CSE558 **Data Science**

Final Project Presentation

Aditya Girdhar (**2021005**)
Bhavya Narnoli (**2021316**)
Hardik Singh (**2021390**)
Dhruv Sood (**2021387**)

INDRAPRASTHA INSTITUTE *of*
INFORMATION TECHNOLOGY
DELHI

Recap (1/4): Dataset Description

Built-in features

1. drugName *(categorical)*: name of drug
2. condition *(categorical)*: name of condition
3. review *(text)*: patient review
4. rating *(numerical)*: 10-star patient rating
5. date *(date)*: date of review entry
6. usefulCount *(numerical)*: no. of users who found review useful

Engineered Features

1. sentimentPolarity *(numerical)*: measured using NLTK
2. usefulScore *(numerical)*: normalised usefulCount*rating, essentially a weighted average of rating scaled down from 0 - 1 . (usefulCount

Recap (2/4): Problem Statement

We aim to cluster users based on their review patterns and preferences and recommend drugs to users who have reviewed drugs for similar conditions or have given similar ratings to drugs.

1. **Problem A**: Can we predict user’s condition from their review?
2. **Problem B**: Can we recommend medicines on the basis of given condition and review?

Recap (3/4): Preprocessing

1. Made all reviews **lower-case**
2. Cleaned the text by removing any HTML artefacts
3. Removed numbers (since they often mess with language processing)
4. Expanded all contractions (you’ve is a contraction of you have)

Recap (4/4): Empirical Evaluations

1. **Hypothesis Testing**:
   a. Conducted chi-square tests for rating independence.
   b. Performed t-tests to assess correlation between ratings and sentiment scores.
2. **Insights Gained**:
   a. Identified key patterns and trends in patient reviews.
   b. Discovered correlations between review sentiment and drug ratings.
3. **Challenges Overcome**:
   a. Addressed data quality issues including unformatted HTML codes and missing values.
   b. Managed complexities in data standardization and feature extraction.

