Social Media Web Application for prevention of mental disorders in the posts using Natural Language Processing

Zainab Saeed BSEF19M516
Ehtisham Sadiq BSEF19M521
Tayyaba Nasir BSEF19M525
Ayesha Rashid BSEF19M531

Outline

Problem Statement

Data Acquisition

Data/Text Preprocessing

Data Visualization

Feature Engineering

Model Building

Model Evaluation

NLP Pipeline

Application Development

Deployment

Problem Statement

Mental health is a serious issue of the modern-day world. According to the World Health Organization’s 2021 report more 20% of the world's children and adult population suffers from an episode of a mental disorder in their life. The problem grows bigger with the fact that as much as 35–50% of those affected go undiagnosed and receive no treatment for their illness.

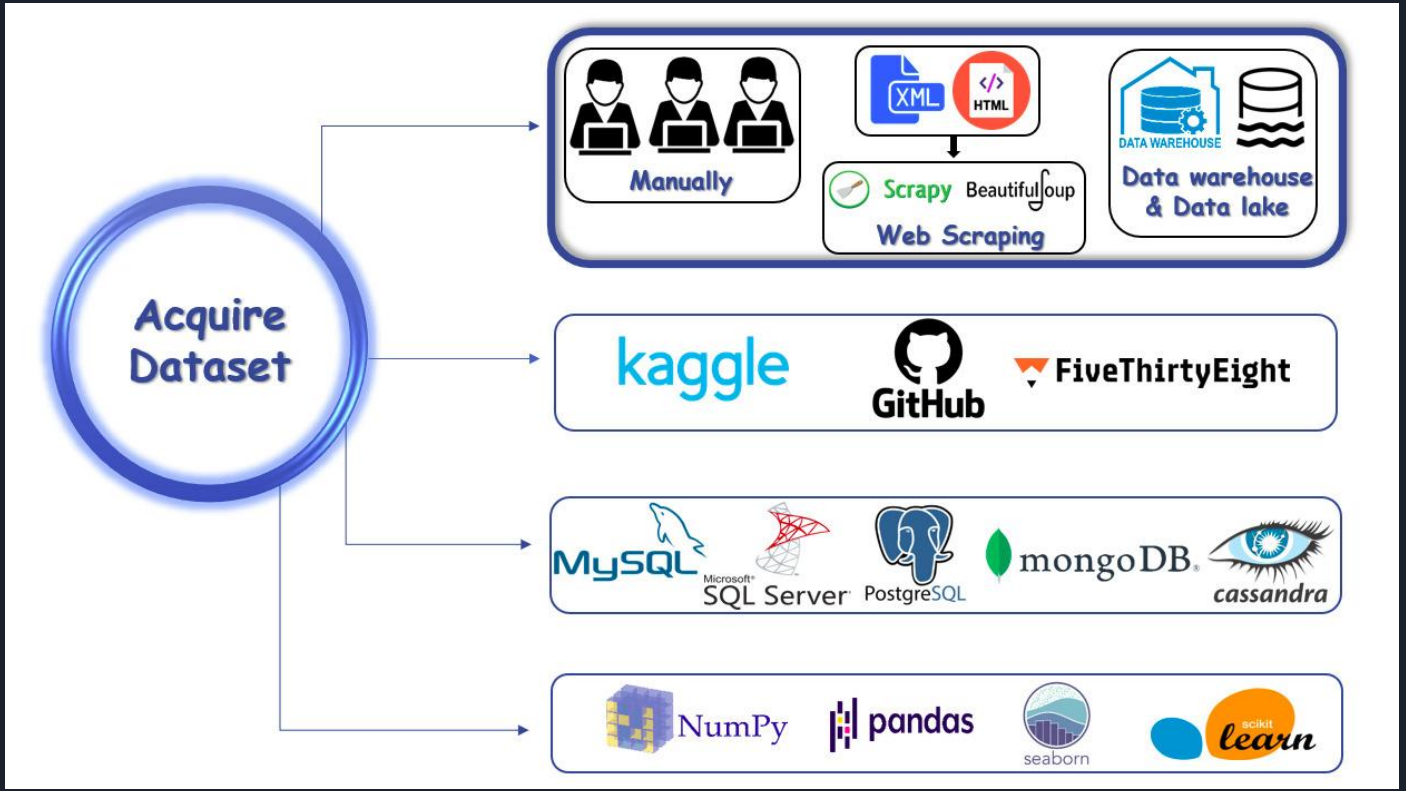
we want to build a system/application where a user can view the posts/text/images according to his/her interest , but these posts/text/images are prevented from mental disorders like depression, tension, anxiety etc. Here we use different techniques and tools of Machine Learning, Natural Language Processing and Deep learning to analyze the text/images of social media posts to find the mental health disorders. After filtering out the text/images, we will recommend/display those posts to user which will not be harmful for her/his mental health.

