*e*Drug

by Team 4:

Venkata Vamsi Krishna Bhuvanam

Priyadarsini Nidadavolu

Bhulakshmi Makkena

Tej Kumar Yentrapragada

TABLE OF CONTENTS

1 MOTIVATION

The motivation of building a recommendation system for personalized treatment is to diagnosis a disease based on symptoms provided by the users and then recommend a suitable and appropriate drug for the user based on few criteria's like past behavior and historical data of past similar symptoms of the patients along with the drug allergies that the patient has in the past.

Our system will recommend few drugs to select from a list which are approved by the pharmacist. This option will help a user to stream into the list of interest across more than one domain. The system would be able to return a suggestive drug in the list using various technologies that we plan on implementing.

2 OBJECTIVES

The main objective of the recommendation system for E-drug is to recommend a drug to the user based on the symptoms that he provided.

Our system takes user symptoms as input and it analyses the input with the types of diseases and produces a dataset with various possible diseases.

It analysis the type of diseases and ultimately recommends an appropriate drug based on the following criteria:

- Past Behavior of the patient
- Historical data of the past similar symptoms of the patients
- Drug allergies
- Drug similarities that the doctor recommended previously

Our recommendation system would return the dataset of list of recommended drugs with a hyperlink which leads to the composition, availability of the drug in various stores, the cost etc. This would help the user to view the list and can go and check for nearby pharma and buy it.

Machine learning and ALS algorithms run in the background and the result would be an recommended drug dataset.

3 EXPECTED OUTCOME

At the end, this machine should be able to take the dataset and recommend the patient/doctor with the suitable drug without any future side effects.

4 DOMAIN CHOSEN

For this project, Recommendation system will help a lot to analyze the data of symptoms for diseases and its drug compositions.

Also we have used a Spark Core Libraries and Spark Machine Learning Libraries.

5 DATA COLLECTION

Static Data:

http://human-phenotype-ontology.github.io/downloads.html

https://www.nlm.nih.gov/medlineplus/encyclopedia.html

Hospital Charge Data:

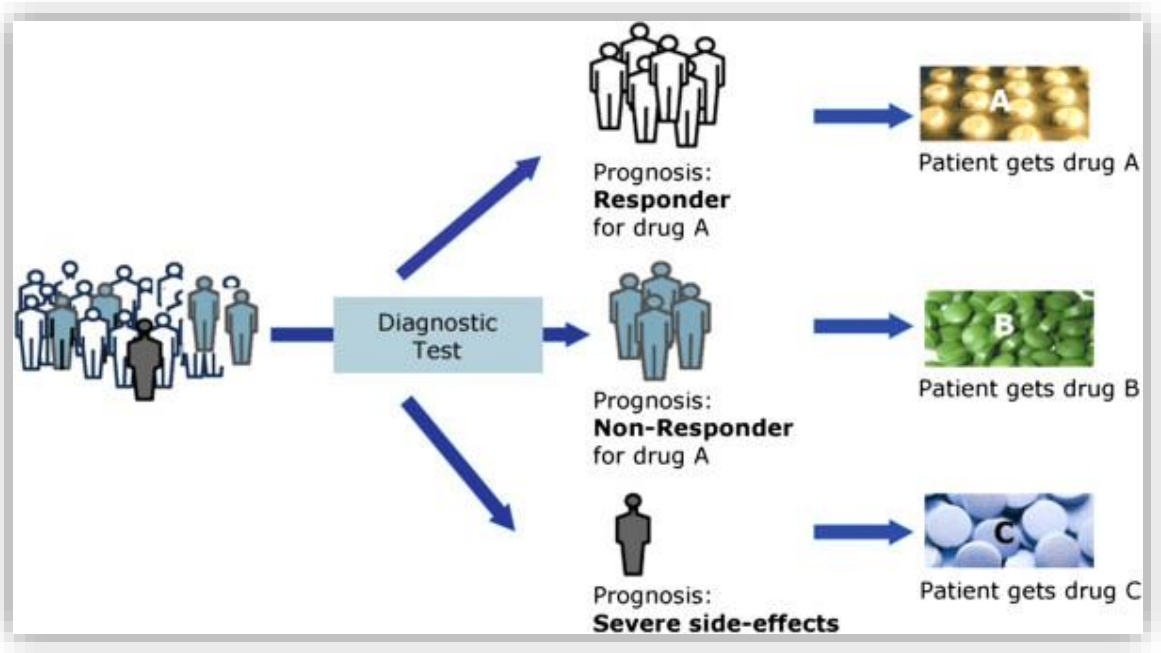
https://data.cms.gov/Medicare/Inpatient-Prospective-Payment-System-IPPS-Provider/97k6-zzx3