First Goal

Predict Condition from Review

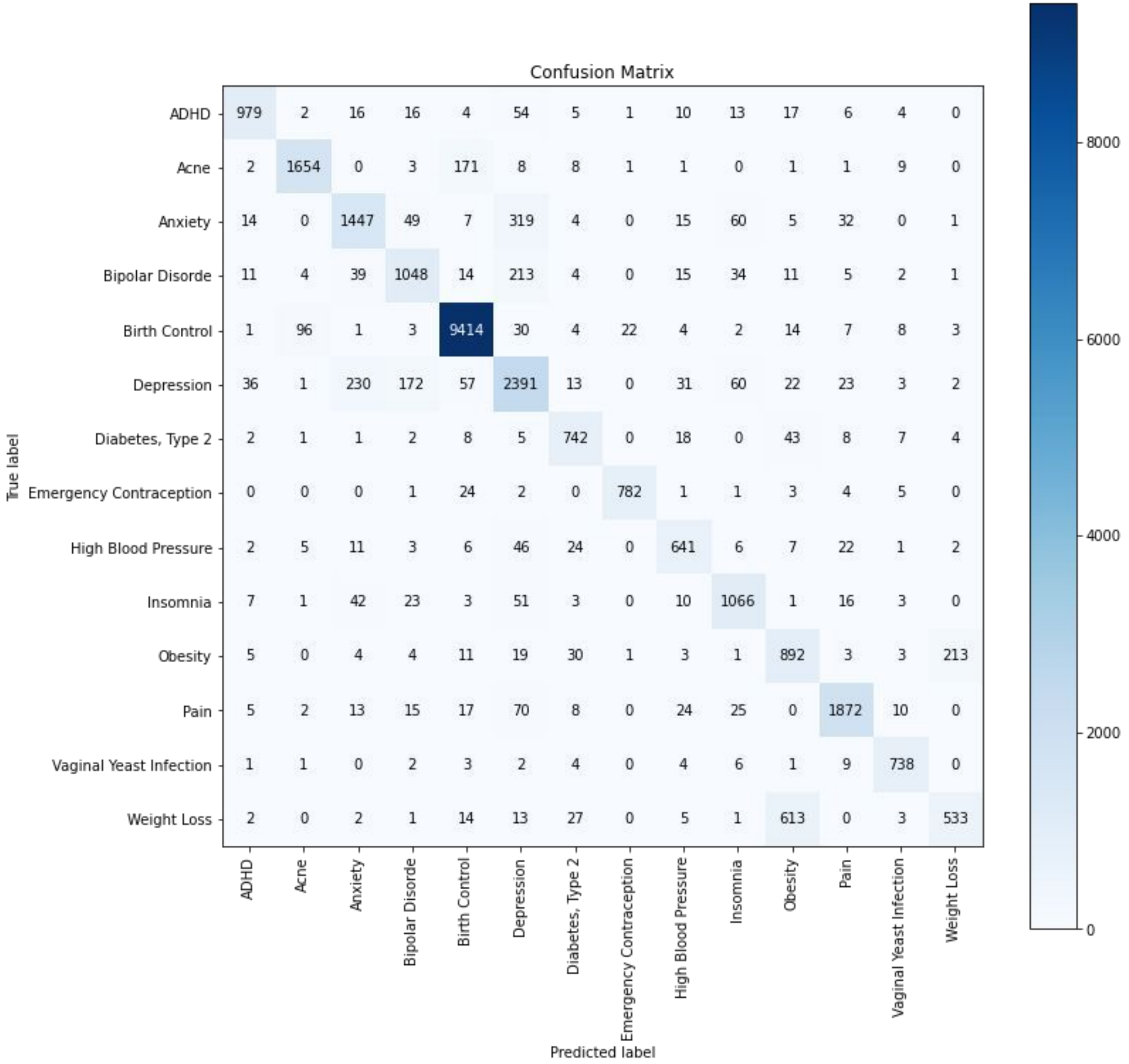
Model: Long Short-Term Memory Network

Architecture Overview

1. **Embedding Layer**: Uses nn.Embedding for token-to-vector conversion with fixed pre-trained word vectors (embedding_matrix).
2. **Bidirectional LSTM**: Employs nn.LSTM bidirectionally to capture past/future context (Input: embed_size, Hidden: self.hidden_size).
3. **Linear Layer & Activation**: Reduces LSTM output, applying ReLU for non-linearity.
4. **Pooling & Concatenation**: Uses average and max pooling to capture temporal information, concatenating these for feature extraction.
5. **Dropout**: Helps prevent overfitting by deactivating 10% of units during training (drp = 0.1) using self.dropout.

