Illinois Institute of Technology

Final Project Report

“Disease Prediction and Drug Recommendation”
CS 584 - Machine Learning

Group number : 21
Geeta Sudhakar Matkar (A20516262)
Pradaap Shiva Kumar Shobha (A20512400)

INTRODUCTION

All across the world, many people struggle with their health and medical diagnoses. Patients can readily learn about the illness they have and the medication that can help treat it using the approaches utilized in this study by simply entering the symptoms they are experiencing.

In this project, we suggest medications to individuals based on their conditions and reviews. Four distinct models are used to predict the diseases. The Vader tool is used to assess the reviews. Finally, weighted average methods are used to recommend the drugs. Each model and strategy utilized in this project are described in detail.

- **Summary**

Disease prediction and drug recommendation have always been a crucial task in healthcare. A timely and accurate diagnosis of diseases is critical for effective treatment and management. In recent years, there has been an increasing interest in using machine learning techniques for disease prediction and drug recommendation. These techniques use patient data such as symptoms, medical history, and genetic information to predict the likelihood of a disease and recommend appropriate drugs.

The main objective of this research is to develop a disease prediction and drug recommendation system using machine learning techniques. The system will take patient data as input and predict the likelihood of various diseases, as well as recommend appropriate drugs for treatment. To achieve this, several machine learning algorithms will be trained and evaluated using real-world patient data. The performance of these algorithms will be compared to select the best model for disease prediction and drug recommendation.

The proposed system has the potential to significantly improve the accuracy and efficiency of disease diagnosis and drug recommendation, ultimately leading to better patient outcomes and reduced healthcare costs.

The objective of this project is to assist individuals who are at risk of catching specific diseases and help them take proactive steps to treat or prevent the disease before it becomes a significant health concern. Machine learning algorithms can assist in locating undiscovered correlations between various symptoms that might lead to the emergence of a specific disease by evaluating enormous volumes of data.

In order to accurately anticipate the risk of a disease occurring, the project calls for the gathering, analysis, and development of vast volumes of data. Several diseases, such as jaundice, diabetes, tuberculosis, and many others, can benefit from the effort.

- **Previous work**

1. Kamaraj, K. Gomathi, and Priyaa D. Shanmuga (2016) presented the results of using Decision Tree and Naive Bayes models to predict three diseases: Heart Disease, Diabetes, and Breast Cancer. [1]

2. Youjun Bao and Xiaohong Jiang (2016) designed and implemented a universal medicine recommender system that uses data mining technologies. The system includes modules for database management, data preparation, recommendation models, model evaluation, and data visualization.[2]

3. H. Wang, Q. Gu, J. Wei, Z. Cao, and Q. Liu's (2015) paper, they developed a framework to determine novel drug indications and side effects in one integrated system. Their method complements medical genetics-based drug

repositioning and reduces false positives, resulting in a ranked list of new candidate indications and/or side effects with varying confidence levels. [3]

4. Yin Zhang, Daqiang Zhang, Mohammad Mehedi Hassan, Atif Alamri, and Limei Peng (2014) proposed a novel cloud-assisted drug recommendation (CADRE) system that recommends top-N related medicines based on symptoms. CADRE first clusters drugs based on functional description information and then designs a personalized drug recommendation using user collaborative filtering.

5. V. M.A. Nishara Banu and B. Gomathy (2013) used K-Mean clustering on a medical dataset to identify relevant data, which was then fed to the MAFIA algorithm to generate rules and frequent patterns for the C4.5 (Decision Tree) model to classify patterns and predict heart diseases.

PROBLEM DESCRIPTION

Nowadays, a wide range of illnesses affect people due to their lifestyle and environment. Thus, it is critical to make an early diagnosis of disorders. In a crisis or emergency situation where they don't seem to have an immediate option, it would be simpler for patients to interpret their test results, assisting them in determining if they have already contracted any diseases or not. In order to address this issue, we have developed a model that recognizes a disease from a patient's symptoms and suggests appropriate medications. Drug recommendations are made based on reviews from prior patients, and disease prediction is done based on the patient's symptoms.

In this project, we have developed various machine learning models that, using the data we have obtained, can determine the disease a person is suffering from. Several symptom attributes connected to one or more diseases are included in the Symptoms-Disease dataset that is provided. The medical condition someone is dealing with is determined by different combinations of these symptom characteristics. This dataset contains 133 symptom features, such as itching, breathlessness, cold, and cough, which work together to identify health conditions like allergy, malaria, typhoid, and dengue, among others.

- **Dataset details**

For the accurate recommendation of drugs, we first predict the diseases based on symptoms and then recommend the drugs based on the ratings. For this purpose, we have tried to gather information from two main datasets.