Data Acquisition

Data acquisition meaning is to collect data from relevant sources before it can be stored, cleaned, preprocessed, and used for further mechanisms. It is the process of retrieving relevant business information, transforming the data into the required business form, and loading it into the designated system.

We can acquire our dataset through many resources like online data platforms(Kaggle etc) , data warehouses, built in data of numpy pandas, web scraping etc.

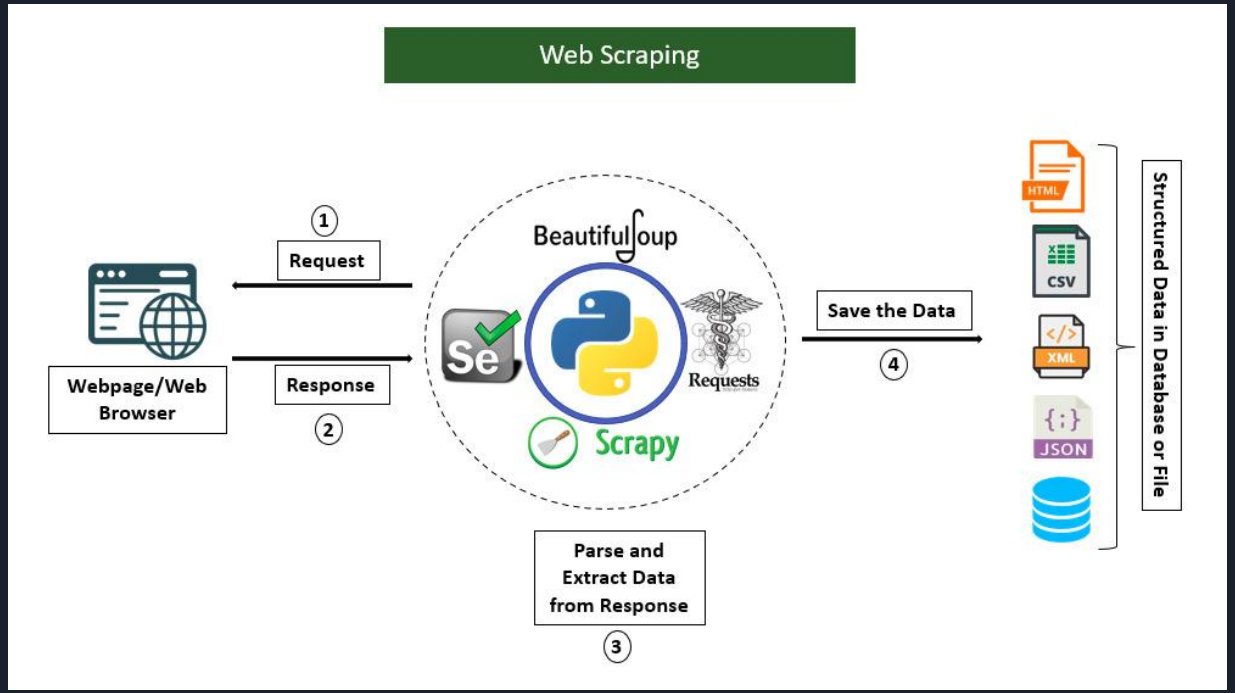
Data Acquisition(Cont.)