Model Accuracy: **86.8%**

LSTM Confusion Matrix



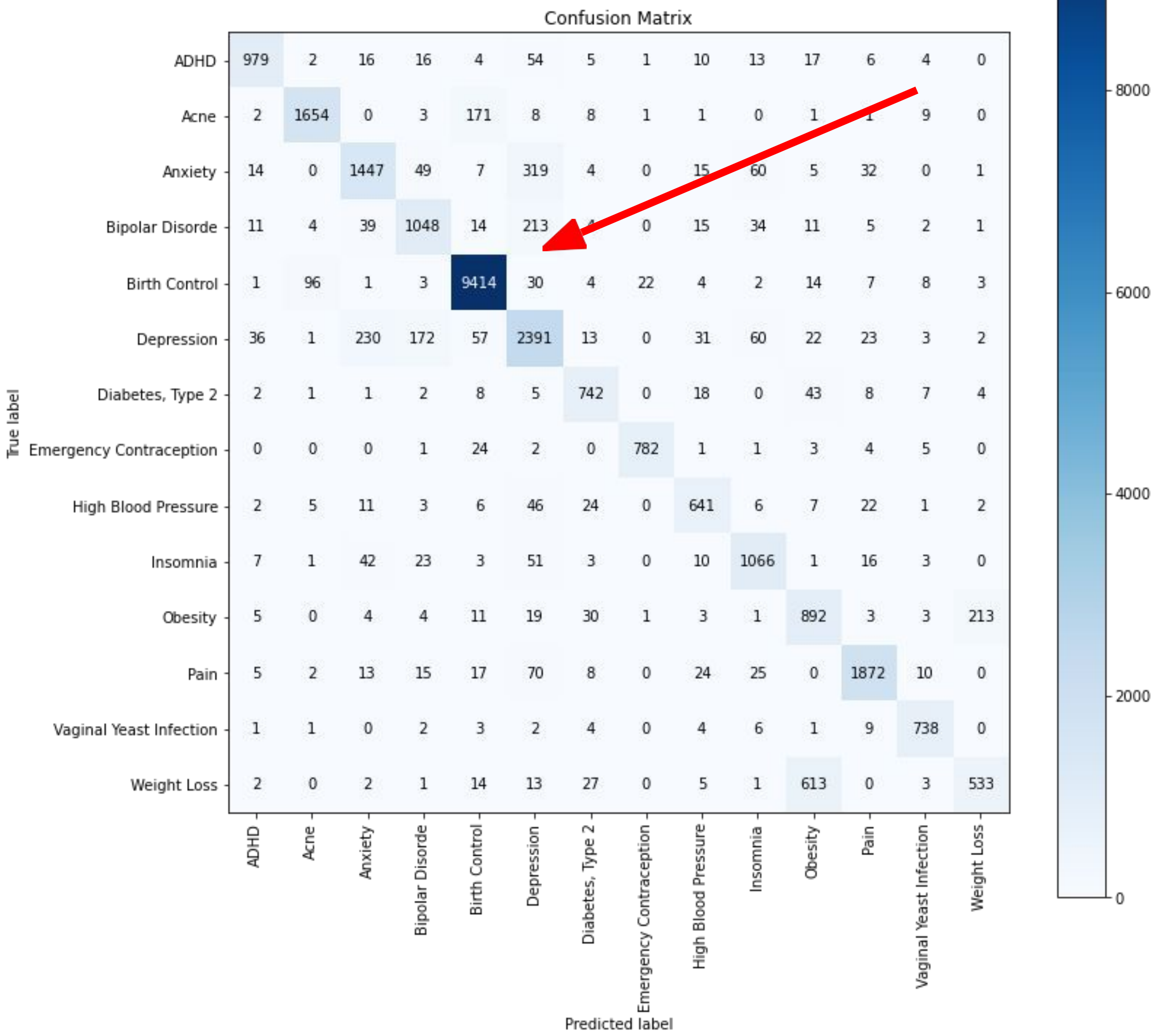
Confusion Matrix

True label

Predicted label

	ADHD	Acne	Anxiety	Bipolar Disorde	Birth Control	Depression	Diabetes, Type 2	Emergency Contraception	High Blood Pressure	Insomnia	Obesity	Pain	Vaginal Yeast Infection	Weight Loss
ADHD	979	2	16	16	4	54	5	1	10	13	17	6	4	0
Acne	2	1654	0	3	171	8	8	1	1	0	1	1	9	0
Anxiety	14	0	1447	49	7	319	4	0	15	60	5	32	0	1
Bipolar Disorde	11	4	39	1048	14	213	4	0	15	34	11	5	2	1
Birth Control	1	96	1	3	9414	30	4	22	4	2	14	7	8	3
Depression	36	1	230	172	57	2391	13	0	31	60	22	23	3	2
Diabetes, Type 2	2	1	1	2	8	5	742	0	18	0	43	8	7	4
Emergency Contraception	0	0	0	1	24	2	0	782	1	1	3	4	5	0
High Blood Pressure	2	5	11	3	6	46	24	0	641	6	7	22	1	2
Insomnia	7	1	42	23	3	51	3	0	10	1066	1	16	3	0
Obesity	5	0	4	4	11	19	30	1	3	1	892	3	3	213
Pain	5	2	13	15	17	70	8	0	24	25	0	1872	10	0
Vaginal Yeast Infection	1	1	0	2	3	2	4	0	4	6	1	9	738	0
Weight Loss	2	0	2	1	14	13	27	0	5	1	613	0	3	533

