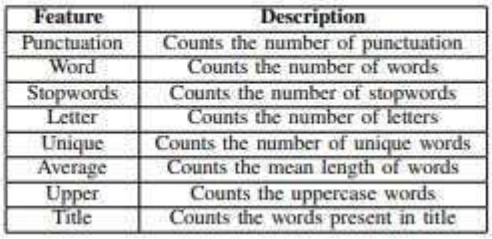
3: Word2Vec Although TF and TF-IDF are well- known vectorization algorithms utilized in a variety of natural language pre-processing applications [27], they ignore the semantic and syntactic similarities between words. For instance, while the terms wonderful and pleasant are almost synonymous, they are treated as separate entities in both TF and TF-IDF extraction methods. The model Word2Vec [18] is used to generate word embedding's. Using a variety of deep learning methods, we can recreate worded beddings from massive corpora [19]. Word2Vec takes a massive textual corpus as input and produces a vector space with often hundreds of dimensions. The underlying concept was to use the words' semantic meanings to arrange word vectors in vector space, with the ultimate goal of having words that have a similar sense in the dataset be located in close proximity to one another.

Manual Functions 4: An increasingly common method used to better the precision of a model is feature engineering. There were fifteen features in total used, such as the number of times each feature was found to be helpful, the condition column being label encoded with the label encoder function from the Scikit library, and the day, month, and year features being developed from the date column with the Date Time function in pandas.

In addition to the 8 characteristics indicated in Table I. C., the polarity of both the cleaned and unlearned reviews was extracted using the Text blob toolkit [20] and included as features. Separate Testing Train

Using Bow, TF-IDF, Word2Vec, and human- created features, we generated four datasets. Each of these four datasets was divided into a training and testing set. We guarantee that the train-test split of each of the four created datasets uses the same seed value by setting an equal random state when performing the split.

TABLE I
LIST OF FEATURES EXTRACTED MANUALLY FROM USER REVIEWS

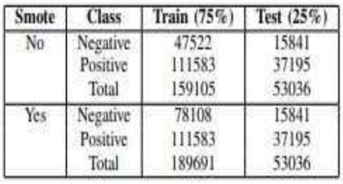


Feature	Description
Punctuation	Counts the number of punctuation
Word	Counts the number of words
Stopwords	Counts the number of stopwords
Letter	Counts the number of letters
Unique	Counts the number of unique words
Average	Counts the mean length of words
Upper	Counts the uppercase words
Title	Counts the words present in title

D. Smote After the Train Test split, only the training data undergone a synthetic minority over- sampling technique (Smote) [22] to prevent the class imbalance problem. Smote is an oversampling technique that synthesized new data from existing data. Smote generates the new minority class data by linear interpolation of randomly selected minority instance ’a’ in combination with its k nearest neighbor instance ’b’ in the feature space.

Table II shows the total distribution of data on final dataset i.e. after data cleaning. Fig. 6 shows the projection of non-smote and smote using t- distributed stochastic neighbor embedding (t-SNE)
[21] of 1000 rows on manual features data. It displays that there are more orange points in the non-smote t-SNE projection, which represents the majority class dominance. It also shows that there has been an increment in blue points after using smote that brings out the balance between a majority and minority class that curbs the predominance of the majority class.

TABLE II
DATASET DISTRIBUTION



Smote	Class	Train (75%)	Test (25%)
No	Negative	47522	15841
	Positive	111583	37195
	Total	159105	53036
Yes	Negative	78108	15841
	Positive	111583	37195
	Total	189691	53036