Data Acquisition(Cont.)

Web scraping is an automatic method to obtain large amounts of data from websites.

- Scapy ,BeautifulSoup and Selenium are some tools through which request can be sent and in response data will be sent in json form.



Data Acquisition(Cont.)

- **Selenium** is an open-source tool that automates web browsers. It provides a single interface that lets you write test scripts in programming languages. Selenium is a automated testing framework used to validate web applications across different browsers and platforms.
- **Tweepy** is an open source Python package that gives you a very convenient way to access the Twitter API with Python.

- The **Instaloader module** is a Python package having great functionalities to scrap instagram, it's functions can be used as command-line utility. The Instaloader key is used to download the Posts of public/private account, stories , IGTV, Comments on post, Profile information and Story highlights.
- **Facebook scraper** is a tool to extract publically available data from facebook

Data/Text Pre-Processing

Data Preprocessing is a technique that is used to convert the raw data into a clean data set. In other words, whenever the data is gathered from different sources it is collected in raw format which is not feasible for the analysis.

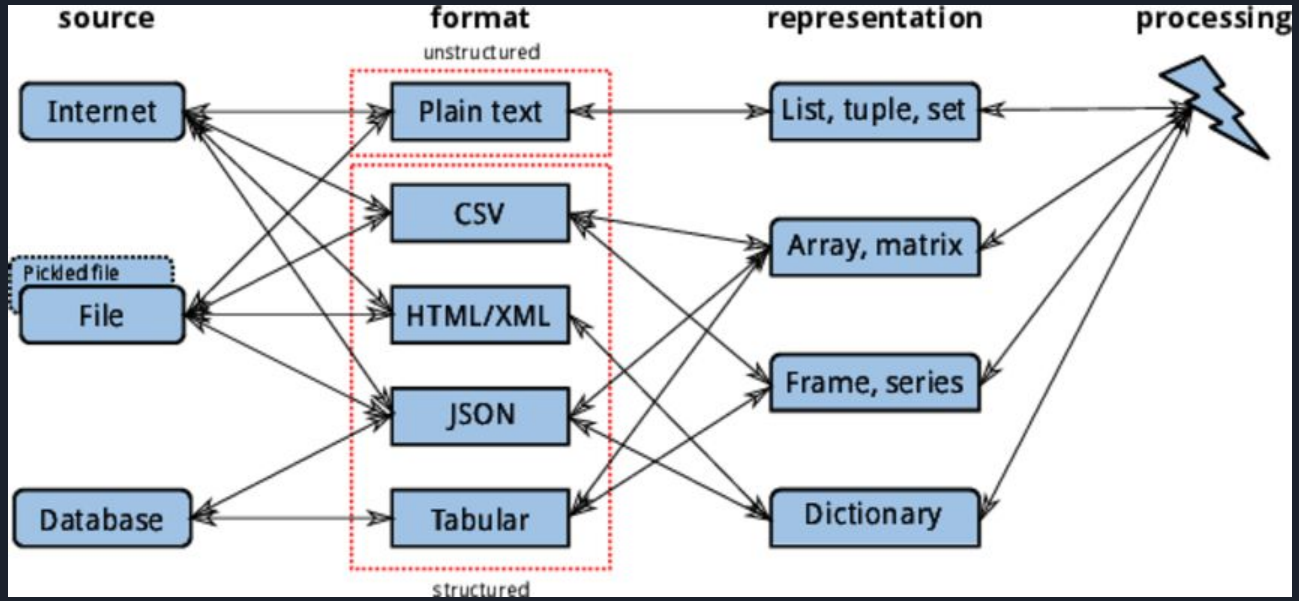
There are four major steps of data preprocessing.

- Data Quality Management
- Data Cleaning
- Data Transformation
- Data Reduction

Text Preprocessing(Cont.)