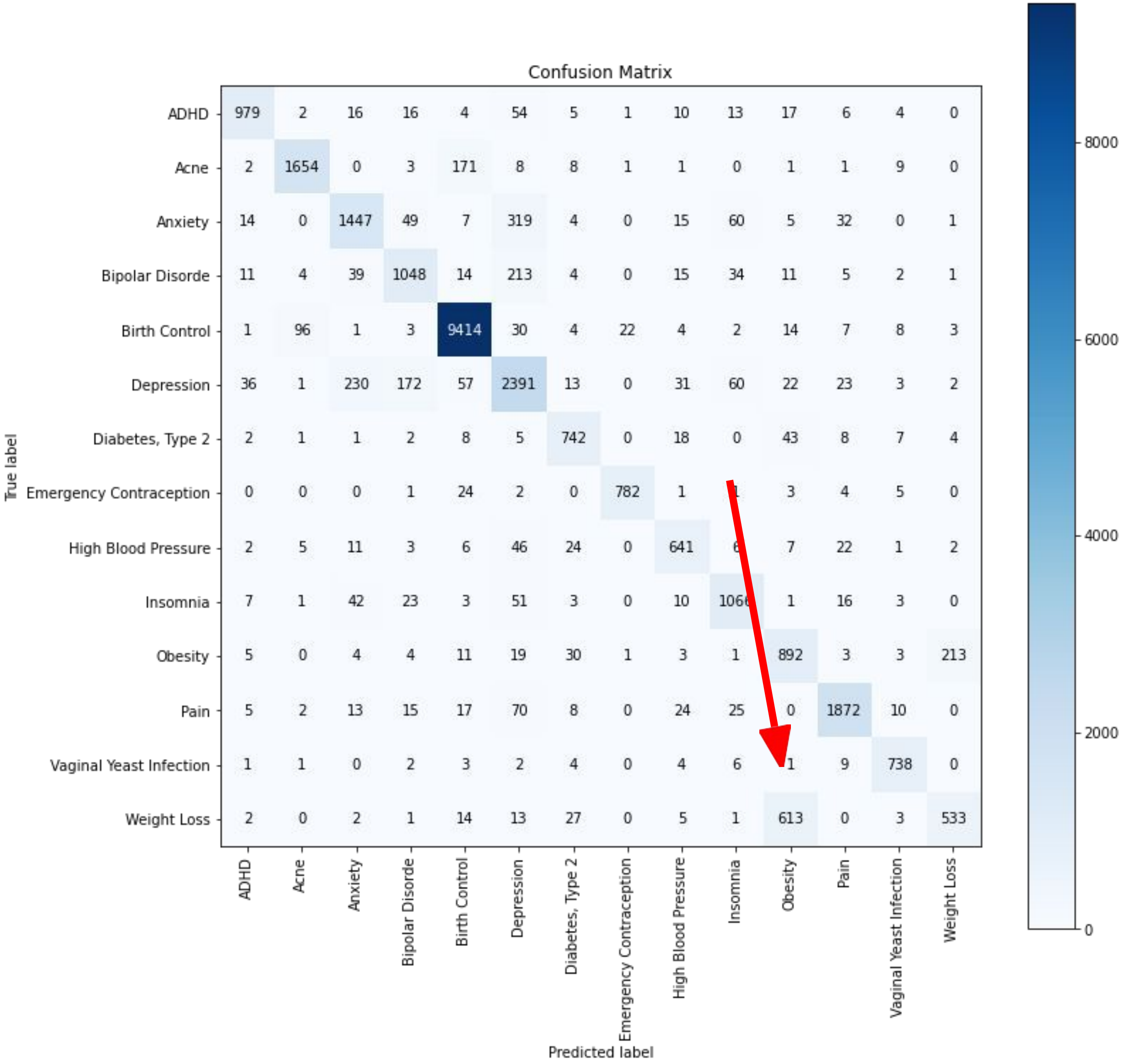
LSTM Confusion Matrix



Confusion Matrix

True label \ Predicted label	ADHD	Acne	Anxiety	Bipolar Disorder	Birth Control	Depression	Diabetes, Type 2	Emergency Contraception	High Blood Pressure	Insomnia	Obesity	Pain	Vaginal Yeast Infection	Weight Loss
ADHD	979	2	16	16	4	54	5	1	10	13	17	6	4	0
Acne	2	1654	0	3	171	8	8	1	1	0	1	1	9	0
Anxiety	14	0	1447	49	7	319	4	0	15	60	5	32	0	1
Bipolar Disorde	11	4	39	1048	14	213	4	0	15	34	11	5	2	1
Birth Control	1	96	1	3	9414	30	4	22	4	2	14	7	8	3
Depression	36	1	230	172	57	2391	13	0	31	60	22	23	3	2
Diabetes, Type 2	2	1	1	2	8	5	742	0	18	0	43	8	7	4
Emergency Contraception	0	0	0	1	24	2	0	782	1	1	3	4	5	0
High Blood Pressure	2	5	11	3	6	46	24	0	641	6	7	22	1	2
Insomnia	7	1	42	23	3	51	3	0	10	1066	1	16	3	0
Obesity	5	0	4	4	11	19	30	1	3	1	892	3	3	213
Pain	5	2	13	15	17	70	8	0	24	25	0	1872	10	0
Vaginal Yeast Infection	1	1	0	2	3	2	4	0	4	6	1	9	738	0
Weight Loss	2	0	2	1	14	13	27	0	5	1	613	0	3	533