6 TASKS AND FEATURES

This project definitely need a lot of data and to analyze that and get to a strong knowledge base. Using that knowledge, the machine will recommend the drug. Here there will be a lot of features to be considered. For example, first the machine will analyze the symptoms and different types of diseases. This machine will take a lots of criteria into consideration like past behavior of the patient, historical data of other patients and their acceptance or rejection of the drug. Also this machine will analyze the drug allergies of a particular patient and based on the finalized knowledge created, the machine will recommend some drugs, so that the doctors can have a look and prescribe particular drugs to patients.

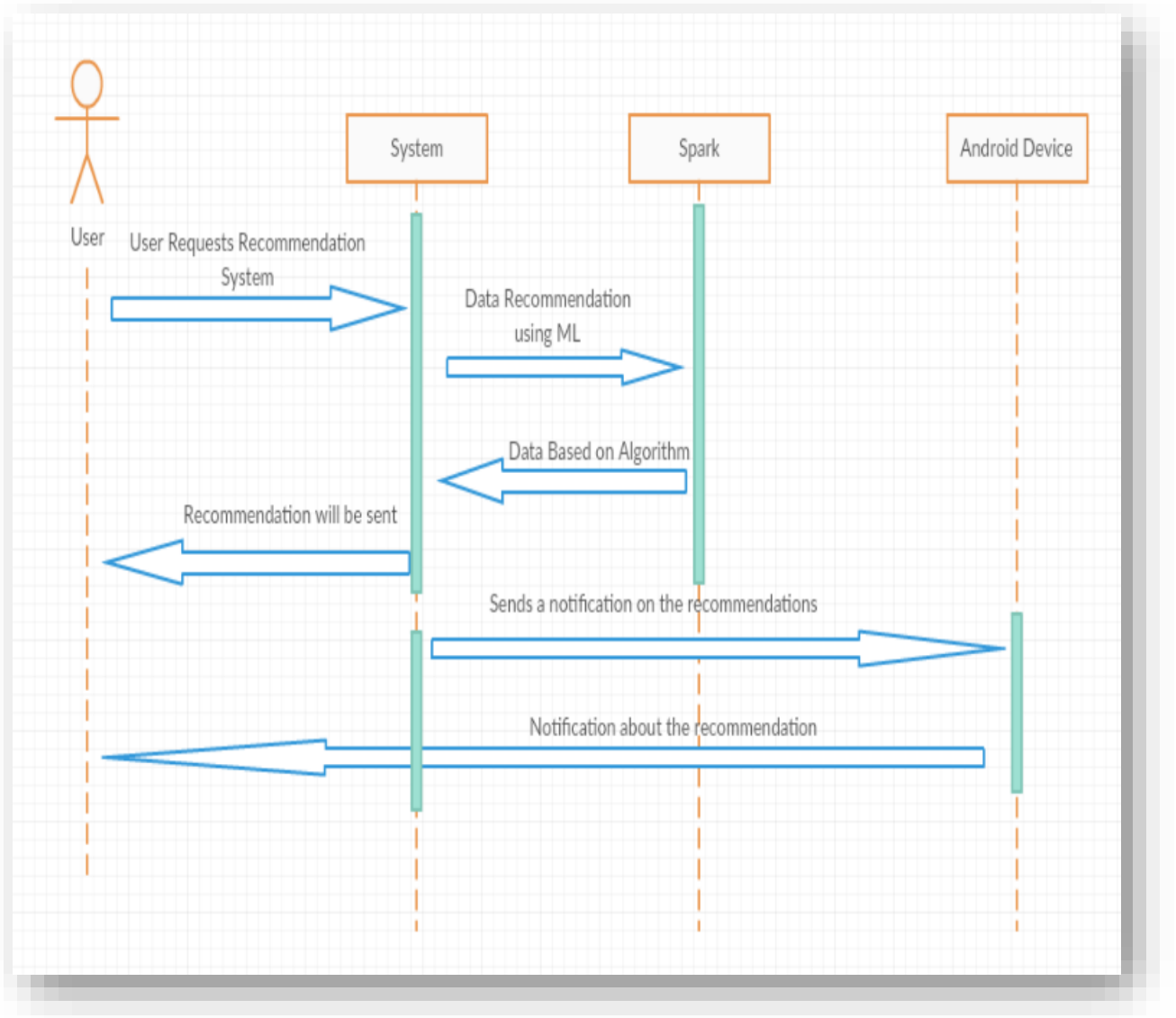
7 IMPLEMENTATION

7.1 ARCHITECTURE & UML DIAGRAMS:

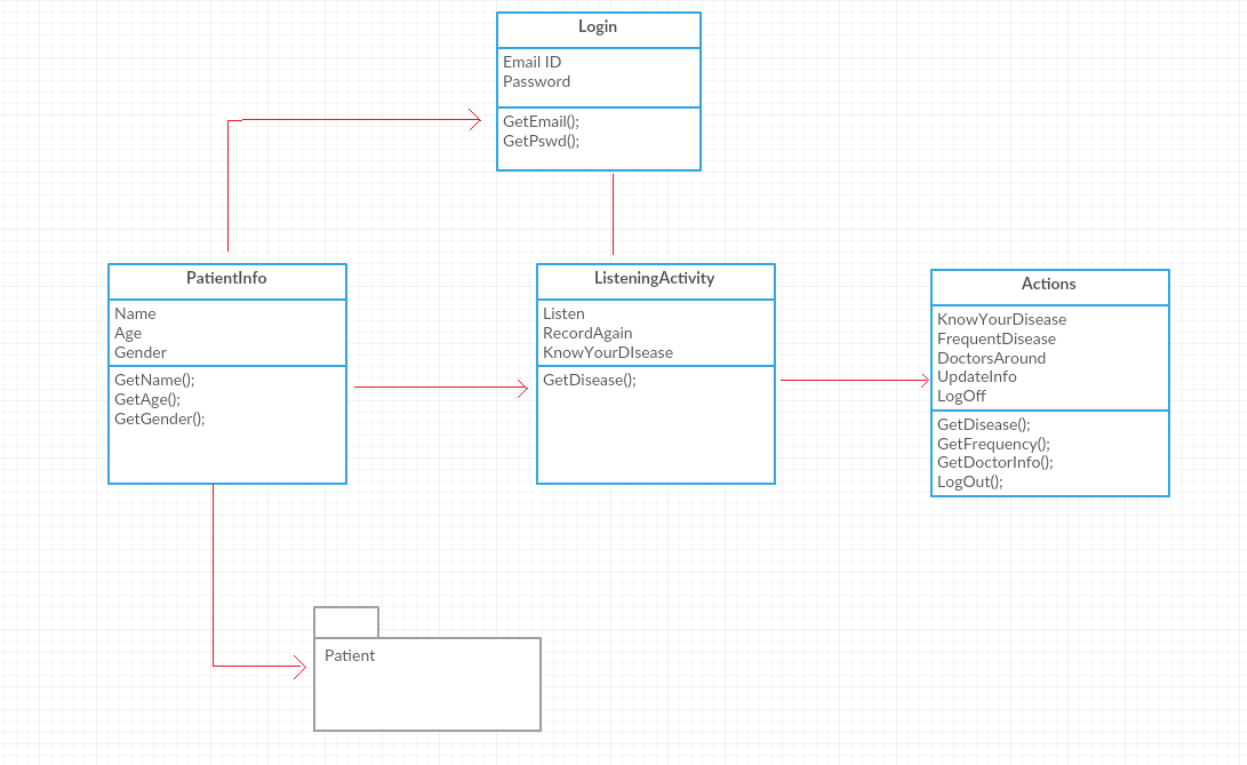


In traditional way, diagnostic test will be done first and then a drug will be used to a person where if he responds to it, then doctor will recommend to continue that drug A for better outcome, if by any reason the patient will not respond to that drug A, then doctor will recommend drug B. At-last if patient shows some severe side effects even after using drug B, then doctor will recommend drug C which is a long and time taking process. This can be changed by the above proposed approach.