- In NLP, text preprocessing is **the practice of cleaning and preparing text data**.
- Convert all text to single case(Upper or lower case) and Remove all punctuation marks by using built in libraries like **word-cloud, text-clean** etc.
- In Python, **tokenization** basically refers to splitting up a larger body of text into smaller lines, words . The tokenization helps in interpreting the meaning of the text by analyzing the sequence of the words. There are different methods and libraries available to perform tokenization.
- **Stemming and lemmatization** are methods used by search engines and chatbots to analyze the meaning behind a word. Stemming uses the stem of the word, while lemmatization uses the context in which the word is being used.
- **NLTK, Gensim, Spacy** are some of the libraries that can be used to accomplish these tasks.

Graphical representation of Data



Data Visualization

Analyzing text statistics

Text statistics visualizations contains very insightful techniques. They include:

- word frequency analysis,
- sentence length analysis,
- average word length analysis,

- **Seaborn, Plotly, Bokeh, Matplotlib, Scipy** libraries can be used for data visualization
- **Histograms** (continuous data) and **bar charts** (categorical data) .
- To get the corpus containing **stop-words** we can use the NLTK library. NItk contains stopwords from many languages.
- **Stopwords are the words that are most commonly used in any language** such as *“the”,” a”,” an”* etc. As these words are probably small in length these words may have caused the above graph to be left-skewed.

Data Visualization(Cont.)

Word-cloud is a great way to represent text data. The size and color of each word that appears in the word-cloud indicate it's frequency or importance.

- ***stopwords***: The set of words that are blocked from appearing in the image.
- ***max_words***: Indicates the maximum number of words to be displayed.
- ***max_font_size***: maximum font size.
- There are many more options to create beautiful word clouds.

Feature Engineering

Feature engineering is a process of extracting meaningful information from the raw data to make it usable for machine learning models.

Feature Engineering in NLP

- Firstly and most importantly, it is essential to understand our problems completely. Some techniques can be useful for a certain type of problem only.
- **For example**, for some problems we might need to extract **grammatical features from data** whereas for some problems, we might need to just get the **most frequently occurring words** only.
- It is important to note that feature engineering in NLP is a little different from the other types of data.
- **In NLP**, we are dealing with language or texts, so to derive inputs for our machine learning models, we would need **to transform our text into some sort of numeric representation** so computers can process it. One of our goals would be to represent our text in a computer-friendly manner.

Feature Engineering(Cont.)

Feature extraction methods can be divided into 3 major categories, basic, statistical and advanced/vectorized.

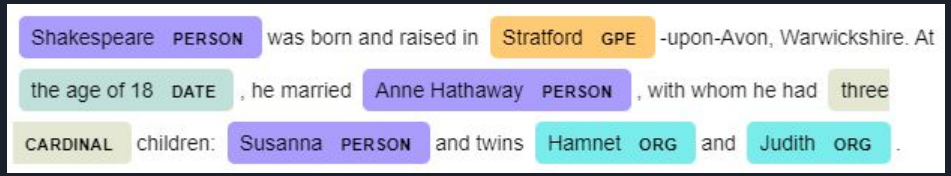
- **Parsing:**
  - **Parsing** is a process of breaking a sentence (or some text) into smaller chunks that helps us understand the syntactic structure and syntactic meaning of the sentence.
  - In NLP, rules of **context-free grammar (CFG)** or **probabilistic context-free grammar (PCFG)** are used to analyze sentences. Building a parser from scratch is a very complex task in itself. We would pick a grammar like CFG or PCFG, then decide upon which type of parser we want to build.
  - **Spacy** or **NLTK**

- **PoS Tagging**
  - **Parts of Speech** (PoS) tagging is the process of marking each word in a corpus with their corresponding part of speech.
  - A tagger or a PoS tagger is a tool that assigns the relevant PoS tag to a given word. Tagging is a tricky task because the part of speech can vary based on the meaning of the sentence as a whole.
  - **Polyglot library** , **NLTK**

Feature Engineering(Cont.)