LSTM Confusion Matrix



	ADHD	Acne	Anxiety	Bipolar Disorde	Birth Control	Depression	Diabetes, Type 2	Emergency Contraception	High Blood Pressure	Insomnia	Obesity	Pain	Vaginal Yeast Infection	Weight Loss
ADHD	979	2	16	16	4	54	5	1	10	13	17	6	4	0
Acne	2	1654	0	3	171	8	8	1	1	0	1	1	9	0
Anxiety	14	0	1447	49	7	319	4	0	15	60	5	32	0	1
Bipolar Disorde	11	4	39	1048	14	213	4	0	15	34	11	5	2	1
Birth Control	1	96	1	3	9414	30	4	22	4	2	14	7	8	3
Depression	36	1	230	172	57	2391	13	0	31	60	22	23	3	2
Diabetes, Type 2	2	1	1	2	8	5	742	0	18	0	43	8	7	4
Emergency Contraception	0	0	0	1	24	2	0	782	1	1	3	4	5	0
High Blood Pressure	2	5	11	3	6	46	24	0	641	6	7	22	1	2
Insomnia	7	1	42	23	3	51	3	0	10	1066	1	16	3	0
Obesity	5	0	4	4	11	19	30	1	3	1	892	3	3	213
Pain	5	2	13	15	17	70	8	0	24	25	0	1872	10	0
Vaginal Yeast Infection	1	1	0	2	3	2	4	0	4	6	1	9	738	0
Weight Loss	2	0	2	1	14	13	27	0	5	1	613	0	3	533