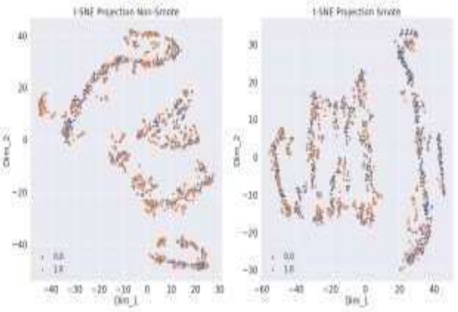


Fig. 6. t-SNE subplot before and after Smote using 1000 training samples

E. Detectors and Sorters

A classifier was constructed using several machine- learning classification techniques for making the prediction. Since the Bow, TF-IDF model is a highly sparse matrix and applying tree-based classifiers would be very time-consuming, we tested it with logistic regression, multinomial naive bayes, stochastic gradient descent, a linear support vector classifier, a perceptron, and a ridge classifier. Used Word2Vec and a handcrafted feature set with the Decision tree, Random Forest, LGBM, and Cat Boost classifiers. The large number of reviews in this dataset (approximately 210K) is a big computational challenge. Only machine learning classification techniques were chosen because of their speed and ability to shortentraining times.

Indicators, Metrics, and KPIs

Precision (Prec), recall (Rec), f1score (F1), accuracy (Acc.), and area under the curve (AUC)
[23] were the five measures used to evaluate the predictability of the emotion. Let the abbreviation be Tp for instances when the model accurately predicted a positive attitude. Tn = Instances where the model correctly predicted the negative class; Fp
= Instances where the model incorrectly predicted the positive class; Fn = Instances where the model incorrectly predicted the negative class. The following equations detail the measures of precision, recall, accuracy, and f1score:

$$Precision = \frac{T_p}{T_p + F_p} \quad (4)$$

$$Recall = \frac{T_p}{T_p + F_n} \quad (5)$$

$$Accuracy = \frac{T_p + T_n}{T_p + T_n + F_p + F_n} \quad (6)$$

$$F1score = 2.\frac{Precision.Recall}{Precision + Recall} \quad (7)$$

The area under the curve (Auc) is a measure of a classifier's ability to compare classes and is calculated by using the ROC curve. The Roc curve represents the trade-off between Tpr and Fpr atdifferent confidence levels.

System for recommending medications, letter G

After weighing the criteria, the four most promising forecasts were chosen and merged to form the final forecast. When the combined data was multiplied by the drug's normalized useful count, an overall score for that condition could be calculated. More positive ratings mean the medicine is more effective. A closer inspection of Fig. 7's distribution of relevant counts reveals the need for this uniformity; a comparison of the minimum and maximum values reveals a significant difference of roughly 1300.

More importantly, the standard deviation is a whopping 36. The rationale behind this is that a high useful count is a direct result of a large volume of pharmaceutical searches; more people will read the survey whether or not the drug theyare looking for is included. As a result, we standardized the usable count by circumstances when developing the recommender system.

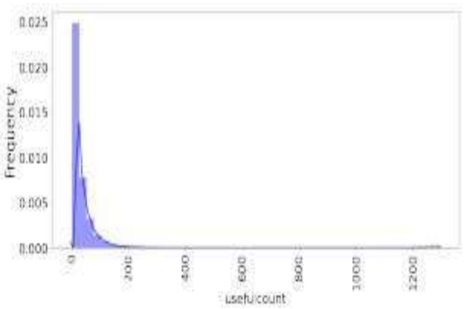


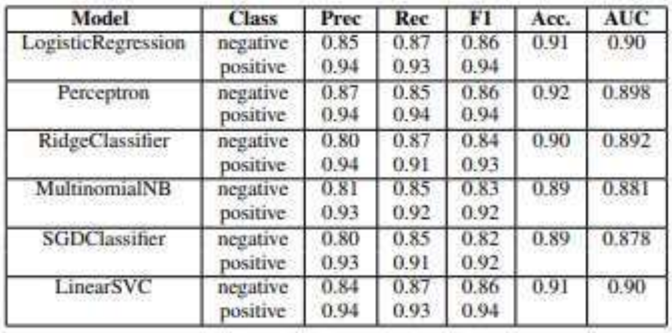
Fig. 7. Distribution of Useful Count

RESULTS

Based on the number of stars a user gave an evaluation, researchers sorted them into positive or negative categories. Ratings more than five are considered favorable, whereas ratings between one and five are considered bad. There were initially 111583 good ratings and 47522 negative ratings in the training data. To even things out, we used smote to make the minority class 70 percent as numerous as the dominant class. The new training set has 111583 positive courses and 78108 negative classes. For this task, we used 10 distinct ML algorithms for binary classification with four distinct text representation techniques (Bow, TF- IDF, Word2Vec, and Manual feature). Tables III, IV, V, and VI include results for five distinct measures.