Name Entity Recognition (NER)

- It is a process of extracting named entities i.e. noun phrases that represent real-world objects like person, location, organization etc from some text.
- It can be used in information extraction, information retrieval, search and recommendation systems.



Feature Engineering(Cont.)

Bag of Words (BoW)

- BoW just represents text in a form of a collection like a bag/set of words where the text can be in the form of documents, sentences etc. BoWs can be used in a wide variety of NLP tasks like document classification, neural feature generation, sentiment analysis etc.
- **Sentence 1:** Matt is a fan of football.
- **Sentence 2:** He also likes to cook occasionally.
- **Sentence 3:** He is a nice guy.
- Based on these sentences, we can create a BoW list as follows.
- **BoW_List** = [“Matt”, “is”, “a”, “fan”, “of”, “football”, “He”, “also”, “likes”, “to”, “cook”, “occasionally”, “nice”, “guy”]
- **scikit-learn library , NLTK**

Feature Engineering(Cont.)

Term Frequency-Inverse Document Frequency (TF-IDF)

Words that occur frequently in a document are more important. But the words that occur very frequently like **“a”, “the”** etc are not important and doesn’t carry any meaning.

- TF-IDF aims to satisfy these two constraints and helps us extract meaningful words from documents. The first component of TF-IDF is the **term frequency method**.
- **Term-Frequency,** as the name suggests calculates the frequency of each of the words present in a document/dataset.
- TF(t) = (Number of times term t appears in a document) / (Total number of terms in the document)
- **IDF** gives higher weightage to the terms that occur only in a few documents. Such terms are useful for discriminating those documents.
- IDF(t) = log10(Total number of documents / Number of documents with termt t in it)
- Our TF-IDF equation would be as **TF-IDF = TF*IDF**
- **Scikit-learn library**
- You can use TF-IDF for text analysis, summarization, search, document classification etc.

Feature Engineering(Cont.)

Named entity recognition

- Named entity recognition is an information extraction method in which entities that are present in the text are classified into predefined entity types like “Person”,” Place”,” Organization”, etc. By using **NER we can get great insights about the types of entities present in the given text dataset**.
- There are three standard libraries to do Named Entity Recognition: Standford NER, spaCy, NLTK
- **SpaCy provides many other functionalities like pos tagging, word to vector transformation, etc.**

Sentiment Analysis:

- Sentiment analysis is a very common natural language processing task in which we **determine if the text is positive, negative or neutral.**
- This is very useful for finding the sentiment associated with reviews, comments which can get us some valuable insights out of text data.

Textblob

- Textblob is a python library built on top of nltk. It has been around for some time and is very easy and convenient to use.
- The sentiment function of TextBlob returns two properties:
  - ***polarity:*** is a floating-point number that lies in the range of *[-1,1]* where **1 means positive** statement and **-1 means a negative** statement.
  - ***subjectivity*:** refers to **how someone’s judgment is shaped by personal opinions** and feelings. Subjectivity is represented as a floating-point value which lies in the range of [0,1].

Feature Engineering(Cont.)

```
from textblob import TextBlob
TextBlob('100 people killed in Iraq').sentiment
```

```
Sentiment(polarity=-0.2, subjectivity=0.0)
```

TextBlob claims that the text *"100 people killed in Iraq"* is negative and is not an opinion or feeling but rather a factual statement.

Vader sentiment analysis

- **Vader works better in detecting negative sentiment**. It is very useful in the case of **social media** text sentiment analysis.
- VADER sentiment analysis class **returns a dictionary that contains the probabilities of the text for being positive, negative and neutral.** Then we can filter and choose the sentiment with most probability.

Feature Engineering(Cont.)