1. Disease - Symptoms dataset
    a. This dataset is used to take symptoms as input and predict the disease as an output.
    b. This dataset was obtained from Kaggle. It contains 133 columns, out of which 132 are symptoms. It has a total of 4919 observations.

2. Drug review dataset
    a. This dataset is used in order to take the predicted disease as input and recommend appropriate drugs based on reviews and ratings (sentiment analysis).
    b. The dataset used in this study was obtained from the University of California, Irvine.
    c. This dataset has 318884 rows and 6 columns.
    d. Column features are: drug name, medical condition, review, rating, date, and useful count.

```
DF = pd.read_csv("/Users/geetamatkar/Documents/Part1_NewData.csv", index_col='prognosis')
DF.sample(n = 5)
```

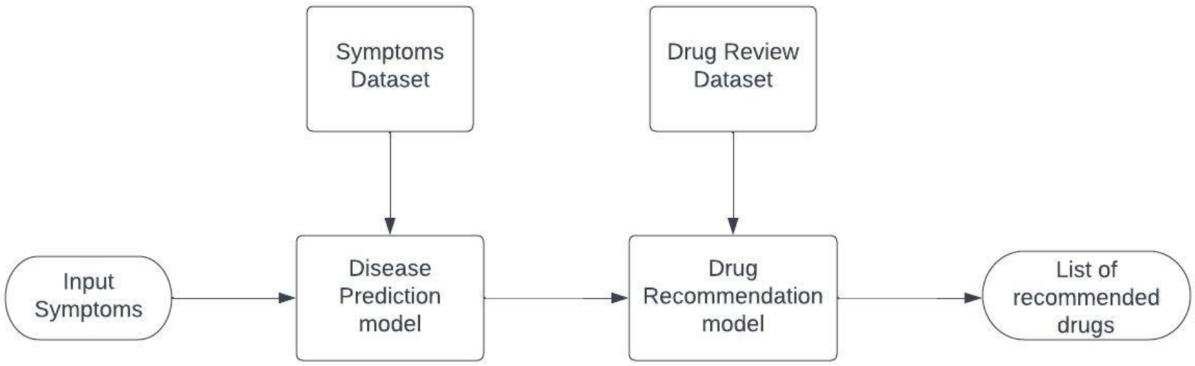
prognosis	itching	skin_rash	nodal_skin_eruptions	continuous_sneezing	shivering	chills	joint_pain	stomach_pain	acidity	ulcers_on_tongue	...	pus_filled
Acne	0	0	0	0	0	0	0	0	0	0	...	
Heart attack	0	0	0	0	0	0	0	0	0	0	...	
Migraine	0	0	0	0	0	0	0	0	1	0	...	
Psoriasis	0	1	0	0	0	0	1	0	0	0	...	
Osteoarthristis	0	0	0	0	0	0	1	0	0	0	...	

5 rows × 132 columns

	drugName	condition	review	rating	date	usefulCount
22516	Pristiq	Depression	"Went to the doctor complaining of anxiety. W...	10	January 23, 2014	121
33083	Saxenda	Obesity	"I started Saxenda around the 3rd week of July...	10	August 15, 2017	4
13214	Nitrofurantoin	Urinary Tract Infection	"I would not choose this drug again. SEVERE ab...	3	April 5, 2016	18
36445	Phentermine / topiramate	Weight Loss	"You can go to qsymia website to get a discoun...	8	March 1, 2017	15
38356	Phillips' Milk of Magnesia	Constipation	"Due to pregnancy I experienced constipation, ...	9	August 24, 2013	66

- **Methodologies used**

The main goal of our project is to recommend a drug to a patient based on the symptoms she/he has. In accordance with our objective to implement a drug recommender system there are two main subcategories which are to be addressed i.e., a disease prediction model and a recommendation model



- **Disease prediction**

For accurately predicting the diseases, we have experimented with multiple approaches in our project. We have implemented 4 different approaches, and all of them focus on a different way of prediction. In the end, all these approaches are trained on 4 classifiers and the accuracies and prediction results are compared for choosing the best possible way. The approaches are explained in detail below:

- **Random Forest**

Random Forests are an ensemble learning method used for classification and regression tasks in machine learning. This algorithm builds multiple decision trees and combines them to improve overall prediction accuracy. Random forests build multiple decision trees on different subsets of the training data using random subsets of features at each split. The final prediction is made by taking the majority vote (in classification) or the average (in regression) of the predictions of the individual trees. Randomization of both data and features helps to reduce overfitting and improve the generalization performance of the model.

