





# Vaidhya4u

## DOCTOR APPOINTMENT BOOKING SYSTEM

NLP Project



**By** - Vishwak Sena Reddy (BTECH/10166/21) & Gaurang Dixit (BTECH/10432/21)



# Introduction of the Project

**Objective:** Create a user-friendly system for patients to book doctor appointments based on symptoms.

## **Overview:**

- Utilized NLP techniques to classify patient symptoms and book appropriate appointments.
- Built extensive pre-processing techniques.
- Leveraged Spacy for text processing and Scikit-learn for text classification.
- Included a user-friendly GUI using Tkinter to facilitate user interaction.

# What is Text Classification?

Text Classification is the task of assigning a label or class to a given text. The goal of text classification is to automatically classify the text documents into one or more defined categories.

## Types of Text Classification:

1. **Binary Class Text Classification:** Here we have only two classes (true or false, positive or negative etc..) , this type of classification is mostly used in Sentiment Analysis, Spam Detection etc..
2. **Multiclass Text Classification:** Here we have multiple labels or classes(positive, negative, or neutral, Sports,entertainment, or educational etc..) , this type of classification is used in News Paper article classification, customer review analysis etc..

Text Classification is an example of supervised machine learning task since a labelled dataset containing text documents and their labels is used for train a classifier.

**An end-to-end text classification pipeline is composed of three main components:**

1. **Dataset Preparation:** The first step is the Dataset Preparation step which includes the process of loading a dataset and performing basic pre-processing. The dataset is then splitted into train and validation sets.
2. **Feature Engineering:** The next step is the Feature Engineering in which the raw dataset is transformed into flat features which can be used in a machine learning model. This step also includes the process of creating new features from the existing data.
3. **Model Training:** The final step is the Model Building step in which a machine learning model is trained on a labelled dataset.
4. **Performance evaluation of the model & improvements:** This includes metrics such as accuracy, recall, F1 score and precision. Improvements include using different models or changing model parameters.

# How Text classification is helping our project:

In our project we have a dataset consisting of user text containing symptoms and disease label related to that symptom .We want to find out disease related to the symptoms given by the user so that we can use that disease and further proceed through booking an appointment of a doctor who has specialization in that disease field.Here Text classification serves it's best purpose, it helps in identifying the label or class of set of symptoms given by the user which can be used in appointment booking.

# Different Libraries and technologies used in our project

## I. CLASSIFICATION MODULE:

- A. **SPACY LIBRARY** - SpaCy is an open-source library designed for advanced Natural Language Processing (NLP). It provides an easy-to-use interface for efficient text processing, featuring pre-trained models, linguistic annotations, tokenization, and a range of powerful features for text analysis and information extraction.
- B. **SKLEARN** - for feature extraction, classification and evaluation
- C. **NUMPY and PANDAS LIBRARY** - for data manipulation and in feature extraction representation
- D. **Pyspellchecker** - for spelling correction
- E. **PICKLE** - python module for reading and writing to binary files called .pkl files (used for storing our data)

## II. Main UI Module:

- A. **Tkinter** - GUI module of python
- B. **Random & Pickle** - Misc python module

# Different functions used in this project

## I. Classification Module:

- A. **socialmediacheck(word)** - Used to find and replace popular short forms with the expanded form
- B. **spellchecker(word)** - Used to find and correct incorrect words in the user text (except for names, short forms, time, etc.)
- C. **preprocess(text)** - The function which is called to preprocess the user text and it performs tokenization, short forms substitution and spelling correction and returns the final user text
- D. **modelTrain()** - Used for preprocessing the dataset, performing word embedding and training the model
- E. **classification(user\_text)** - Used to find the class of the final user text.
- F. **convert\_time(time\_str)** - converts time in different formats to HHMM hours format
- G. **time\_finder(user\_text\_final)** - finds if any time exists or not in the user text and if exists and is in correct format , then it return the time in the correct format

## II. Main UI Module:

- A. **classify\_text(model,no\_to\_label,vectorizer)** - classifies text
- B. **confirmation\_window(time,spec,name='Anonymous')** - menu for user confirmation
- C. **appointment\_booking(time,spec,name)** - caller function for booking appointment (calls booking function)
- D. **booking(time,spec)** - used to book an appointment using the data stored in files and if slot is available then appending the files or else returning that booking is not possible at the specified time
- E. **display\_receipt(time,name,spec,doc\_id,doc\_name)** - displays receipt for the booked appointment