Advanced Methods

Word2Vec

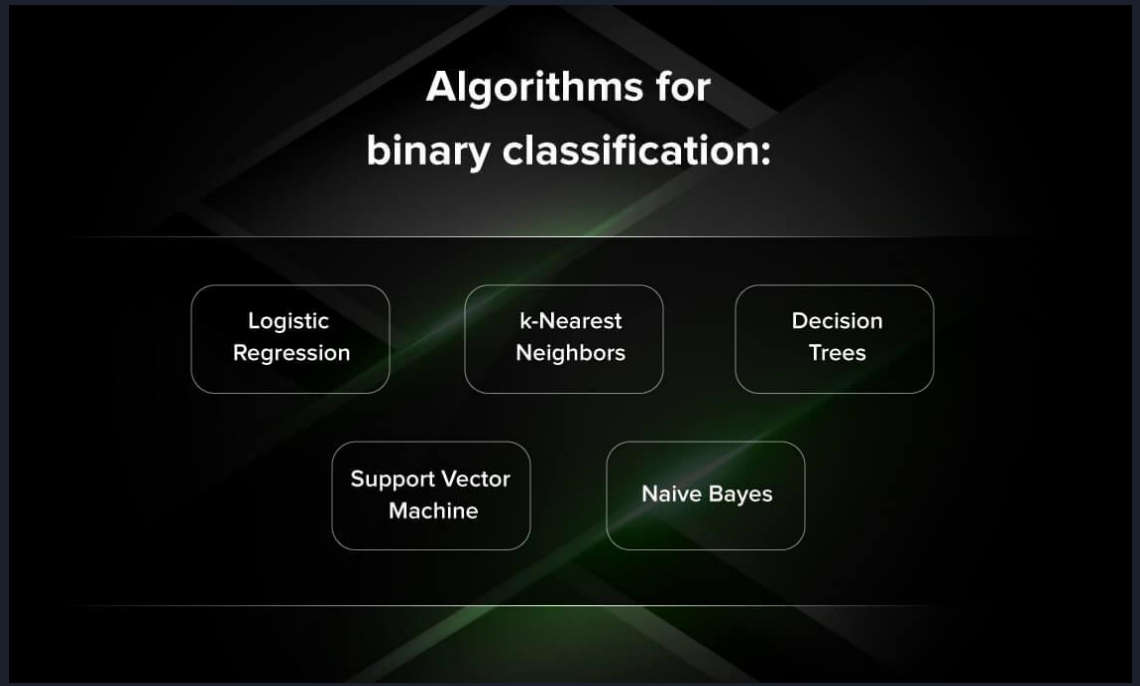
- This technique aims to map a word to a fixed-length vector. We can train a neural network to generate word2vec embeddings for any kind of text data.
- Words that have similar meanings or are related closely, when mapped into a vector space would appear closer, like in a cluster. This can help us understand the semantics of the words in a sentence better than any previously mentioned technique.
- We can extract most similar word, odd-one out
- **Gensim**

Model Building in NLP

Model Building Approaches

- **ML Approach**
  a) Supervised Learning(Classification etc)
  b) UnSupervised Learning(Clustering etc)
- **DL Approach**
  a) RNN(Text Processing etc)
  b) CNN(Image Processing etc)

ML Approach



Naive Bayes is used for spam email filtering, language translation, sentiment analysis etc.

Model Building in NLP

DL Approach

- CNN (better for images)
- RNN (better for text)
- LSTM (extension of rnn that extend the memory)
- GRU (use gates)

Model Evaluation

- Accuracy
- Precision
- Recall
- F1 Score
- Area Under the Curve (AUC)
- Mean Reciprocal Rank (MRR)
- Mean Average Precision (MAP)
- Root Mean Squared Error (RMSE)

Model Evaluation(Cont.)

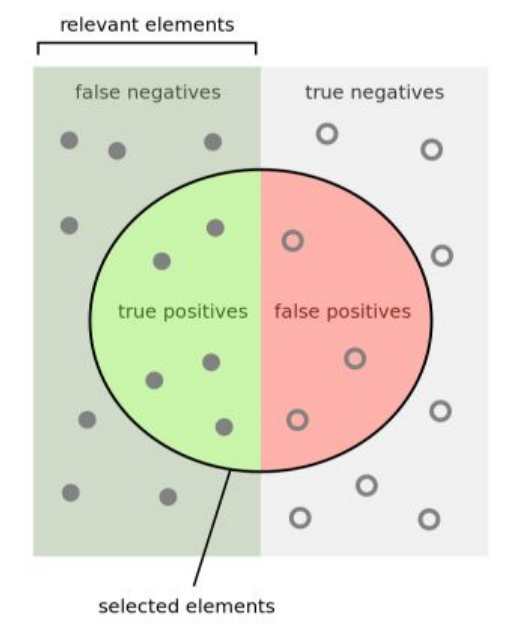
- **Accuracy** is defined as the percentage of correct predictions for the test data. It can be calculated easily by dividing the number of correct predictions by the number of total predictions.