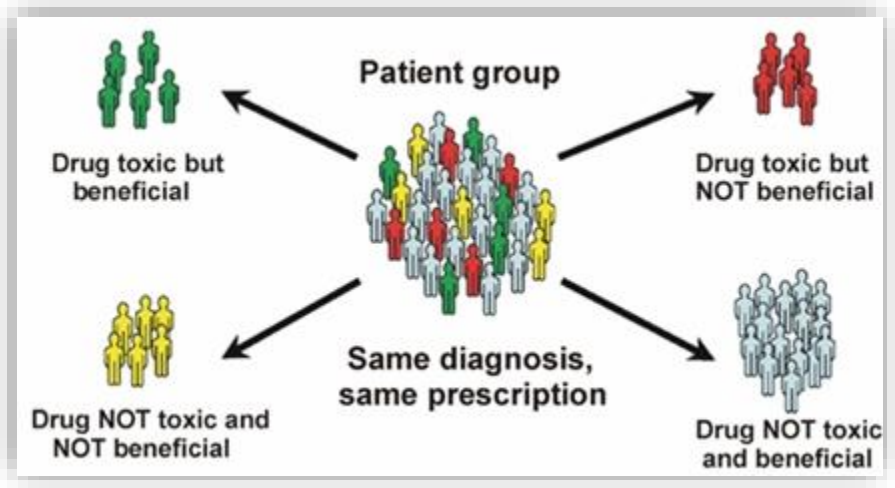
7.1.1 Sequence Diagram:



7.1.2 Class Diagram:



7.2 WORKFLOW:



Here we have a data set with group of people. Later this machine should seggregate the data to groups like drug toxic but beneficial for some time, Drug toxic but NOT beneficial, Drug Not Toxic and Not Beneficial and Drug Not Toxic and Beneficial (this group will be ideal for data collection). Based on this we will give rating to the drugs for that particular group and later we will use that data for drug recommendation.

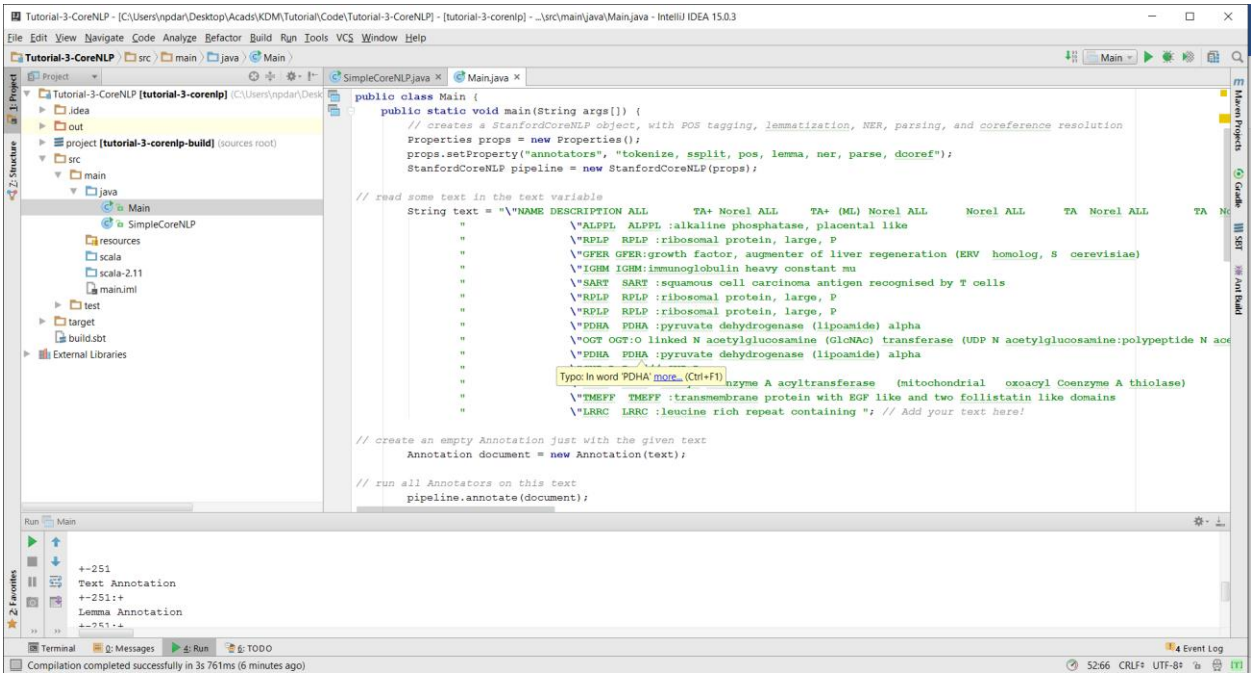
7.3 EXISTING SERVICES:

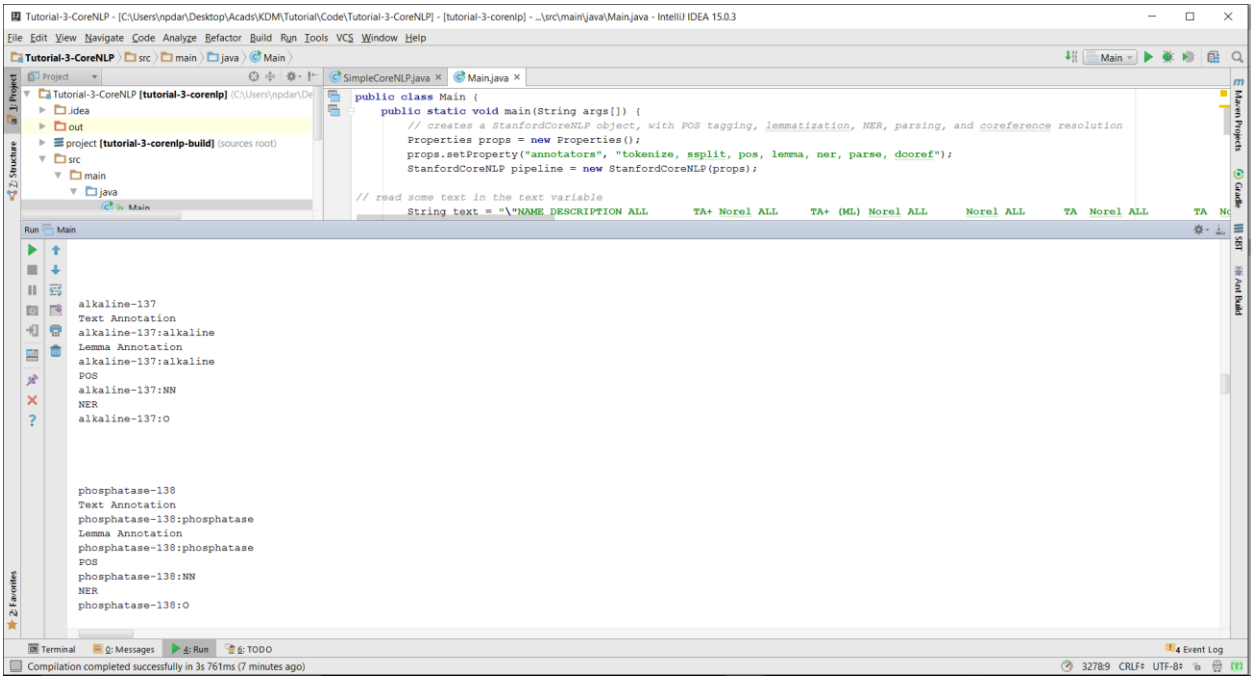
We have used the following Existing services:

Collaborative filtering (CF) is a technique that we used to filter the information and produce an output Dataset. Collaborative filtering methods have been used on the input data which include the symptoms and identifies the type of diseases. Later it is used to analyze the drugs that are prescribed for such kinds of diseases based on the given criteria.