Evaluation metrics for a bag of words vectorization approach are shown in Table III below. The superiority of perceptron over other categorization algorithms is plain to discern. Accuracy levels between 89% and 91% were achieved by all algorithms. With the help of logistic regression and linear SVC, we were able to get an AUC of 90%. Perceptron's AUC score was just 89.8%, despite its superior accuracy compared to logistic and Linear SVC.

TABLE III
BAG-OF-WORDS

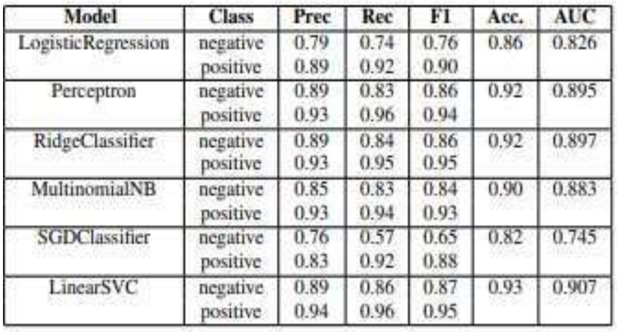


Model	Class	Prec	Rec	F1	Acc.	AUC
LogisticRegression	negative positive	0.85 0.94	0.87 0.93	0.86 0.94	0.91	0.90
Perceptron	negative positive	0.87 0.94	0.85 0.94	0.86 0.94	0.92	0.898
RidgeClassifier	negative positive	0.80 0.94	0.87 0.91	0.84 0.93	0.90	0.892
MultinomialNB	negative positive	0.81 0.93	0.85 0.92	0.83 0.92	0.89	0.881
SGDClassifier	negative positive	0.80 0.93	0.85 0.91	0.82 0.92	0.89	0.878
LinearSVC	negative positive	0.84 0.94	0.87 0.93	0.86 0.94	0.91	0.90

Metrics for the TF-IDF vectorization strategy are shown in Table IV. The accuracy attained by perceptron (91%) using a bag of words model is less impressive than the 93% performance obtained by the TF-IDF vectorization approach after being trained using LinearSVC. LinearSVC, perceptron, and ridge classifier all competed closely, with just a1% margin of error.

Nonetheless, LinearSVC was chosen as the top method because to its superior AUC score of 90.7%.

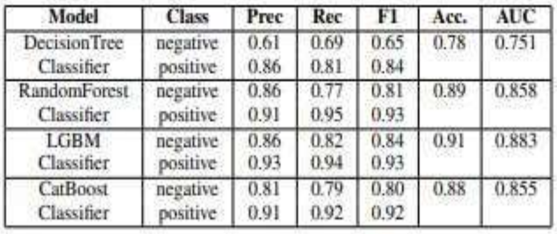
TABLE IV
TF-IDF



Model	Class	Prec	Rec	F1	Acc.	AUC
LogisticRegression	negative positive	0.79 0.89	0.74 0.92	0.76 0.90	0.86	0.826
Perceptron	negative positive	0.89 0.93	0.83 0.96	0.86 0.94	0.92	0.895
RidgeClassifier	negative positive	0.89 0.93	0.84 0.95	0.86 0.95	0.92	0.897
MultinomialNB	negative positive	0.85 0.93	0.83 0.94	0.84 0.93	0.90	0.883
SGDClassifier	negative positive	0.76 0.83	0.57 0.92	0.65 0.88	0.82	0.745
LinearSVC	negative positive	0.89 0.94	0.86 0.96	0.87 0.95	0.93	0.907

Table V compares the results of several Word2Vec categorization strategies based on a variety of performance indicators. In terms of precision, the LGBM model achieves 91%. Comparable outcomes may be obtained using either the random forest or catboost classifiers, whereas the decision tree classifier fares badly. The LGBM's superior AUC of 88.3% is simply shown by inspecting the area operating curve score.