  Accuracy = correct predictions/all predictions
- **Precision** is defined as the fraction of relevant examples (true positives) among all of the examples which were predicted to belong in a certain class.

  precision=true positives/true positives+false positives
- **Recall** is defined as the fraction of examples which were predicted to belong to a class with respect to all of the examples that truly belong in the class.

  recall=true positives/true positives+false negatives

Model Evaluation(Cont.)



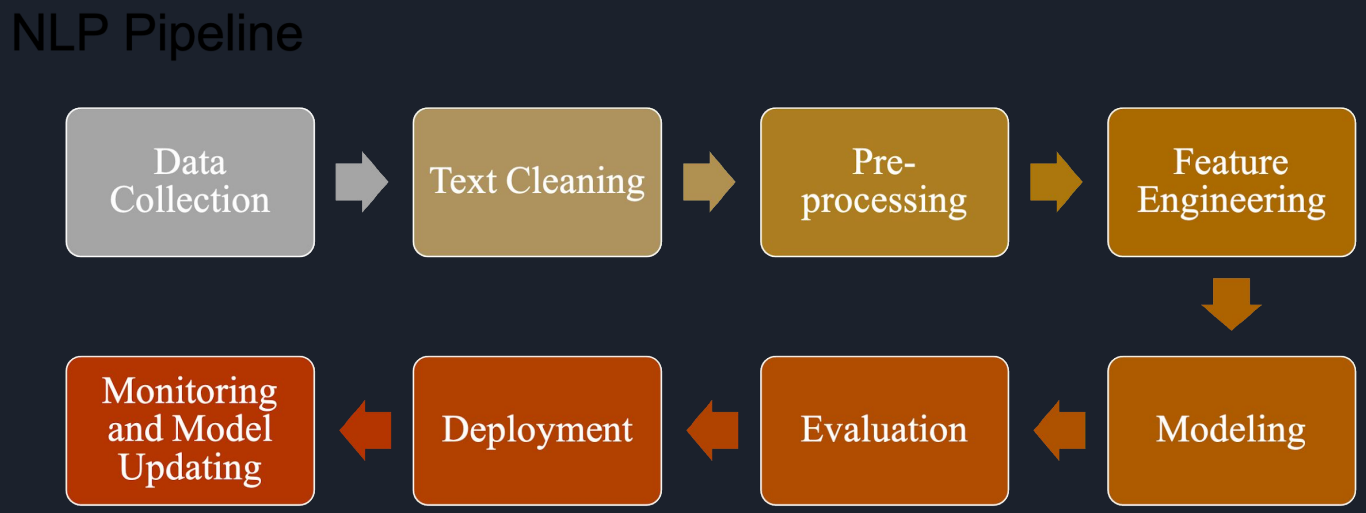
- **True positives** occur when your system predicts that an observation belongs to a class and it actually does belong to that class.
- **True negatives** occur when your system predicts that an observation does not belong to a class and it does not belong to that class.
- **False positives** occur when you predict an observation belongs to a class when in reality it does not. Also known as a type 2 error.
- **False negatives** occur when you predict an observation does not belong to a class when in fact it does. Also known as a type 1 error.

Model Evaluation(Cont.)