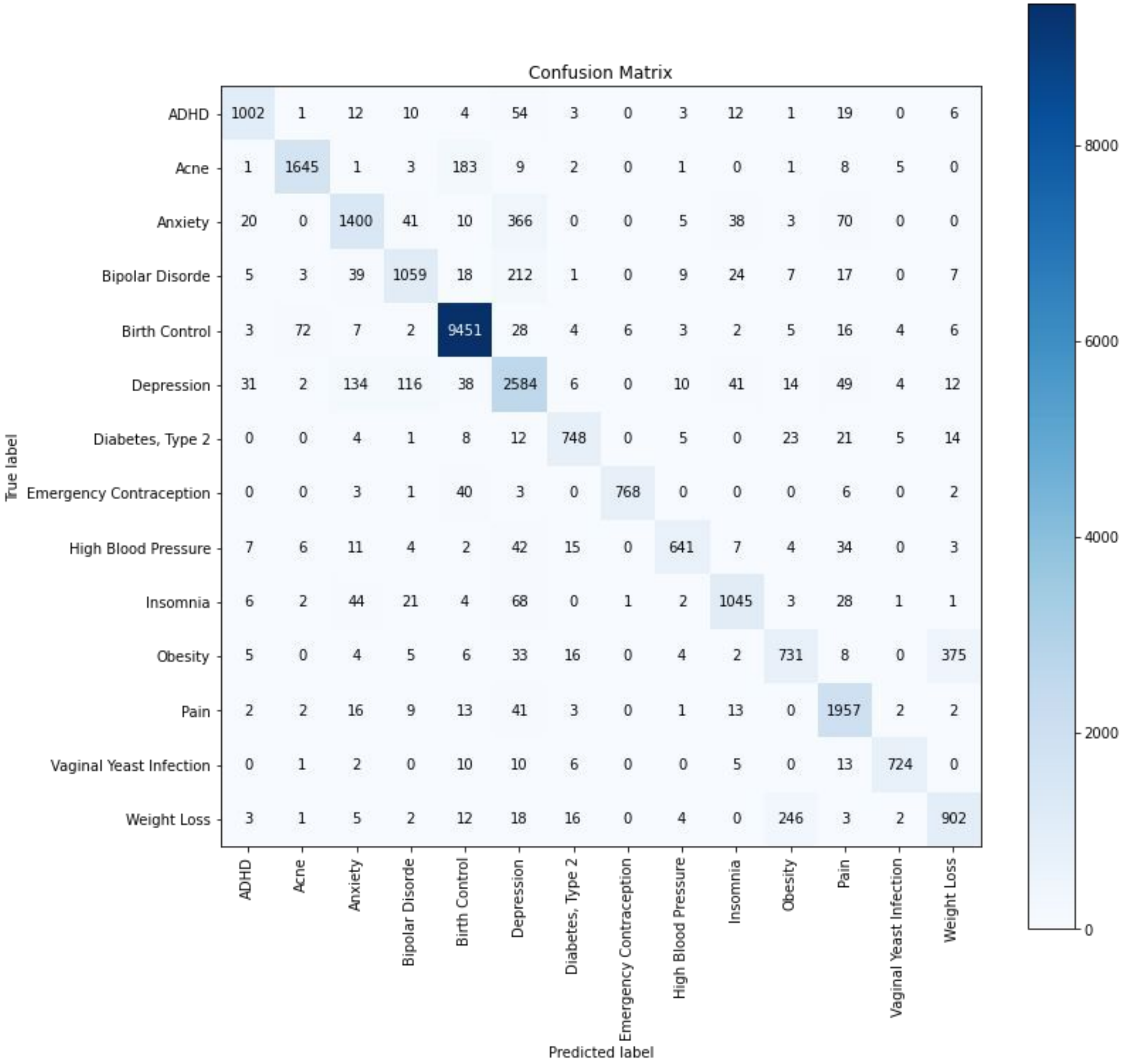
Model: Convolutional Neural Network

Architecture Overview

1. **Embedding Layer**: Converts tokens to dense vectors using pre-trained embeddings (fixed weights).
2. **Convolutional Layers**: Multiple layers capture n-grams with varied kernel sizes. Extracts features using num_filters filters.
3. **Activation and Pooling**: Applies ReLU for non-linearity and max pooling along the temporal dimension.
4. **Dropout**: Mitigates overfitting by deactivating 10% units randomly.
5. **Fully Connected Layer**: Flattens pooled output for predictions with neuron count determined by filter_sizes and num_filters.

Model Accuracy: **88.4%**

CNN Confusion Matrix



Confusion Matrix

True label \ Predicted label	ADHD	Acne	Anxiety	Bipolar Disorde	Birth Control	Depression	Diabetes, Type 2	Emergency Contraception	High Blood Pressure	Insomnia	Obesity	Pain	Vaginal Yeast Infection	Weight Loss
ADHD	1002	1	12	10	4	54	3	0	3	12	1	19	0	6
Acne	1	1645	1	3	183	9	2	0	1	0	1	8	5	0
Anxiety	20	0	1400	41	10	366	0	0	5	38	3	70	0	0
Bipolar Disorde	5	3	39	1059	18	212	1	0	9	24	7	17	0	7
Birth Control	3	72	7	2	9451	28	4	6	3	2	5	16	4	6
Depression	31	2	134	116	38	2584	6	0	10	41	14	49	4	12
Diabetes, Type 2	0	0	4	1	8	12	748	0	5	0	23	21	5	14
Emergency Contraception	0	0	3	1	40	3	0	768	0	0	0	6	0	2
High Blood Pressure	7	6	11	4	2	42	15	0	641	7	4	34	0	3
Insomnia	6	2	44	21	4	68	0	1	2	1045	3	28	1	1
Obesity	5	0	4	5	6	33	16	0	4	2	731	8	0	375
Pain	2	2	16	9	13	41	3	0	1	13	0	1957	2	2
Vaginal Yeast Infection	0	1	2	0	10	10	6	0	0	5	0	13	724	0
Weight Loss	3	1	5	2	12	18	16	0	4	0	246	3	2	902

Why does **CNN** perform better than **LSTM**?

1. **Local Feature Extraction**: CNNs excel at capturing local patterns and features within data. In text data, for instance, they can identify and learn representations of specific word sequences or n-grams efficiently. For tasks where local structures are crucial (like in image analysis or certain types of text classification), CNNs might outperform LSTMs.
2. **Translation Invariance**: CNNs inherently possess translation invariance, which means they can recognize patterns regardless of their location in the input. This property is particularly useful in tasks where the position of features doesn’t significantly impact the final prediction, such as in image classification.

Second Goal

Recommend Medicines

Term Frequency: Inverse Document Frequency

TF-IDF : Weighting strategy in which words are offered with weight, not count.

It gives low importance to the terms that often appear in the dataset, which implies TF-IDF estimates relevance, not a recurrence. Term frequency (TF) can be called the likelihood of locating a word in a document.

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

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.

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

TF-IDF is the multiplication of TF with IDF, suggesting how vital and relevant a word is in the document. The selected n-gram range for TF-IDF in this is (1,2). [unigram and bigram]