- **Decision Tree**

Decision Trees are a machine learning algorithm commonly used for classification and regression problems. This algorithm partitions data into subsets based on a selected attribute that maximizes the separation of the target variable.

This creates a tree-like model where internal nodes represent tests on attributes, branches represent outcomes, and leaf nodes represent predicted values or classes. To construct the tree, the algorithm recursively selects the best feature to split the data on, stopping when a criterion is met, such as a minimum number of instances per leaf node or maximum tree depth. Commonly used criteria for selecting the best feature include information gain, Gini impurity, and chi-squared tests.

- **Naive Bayes**

Naive Bayes is a machine learning classification algorithm based on Bayes' theorem. It makes a simplifying assumption that input features are conditionally independent of each other given the class label. The algorithm predicts the class label of an input data point by calculating the probability of the input data point belonging to each class label and choosing the label with the highest probability. Naive Bayes calculates the probability of a data point belonging to a particular class label using Bayes' theorem, which states that the probability of a hypothesis (class label) given the observed evidence (input data point) is proportional to the probability of the evidence given the hypothesis, multiplied by the prior probability of the hypothesis.

- **K Nearest Neighbor**

K-Nearest Neighbors (KNN) is a machine learning algorithm for classification and regression tasks. KNN is an instance-based learning algorithm that memorizes the training data instances and predicts the class or value of new instances based on their similarity with the nearest K training instances. The similarity between two instances is calculated using a distance metric, such as Euclidean distance or Manhattan distance. The KNN algorithm finds the K training instances closest to the new data point in terms of the distance metric and assigns the majority class label (in classification) or the average value (in regression) of these K instances to the new data point.

Drug recommendation

Upon obtaining the most probable diseases the next task is to map the list of drugs that
can be prescribed for this particular disease. For recommending drugs, we have used the VADER tool. It is a simple rule based model for general sentiment analysis.

- **<u>VADER Tool:</u>**

The VADER tool is a powerful sentiment analysis tool that is specifically designed to analyze sentiments expressed in social media. Its full form is Valence Aware Dictionary and sentiment Reasoner. Unlike other models, it doesn't require any training data as it already uses a combination of qualitative and quantitative methods to produce and then empirically validate a gold-standard sentiment lexicon to evaluate the sentiment of the sentence.

One of the biggest advantages of the VADER tool is that it works exceptionally well on social media-style texts, which may include emoticons and punctuations. It not only tells us about the polarity (positive or negative) of the review, but also provides information about how positive or negative the review is. The output generated by the VADER tool has four components, namely Positive, Negative, Neutral, and Compound. Positive, Neutral, and Negative specify how many parts of the sentence have a positive, neutral, and negative tone, respectively. The Compound component specifies the overall sentiment of the sentence.

Below thresholds help us to interpret results of compound:
1. Positive sentiment : compound score >= 0.05
2. Neutral sentiment : -0.05 < compound score < 0.05
3. Negative sentiment : compound score <= -0.05

	Review	Positive	Negative	Neutral	Compound
0	"Even though I have read that atenolol may rai...	0.080	0.846	0.074	0.0408
1	"I took this medication for Vertigo,, it made ...	0.000	0.757	0.243	-0.7521
2	"I use meclizine/antivert every day of my life...	0.030	0.725	0.246	-0.9775
3	"My doctor put me on this for vertigo. I had d...	0.000	0.923	0.077	-0.3291
4	"I have taken Antivert for years. For some re...	0.091	0.909	0.000	0.4767
...	...	...	...	...	...
45549	"I got this medication through my IV while in ...	0.082	0.874	0.044	0.3110
45550	"The medicine works but the procedure is intra...	0.000	0.834	0.166	-0.7343
45551	"I am a Intern Doctor (Cardiologist) this is t...	0.000	1.000	0.000	0.0000
45552	"I was diagnosed with a bacterial sinus infect...	0.063	0.889	0.048	0.3400
45553	"Brand name Zithromax works better than the lo...	0.116	0.737	0.147	0.0377

	condition	drugName	review	Review_Sentiment	rating	usefulCount	date
0	Vertig	Tenormin	"Even though I have read that atenolol may rai...	Neutral	9	35	April 7, 2011
1	Vertig	Antivert	"I took this medication for Vertigo,, it made ...	Negative	1	29	August 6, 2015
2	Vertig	Antivert	"I use meclizine/antivert every day of my life...	Negative	10	49	January 4, 2016
3	Vertig	Antivert	"My doctor put me on this for vertigo. I had d...	Negative	8	69	August 13, 2011
4	Vertig	Antivert	"I have taken Antivert for years. For some re...	Positive	10	42	July 29, 2011

To get the best possible drug recommendation for a predicted disease, we calculated a weighted average using ratings and useful count numbers. Hence we get a weighted average rating which gives more importance to the useful count.