TABLE V
WORD2VEC



Model	Class	Prec	Rec	F1	Acc.	AUC
DecisionTree Classifier	negative positive	0.61 0.86	0.69 0.81	0.65 0.84	0.78	0.751
RandomForest Classifier	negative positive	0.86 0.91	0.77 0.95	0.81 0.93	0.89	0.858
LGBM Classifier	negative positive	0.86 0.93	0.82 0.94	0.84 0.93	0.91	0.883
CatBoost Classifier	negative positive	0.81 0.91	0.79 0.92	0.80 0.92	0.88	0.855

Table VI displays the performance metrics of four different classification algorithms on manually created features on user reviews. Compared to all other text classification methods, the results are not pretty impressive. However, the random forest achieved a good accuracy score of 88%.

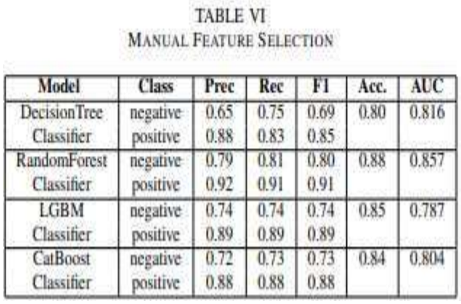
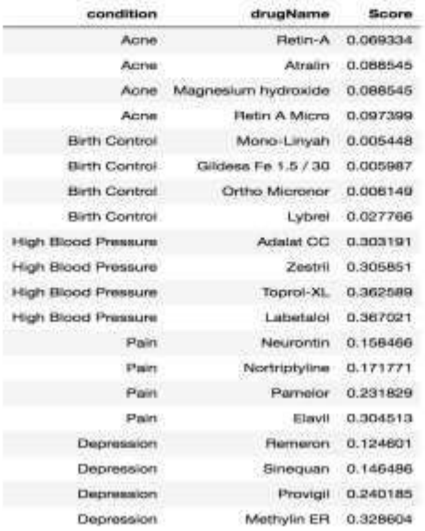


TABLE VI
MANUAL FEATURE SELECTION

Model	Class	Prec	Rec	F1	Acc.	AUC
DecisionTree Classifier	negative positive	0.65 0.88	0.75 0.83	0.69 0.85	0.80	0.816
RandomForest Classifier	negative positive	0.79 0.92	0.81 0.91	0.80 0.91	0.88	0.857
LGBM Classifier	negative positive	0.74 0.89	0.74 0.89	0.74 0.89	0.85	0.787
CatBoost Classifier	negative positive	0.72 0.88	0.73 0.88	0.73 0.88	0.84	0.804

Perceptron (Bow), LinearSVC (TF-IDF), LGBM (Word2Vec), and RandomForest (Manual Features) prediction results were pooled after model evaluation. The primary objective is to guarantee that all four models agree on how to classify the most highly suggested medications. If a drug's overall rating drops because one model got it incorrect, something must be wrong with that model. Each drug's total score was calculated by multiplying its combined forecasts by its normalized useful count. This was done to make sure there were sufficient reviewers for that medication. The final score is the mean score, calculated by dividing the total score by the number of medications used to treat each disease. Acne, contraception, high blood pressure, pain, and depression are the top five disorders for which our model suggests treatment (see Fig. 8).



condition	drugName	Score
Acne	Retin-A	0.069334
Acne	Atralin	0.088545
Acne	Magnesium hydroxide	0.088545
Acne	Retin A Micro	0.097399
Birth Control	Mono-Linyah	0.005448
Birth Control	Gildess Fe 1.5 / 30	0.005987
Birth Control	Ortho Micronor	0.006149
Birth Control	Lybrel	0.027766
High Blood Pressure	Adalat CC	0.303191
High Blood Pressure	Zestril	0.305851
High Blood Pressure	Toprol-XL	0.362589
High Blood Pressure	Labetalol	0.367021
Pain	Neurontin	0.158466
Pain	Nortriptyline	0.171771
Pain	Pamelor	0.231829
Pain	Elavil	0.304513
Depression	Remeron	0.124601
Depression	Sinequan	0.146486
Depression	Provigil	0.240185
Depression	Methylin ER	0.328604