Source: link

Features and Metric for recommendation

Metric :

usefulScore was normalised if usefulScore > 0.5 : Recommendable Drug

else if usefulScore < 0.5 : Not Recommendable Drug

Training features : TF-IDF matrix, along with condition, sentiment_score

Output Binary Prediction : useful Score

Distribution of the data

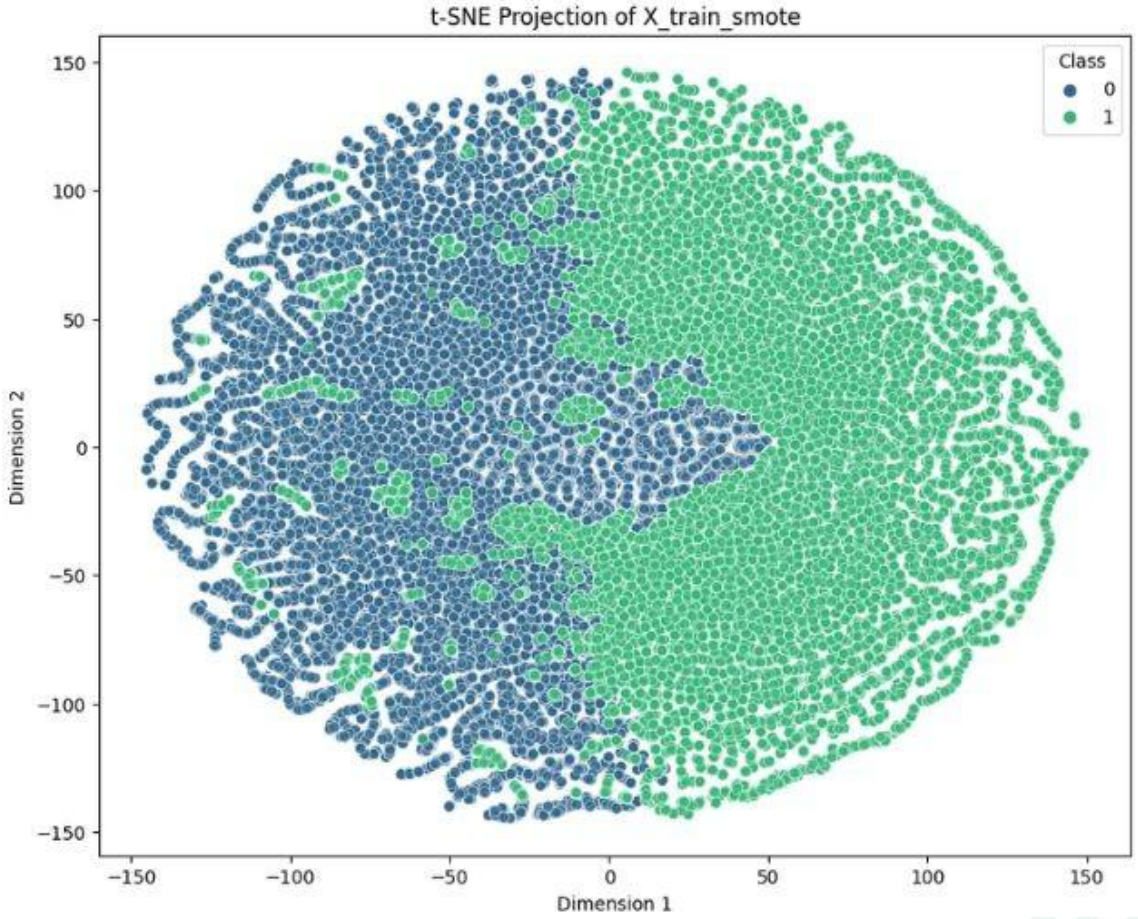
0 : 206755

1 : 5923

This had class imbalance issue, which was overcome by using SMOTE oversampling which uses KNN in the feature space to oversample the minor class, by generating similar data

T-sne of above trainable features

The linear separability could indicate that these clusters are easily distinguishable from each other.



Models (Preliminary)

Perceptron: A perceptron is a basic artificial neuron that takes binary inputs, applies weights, sums them up, passes through an activation function, and outputs a binary result. The same above inputs and outputs were considered in building the model. It demonstrates in an overall accuracy of **51%**.

Logistic Regression is a binary classification algorithm that models the probability of an instance belonging to a particular class, providing precision, recall, and F1-score metrics, with an overall accuracy of **62%**.The same above features (target and input were used in the model)

Random Forest is an ensemble model known for its robustness and ability to handle complex relationships.It was used to predict that for given (sentiment score, TF-IDF vectors, condition) the medicine should be recommended or not. The model showed high precision, recall, and F1-scores for both positive and negative classes, resulting in an overall accuracy of **98%**.

Best Medicine for a given condition

The code iterates through each unique medical condition in the training set. For each condition, it identifies the data point with the highest confidence score (distance from the decision boundary) as predicted by the logistic regression model.