- For recommending drugs to the user, we have only considered drugs that have a positive sentiment in their reviews. Hence, filtering out unwanted drugs.
- Also, we have taken a total of all the useful counts present for a particular drug to sort the drug that has to be recommended with the highest useful count first and then the highest average rating.

	drugName	Rating_Wavg
0	Abatacept	7.000000
1	Abilify	7.917470
2	Absorica	7.230769
3	Acanya	8.482828
4	Accutane	8.909503

RESULTS

1. Disease Prediction

For disease prediction, we are using four machine learning models: Random Forest, Decision Tree, Naive Bayes and K Nearest Neighbor. Accuracy of these models are written below:
Accuracy of Random Forest classifier : 95.32%
Accuracy of Decision Tree classifier : 91.05%
Accuracy of Naive Bayes classifier :
Accuracy of K Nearest Neighbor classifier :

```
predicted_disease = (mod.predict(arr))
predicted_diseaseDT = (classifierDT.predict(arr))
predicted_diseaseNB = (gaussian_nb.predict(arr))
predicted_diseaseKNN = (knn_pipeline.predict(arr))

print("The disease predicted based on given symptoms is by Random Forest: " + predicted_disease[0])
print("The disease predicted based on given symptoms is by Decision Tree: " + predicted_diseaseDT[0])
print("The disease predicted based on given symptoms is by Naive Bayes: " + predicted_diseaseNB[0])
print("The disease predicted based on given symptoms is by KNN: " + predicted_diseaseKNN[0])
```

```
The disease predicted based on given symptoms is by Random Forest: GERD
The disease predicted based on given symptoms is by Decision Tree: Psoriasis
The disease predicted based on given symptoms is by Naive Bayes: GERD
The disease predicted based on given symptoms is by KNN: GERD
```

2. Drug Recommendation

There are multiple drugs for a single disease. Hence, by using sentiment analysis, we were able to filter out the negative and neutral reviews, leaving us with only positive ones.
The results obtained from this approach gave us the correct and highly rated drug for the predicted disease. The disease that we predicted is depression based on the given, and we are recommending the top 3 drugs from our dataset as an output. It can be seen below:

```
recommended_drug = pd.DataFrame(groupedByDisease.get_group((predicted_disease[0])).nlargest(5, ['Rating_Wavg', 'usefulC
recommended_drug
```

	condition	drugName	Rating_Wavg	usefulCount
473	GERD	Calcium carbonate / simethicone	10.000000	9
485	GERD	Nizatidine	10.000000	5
503	GERD	Zegerid OTC	10.000000	5
471	GERD	Aciphex	9.564054	256
487	GERD	Omeprazole / sodium bicarbonate	9.336134	79

```
print("Following are the drugs recommended for ", predicted_disease[0],  " :\n ", recommended_drug["drugName"].unique()
```

```
Following are the drugs recommended for  GERD  :
  ['Calcium carbonate / simethicone' 'Nizatidine' 'Zegerid OTC' 'Aciphex'
 'Omeprazole / sodium bicarbonate']
```

CONCLUSION AND FUTURE WORK

We have successfully built a drug recommendation system that predicts diseases and recommends drugs along with possible side effects based on user symptom input. We experimented with different approaches for the task “Disease Prediction." Each of the three models showed good accuracy, contributing to the overall reliability of the drug recommendation model.

The disease prediction and drug recommendation project has great potential for future growth and expansion. Here are some possible future scope options for this project:

Building a User Interface: The project could be enhanced by developing a user-friendly interface that allows users to easily input their symptoms and receive accurate disease predictions and drug recommendations. The UI could also provide additional information about each disease, such as causes, symptoms, and treatment options.

Adding More Diseases: Currently, the model may only be trained on a limited set of diseases, but the potential for the project can be greatly increased by expanding the number of diseases it can predict. This can be done by collecting more data and training the model to recognize a wider range of symptoms and patterns.

Improving Model Accuracy: As with any machine learning model, the accuracy of the disease prediction and drug recommendation models can always be improved. To achieve this, the project can use techniques like feature engineering, hyperparameter tuning, and ensemble learning to fine-tune the model and achieve higher accuracy.

Integration with Electronic Health Records: The project could be integrated with electronic health record systems to help doctors and healthcare providers make more informed decisions about patient care. This could include real-time alerts for potential disease outbreaks or personalized drug recommendations based on patient data.

Overall, the future scope of the disease prediction and drug recommendation project is vast and can be expanded in multiple directions. By leveraging new technologies and techniques, the project can greatly benefit patients, doctors, and healthcare providers in the years to come.