Fig. 8. Recommendation of top four drugs on top five conditions

DISCUSSION

While all four approaches provide promising outcomes, there is not yet proof that the recommender architecture is suitable for widespread deployment. There's still room for development.

According to the findings, algorithms like Smote, Adasyn [24], SmoteTomek [25], etc., should be used to balance the training data such that the disparity between the positive and negative class metrics is small. In order to increase the model's precision, classification algorithms also need to have their hyperparameters optimized. We simply included each method's best anticipated outcome into the recommendation system. A good assembly of many expected findings is necessary for improved results and comprehension. To that aim, this study will focus only on the process of using sentiment analysis and classification to construct a recommender system.

CONCLUSION

Checking reviews before doing anything—whether it's going shopping, making an online purchase, or choosing a place to eat at—has become second nature. In light of this, we investigated the use of various machine learning classifiers, including Logistic Regression, Perceptron, Multinomial Naive Bayes, Ridge classifier, Stochastic gradient descent, LinearSVC, applied to Bow, TF-IDF, and classifiers like Decision Tree, Random Forest,

Lgbm, and Catboost, applied to Word2Vec and the Manual Features method, to perform sentiment analysis of drug reviews and develop a recommender system. We used five different metrics—accuracy, AUC, precision, recall, and f1score—to compare the models and found that the Linear SVC on TF-IDF achieved the highest accuracy (93 percent). However, Word2Vec's Decision tree classifier performed poorly, with just 78% accuracy. To create a recommender system, we multiplied the normalized usefulCount by the best-predicted emotion values from each approach, using Perceptron on Bow (91%), LinearSVC onTF-IDF (93%), LGBM on Word2Vec (91%), and
Random Forest on manual features (88%). The performance of the recommender system may be improved in the future by comparing alternativeoversampling methods, experimenting with othervalues of n-grams, and optimizing the algorithms.

REFERENCES

[1] Telemedicine, https://www.mohfw.gov.in/pdf/Telemedicine.pdf

[2] Wittich CM, Burkle CM, Lanier WL. Medication errors: an overview for clinicians. Mayo Clin Proc. 2014 Aug;89(8):1116-25.

[3] CHEN, M. R., & WANG, H. F. (2013). The reason and prevention of hospital medication errors. Practical Journal of Clinical Medicine, 4.

[4] Drug Review Dataset,https://archive.ics.uci.edu/ml/datasets/Drug% 2BReview%2BDataset%2B%2528Drugs.com%25 29#

[5] Fox, Susannah, and Maeve Duggan. ”Healthonline 2013. 2013.” URL:http://pewinternet.org/Reports/2013/Health-online.aspx

[6] Bartlett JG, Dowell SF, Mandell LA, File TM Jr, Musher DM, Fine MJ. Practice guidelines for the management of community-acquired pneumonia in adults. Infectious Diseases Society of America. Clin Infect Dis. 2000 Aug;31(2):347-82. doi: 10.1086/313954. Epub 2000 Sep 7. PMID: 10987697; PMCID: PMC7109923.

[7] Fox, Susannah & Duggan, Maeve. (2012). Health Online 2013. Pew Research Internet Project Report.

[8] T. N. Tekade and M. Emmanuel, ”Probabilistic aspect mining approach for interpretation and evaluation of drug reviews,” 2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES),Paralakhemundi, 2016, pp. 1471-1476, doi: 10.1109/SCOPES.2016.7955684.

