**AYDIN ADNAN MENDERES UNIVERSITY**

**ENGINEERING FACULTY**

**COMPUTER SCIENCE ENGINEERING DEPARTMENT**



**NER On Medical Text Second Part**

**CSE431 – Natural Language Processing with Machine Learning 2023/2024**

**Burak TÜZEL**

**Talha Alper ASAV**

**Lecturer:**

**Asst. Prof. Dr. Fatih SOYGAZİ**

**Named Entity Recognition on Medical Text Second Part**

**Installing necessary environment and importing the libraries:**

1. Install Jupyter Notebook

A screen shot of a computer program

Description automatically generated

**Reading the Medical Text Without NER and Preprocessing:**

Here is the medical text link to download: <https://www.kaggle.com/datasets/tboyle10/medicaltranscriptions>

Firstly we read the medical samples data. Then creating a method to  get unique words(vocabulary) and sentence count in a list of text

A screenshot of a computer program

Description automatically generated

**Finding the Disease, Drugs and Drugs-Doses Named Entities:**

A screenshot of a computer program

Description automatically generatedHere we checked the number of samples in each category. Since some categories have less than 50 we will remove them (You can check the full output from the given .html file)

A screenshot of a computer

Description automatically generated

A screenshot of a computer code

Description automatically generated**Preprocessing to Increase the Accuracy:**

A screenshot of a computer program

Description automatically generatedHere we grouped the subcategories into their specialties.

**Here We Used TF-IDF Vectorizer and Started Building Our Models:**

Here is our vectorizer code. We will split it and feed it to the models.

A white background with text

Description automatically generated

1. Multinomial Naïve Bayes

A green and white grid with black text

Description automatically generated

A screenshot of a computer

Description automatically generated

1. Random Forest

A green and white grid with black text

Description automatically generated

A screenshot of a computer

Description automatically generated

1. XGBoost

A screenshot of a grid

Description automatically generated

A screenshot of a computer screen

Description automatically generated

1. A screenshot of a crossword puzzle

   Description automatically generatedLightGBM

A screenshot of a computer

Description automatically generated

1. 1D CNN, LSTM and GRU

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**Here We Used Bag-of-Words (CountVectorizer) and Started Building Our Models:**

A screen shot of a computer program

Description automatically generatedHere is our vectorizer code. We will split it and feed it to the models.

1. Multinomial Naïve Bayes

A screenshot of a crossword puzzle

Description automatically generated

A screenshot of a computer

Description automatically generated

1. , Random Forest

A screenshot of a graph

Description automatically generated

A screenshot of a computer

Description automatically generated

1. XGBoost

A screenshot of a graph

Description automatically generated

A screenshot of a computer

Description automatically generated

1. LightGBM

A screenshot of a graph

Description automatically generated

A screenshot of a computer

Description automatically generated

1. 1D CNN, LSTM and GRU

A screenshot of a computer screen

Description automatically generated

**Here We Used TF-IDF Vectorizer for the First Part Output and Started Building Our Models:**

Here is our vectorizer code. We will split it and feed it to the models.

A screen shot of a computer program

Description automatically generated

1. Multinomial Naïve Bayes

A blue squares with numbers

Description automatically generated

A screenshot of a computer

Description automatically generated

1. Random Forest

A blue squares with numbers

Description automatically generated

A screenshot of a computer screen

Description automatically generated

1. XGBoost

A blue squares with numbers

Description automatically generated

A screenshot of a computer

Description automatically generated

1. LightGBM

A graph with numbers and squares

Description automatically generated

A screenshot of a computer

Description automatically generated

1. 1D CNN, LSTM and GRU

A diagram of a network model

Description automatically generated

A screenshot of a computer

Description automatically generated

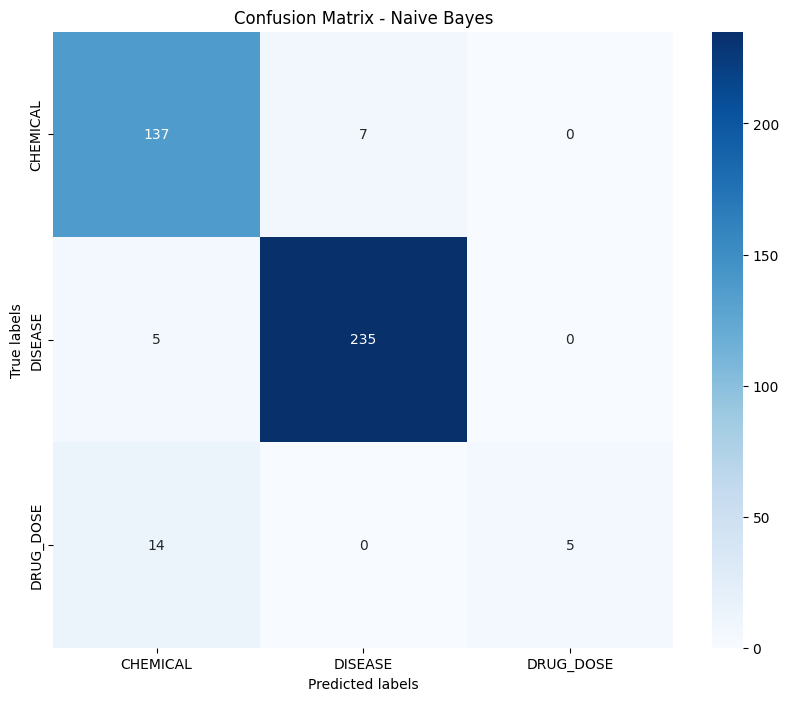
**Here We Used CountVectorizer for the First Part Output and Started Building Our Models:**

Here is our vectorizer code. We will split it and feed it to the models.

A screenshot of a computer code

Description automatically generated

1. Multinomial Naïve Bayes



A screenshot of a computer

Description automatically generated

1. Random Forest

A blue squares with white text

Description automatically generated

A screenshot of a computer

Description automatically generated

1. XGBoost

A blue squares with white text

Description automatically generated

A screenshot of a computer screen

Description automatically generated

1. LightGBM

A graph of blue squares

Description automatically generated with medium confidence

A screenshot of a computer

Description automatically generated

1. 1D CNN, LSTM and GRU

A diagram of a diagram

Description automatically generated with medium confidence

A screenshot of a computer screen

Description automatically generated

**And Finally Applying SMOTE to Our Best Models:**

For the first input our best model was Multinominal Naïve Bayes with the CountVectorizer.

Here is before oversampling scores.

A screenshot of a computer

Description automatically generated

And here is after oversampling scores.

A screenshot of a computer

Description automatically generated

And for the second input our best model was Random Forest. So we applied SMOTE and compared results for it.

Here are the results before oversampling.

A screenshot of a computer

Description automatically generated

And here are the results after oversampling.

A screenshot of a computer screen

Description automatically generated

While we saw some improvements on the first input side, for the second input side it got worse.

**REFERENCES**

1. <https://www.kaggle.com/datasets/tboyle10/medicaltranscriptions>
2. <https://www.kaggle.com/code/ritheshsreenivasan/clinical-text-classification>

ssss