Output file : link

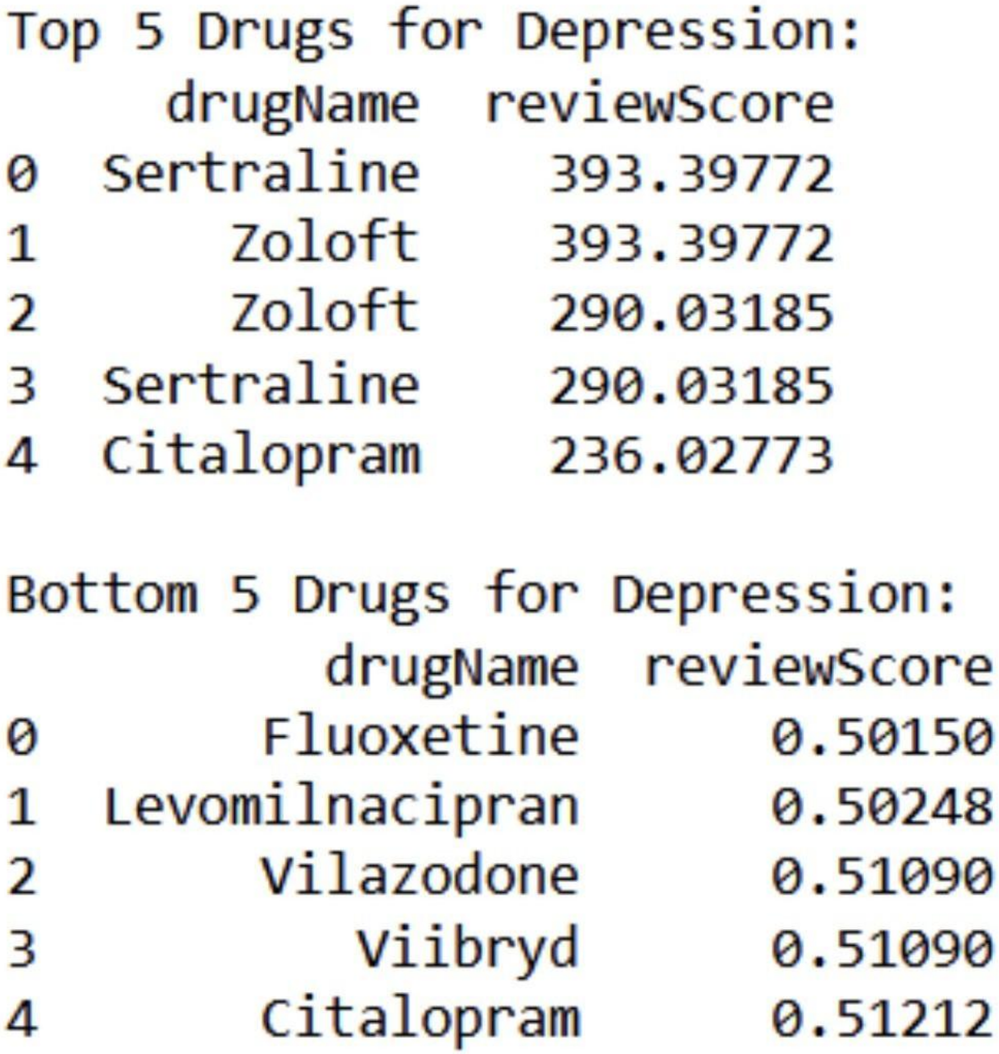
Model: Best-K Review Scores

We devised a composite score to gauge product/service performance based on textual reviews. The formula is as follows:

review_score = 0.6*rating + 0.3*useful_count + 0.1*sentiment_score

Top 5 and Bottom 5 Drugs for a condition *Guides doctors' decisions, showcasing public sentiment. Valuable for research, identifying concerns with poorly-rated drugs, fostering proactive healthcare management.*

Model: Best-K Review Scores



```
Top 5 Drugs for Depression:
     drugName  reviewScore
0  Sertraline    393.39772
1      Zoloft    393.39772
2      Zoloft    290.03185
3  Sertraline    290.03185
4  Citalopram    236.02773

Bottom 5 Drugs for Depression:
          drugName  reviewScore
0       Fluoxetine      0.50150
1  Levomilnacipran      0.50248
2        Vilazodone      0.51090
3          Viibryd      0.51090
4       Citalopram      0.51212
```

Thank you!

Team Members

- Aditya Girdhar (**2021005**)
- Bhavya Narnoli (**2021316**)
- Hardik Singh (**2021390**)
- Sanskar Ranjan (**2021096**)

All members equally contributed to this project.