- **F1 score** is the weighted average of precision and recall.
- **The Area Under the Curve (AUC)** is the measure of the ability of a classifier to distinguish between classes and is used as a summary of the ROC curve. The higher the AUC, the better the performance of the model at distinguishing between the positive and negative classes.
- **Mean Reciprocal Rank** is a measure to evaluate systems that return a ranked list of answers to queries.
- **Root Mean Square Deviation(RMSE)** measures the average magnitude of the errors and is concerned with the deviations from the actual value. RMSE value with zero indicates that the model has a perfect fit. The lower the RMSE, the better the model and its predictions.

NLP Pipeline

The set of ordered stages one should go through from a labeled dataset to creating a classifier that can be applied to new samples (AKA supervised machine learning classification) is called the NLP pipeline.



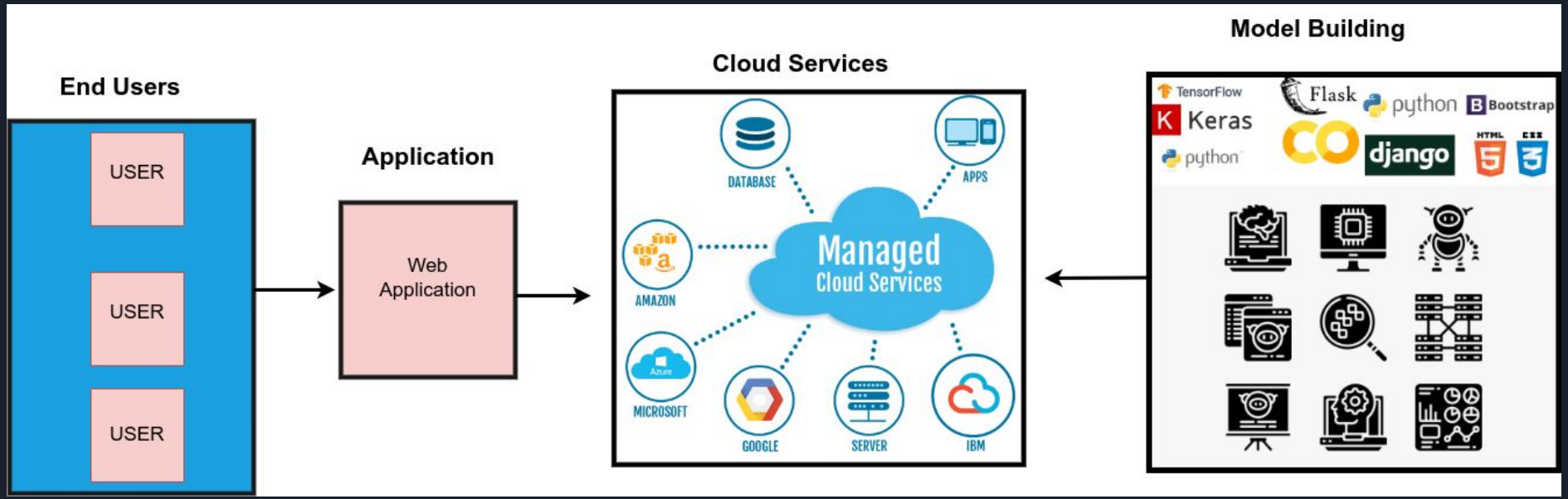
Development

- **Django** is a high-level Python web framework that enables rapid development of secure and maintainable websites. We will use Django for **back end development** .
- We will create the **front end** of our application by simply using Html, CSS, Javascript.

Deployment

- We pickle our trained or tested model using Pickle Library.
- We integrate our pickle file into a web application using Flask or Django.
- After that we need to deploy it on some cloud service (Amazon web services, Google Cloud platform or Microsoft azure).
- We also need of **Docker(for containerized application**) and **Kubernetes** (for automating deployment, scaling and management of your containerized application).

Deployment(Cont.)



References

https://www.mihaileric.com/posts setting-up-a-machine-learning-project/

https://towardsdatascience.com/data-visualization-for-machine-learning-and-data-science-a45178970be7

https://builtin.com/machine-learning/nlp-machine-learning

https://towardsdatascience.com/my-first-nlp-pipeline-99d24aafb773