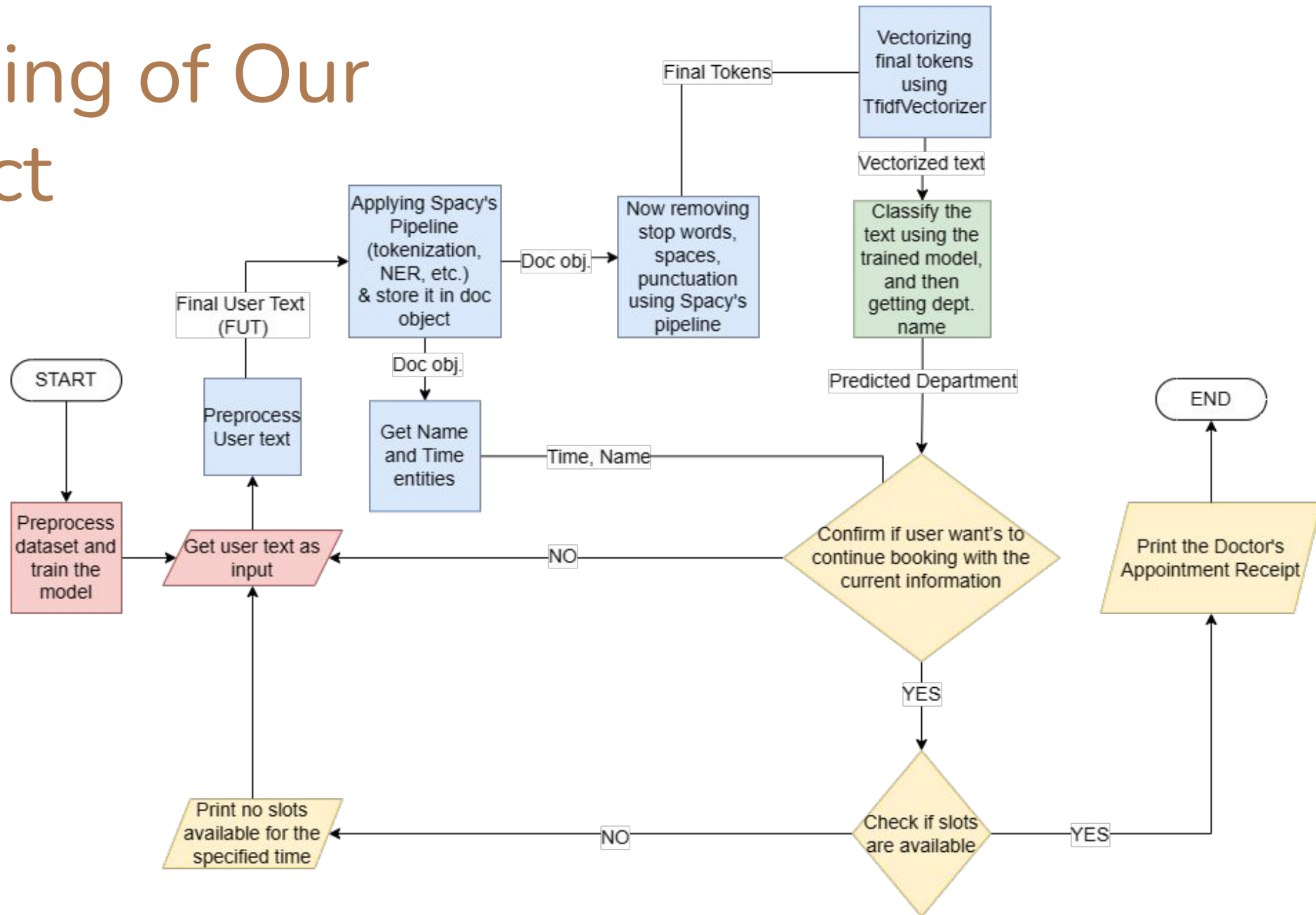
# Dataset used in this project

1. Dataset used is in .csv format and contains 2 columns , one for disease label of the text and the other the text for which the label is given.
2. No of unique labels in the dataset : 24
3. No of unique text sentences : 1200
4. No of sentences for each label : 50

[illegible]



# Working of Our project



# Challenges Faced

- First and foremost challenge we faced was finding the right dataset. We searched about 1000+ datasets and then found a relevant dataset.
- While performing spelling correction in preprocessing of user input, the name of the person was getting altered, and even using entities of Spacy could not solve this because we were sending one word at a time for spelling correction but our name can be of multiple words. So we replaced the name entity with a single character and modified our code to make it function properly.
- We also faced issue while correcting the spelling, preprocessing the text. We initially used 'autocorrect' library but it turned out to be very poor in performance, so we replaced it with 'pyspellchecker'.
- We were also getting wrong output sometimes but preprocessing the dataset text properly made the output accurate.
- Making UI and integrating it with the classification output was also a very tedious task.

# Future Scope

- Integration of voice as an input.
- Another major feature would be classifying text/voice as emergency or not. (Subject to further research)
- One such improvement can be done in the time function which can handle multiple input time formats.
- We can also add one more section while booking which gives basic medications to a particular disease which will improve one's condition if they are unable to get an appointment.

# Applications of this project

- This project can become base for many online doctor appointment booking systems where a particular doctor is appointed based on the class or label of the symptoms mentioned by the user.
- This will also help many people who are unable to take appointment of a particular doctor who can cure their disease.
- By simply mentioning one's symptoms he/she can book their appointment to a particular doctor, at a particular time.

# Acknowledgement

- Firstly, Shruti ma'am for helping in understanding the core concepts of NLP.
- Secondly, Kaggle for helping us find the right dataset for our project.
- Last but not least chatGPT for helping us prepare a proper plan, helping us understand functions and libraries such as Spacy, sklearn, Tk and for helping us in GUI and for solving some bugs.



THANK YOU