[9] Doulaverakis, C., Nikolaidis, G., Kleontas, A. et al. GalenOWL: Ontology-based drug recommendations discovery. J Biomed Semant 3,14 (2012). https://doi.org/10.1186/2041-1480-3-14

[10] Leilei Sun, Chuanren Liu, Chonghui Guo, Hui Xiong, and Yanming Xie. 2016. Data-driven Automatic Treatment Regimen Development and Recommendation. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '16). Association for Computing Machinery, New York, NY, USA, 1865–1874. DOI:https://doi.org/10.1145/2939672.2939866

[11] V. Goel, A. K. Gupta and N. Kumar, ”Sentiment Analysis of Multilingual Twitter Data using Natural Language Processing,” 2018 8th International Conference on Communication Systems and Network Technologies (CSNT), Bhopal, India, 2018, pp. 208-212, doi: 10.1109/CSNT.2018.8820254.

[12] Shimada K, Takada H, Mitsuyama S, et al. Drug-recommendation system for patients with infectious diseases. AMIA Annu Symp Proc. 2005;2005:1112.

[13] Y. Bao and X. Jiang, ”An intelligent medicine recommender system framework,” 2016 IEEE 11th Conference on Industrial Electronics and Applications (ICIEA), Hefei, 2016, pp. 1383-1388, doi: 10.1109/ICIEA.2016.7603801.

[14] Zhang, Yin & Zhang, Dafang & Hassan, Mohammad & Alamri, Atif & Peng, Limei. (2014). CADRE: Cloud-Assisted Drug REcommendation Service for Online Pharmacies. Mobile Networks and Applications. 20. 348-355. 10.1007/s11036- 014-0537-4.

[15] J. Li, H. Xu, X. He, J. Deng and X. Sun, ”Tweet modeling with LSTM recurrent neural networks for hashtag recommendation,” 2016 International Joint Conference on Neural Networks

(IJCNN), Vancouver, BC, 2016, pp. 1570-1577,doi: 10.1109/IJCNN.2016.7727385.

[16] Zhang, Yin & Jin, Rong & Zhou, Zhi-Hua. (2010). Understanding bag-of-words model: A statistical framework. International Journal of Machine Learning and Cybernetics. 1. 43-52.10.1007/s13042-010-0001- 0.

[17] J. Ramos et al., “Using tf-idf to determine word relevance in document queries,” in Proceedings of the first instructional conference on machinelearning, vol. 242, pp. 133–142, Piscataway, NJ, 2003.

[18] Yoav Goldberg and Omer Levy. word2vec Explained: deriving Mikolov et al.’s negative- sampling word-embedding method, 2014; arXiv:1402.3722.

[19] Danushka Bollegala, Takanori Maehara and Kenichi Kawarabayashi. Unsupervised Cross- Domain Word Representation Learning, 2015;arXiv:1505.07184.

[20] Textblob,
https://textblob.readthedocs.io/en/dev/.
Personality Recognition,” in IEEE Access, vol. 7, pp. 129678-129689, 2019, doi: 10.1109/ACCESS.2019.2940061.

van der Maaten, Laurens & Hinton, Geoffrey. (2008). Viualizing data using t-SNE. Journal of Machine Learning Research. 9. 2579-2605.

N. V. Chawla, K. W. Bowyer, L. O. Hall and
W. P. Kegelmeyer. SMOTE: Synthetic Minority Over-sampling Technique, 2011, Journal Of Artificial Intelligence Research, Volume 16, pages 321-357, 2002; arXiv:1106.1813. DOI: 10.1613/jair.953.

Powers, David & Ailab,. (2011). Evaluation: From precision, recall and F-measure to ROC, informedness, markedness & correlation. J. Mach. Learn. Technol. 2. 2229-3981. 10.9735/2229-3981

Haibo He, Yang Bai, E. A. Garcia and Shutao Li, ”ADASYN: Adaptive synthetic sampling approach for imbalanced learning,” 2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence), Hong Kong, 2008, pp. 1322-1328,doi: 10.1109/IJCNN.2008.4633969.

Z. Wang, C. Wu, K. Zheng, X. Niu and X. Wang, ”SMOTETomekBased Resampling for