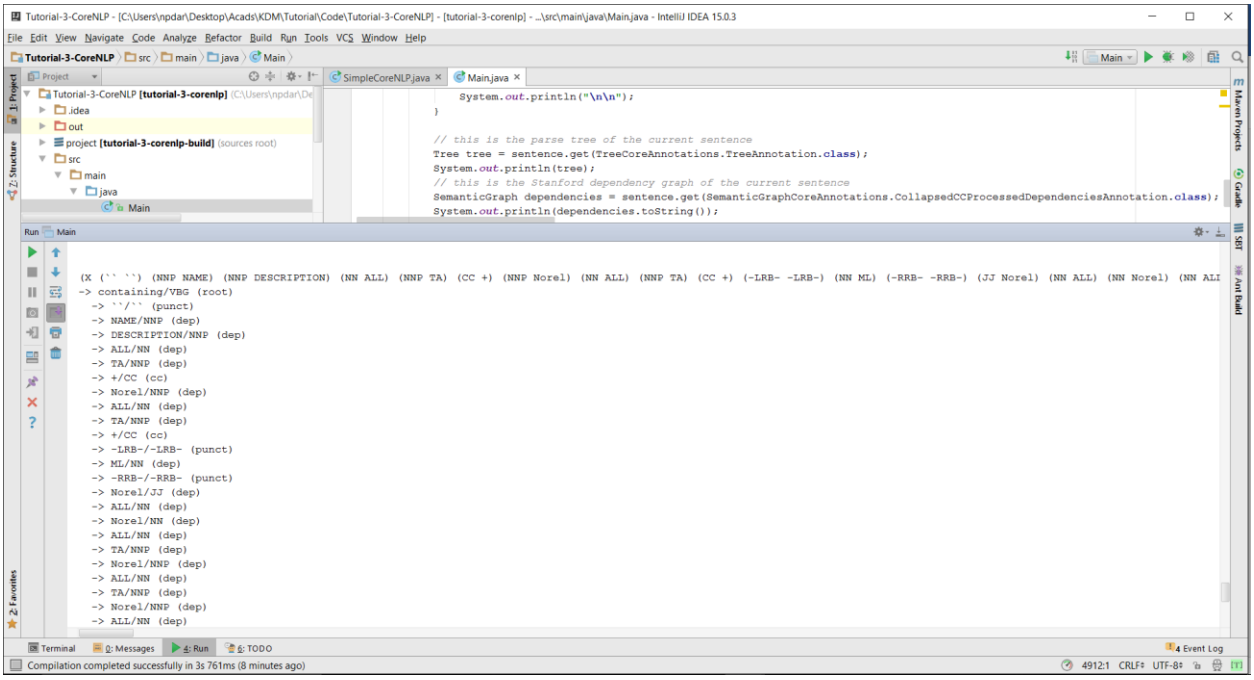
Natural Language Processing (NLP) is the technique that we have used to processes the input given by the User. Here the NLP takes the input as a symptoms and it processes the data to enable computer to derive meaning.

NLP Data provided:

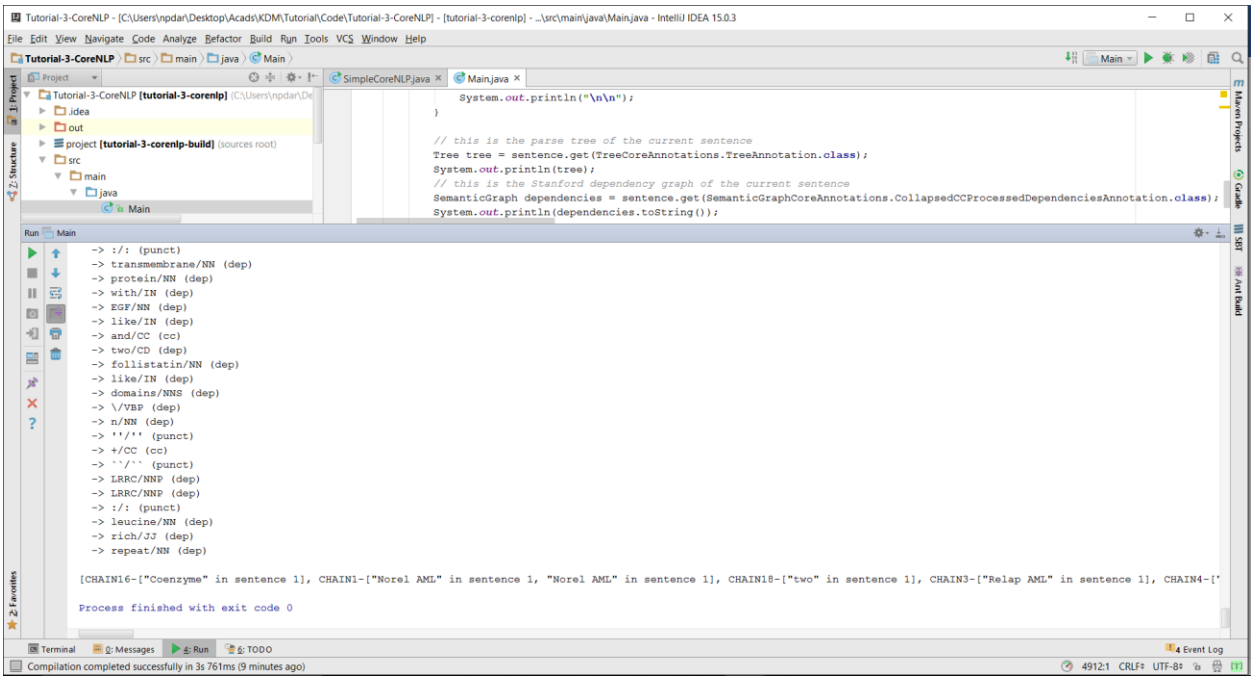



Tutorial-3-CoreNLP - [C:\Users\npdar\Desktop\Acads\KDM\Tutorial\Code\Tutorial-3-CoreNLP] - [tutorial-3-corenlp] - ...\src\main\java\Main.java - IntelliJ IDEA 15.0.3
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
SimpleCoreNLP.java
Main.java
public class Main {
public static void main(String args[]) {
// creates a StanfordCoreNLP object, with POS tagging, lemmatization, NER, parsing, and coreference resolution
Properties props = new Properties();
props.setProperty("annotators", "tokenize, ssplit, pos, lemma, ner, parse, dcoref");
StanfordCoreNLP pipeline = new StanfordCoreNLP(props);
// read some text in the text variable
String text = "\"NAME DESCRIPTION ALL TA+ Norel ALL TA+ (ML) Norel ALL Norel ALL TA Norel ALL TA
alkaline-137
Text Annotation
alkaline-137:alkaline
Lemma Annotation
alkaline-137:alkaline
POS
alkaline-137:NN
NER
alkaline-137:O
phosphatase-138
Text Annotation
phosphatase-138:phosphatase
Lemma Annotation
phosphatase-138:phosphatase
POS
phosphatase-138:NN
NER
phosphatase-138:O
Terminal
0: Messages
4: Run
6: TODO
4 Event Log
Compilation completed successfully in 3s 761ms (7 minutes ago)

Parse tree:


Tutorial-3-CoreNLP - [C:\Users\npdar\Desktop\Acads\KDM\Tutorial\Code\Tutorial-3-CoreNLP] - [tutorial-3-corenlp] - ...\src\main\java\Main.java - IntelliJ IDEA 15.0.3
System.out.println("\n\n");
// this is the parse tree of the current sentence
Tree tree = sentence.get(TreeCoreAnnotations.TreeAnnotation.class);
System.out.println(tree);
// this is the Stanford dependency graph of the current sentence
SemanticGraph dependencies = sentence.get(SemanticGraphCoreAnnotations.CollapsedCCProcessedDependenciesAnnotation.class);
System.out.println(dependencies.toString());
(X (`` ``) (NNP NAME) (NNP DESCRIPTION) (NN ALL) (NNP TA) (CC +) (NNP Norel) (NN ALL) (NNP TA) (CC +) (-LRB- -LRB-) (NN ML) (-RRB- -RRB-) (JJ Norel) (NN ALL) (NN Norel) (NN ALL
-> containing/VBG (root)
-> ``/`` (punct)
-> NAME/NNP (dep)
-> DESCRIPTION/NNP (dep)
-> ALL/NN (dep)
-> TA/NNP (dep)
-> +/CC (cc)
-> Norel/NNP (dep)
-> ALL/NN (dep)
-> TA/NNP (dep)
-> +/CC (cc)
-> -LRB-/-LRB- (punct)
-> ML/NN (dep)
-> -RRB-/-RRB- (punct)
-> Norel/JJ (dep)
-> ALL/NN (dep)
-> Norel/NN (dep)
-> ALL/NN (dep)
-> TA/NNP (dep)
-> Norel/NNP (dep)
-> ALL/NN (dep)
-> TA/NNP (dep)
-> Norel/NNP (dep)
-> ALL/NN (dep)
Terminal
0: Messages
4: Run
6: TODO
4 Event Log
Compilation completed successfully in 3s 761ms (8 minutes ago)

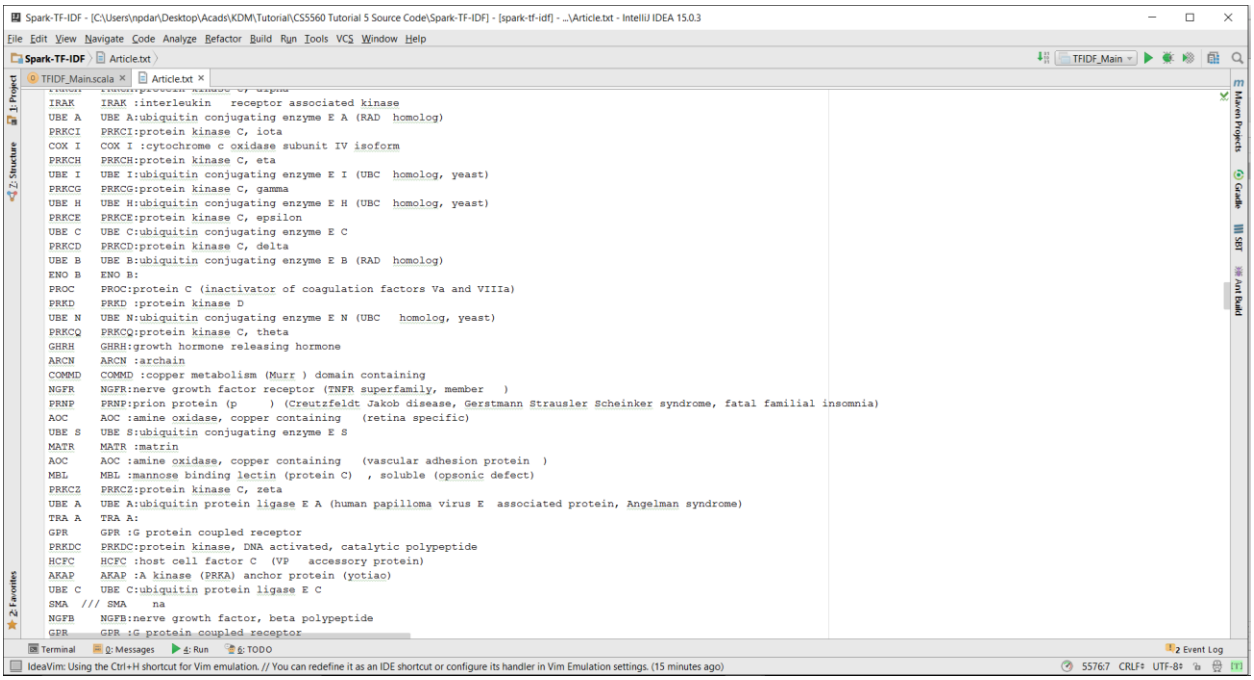
Process execution:



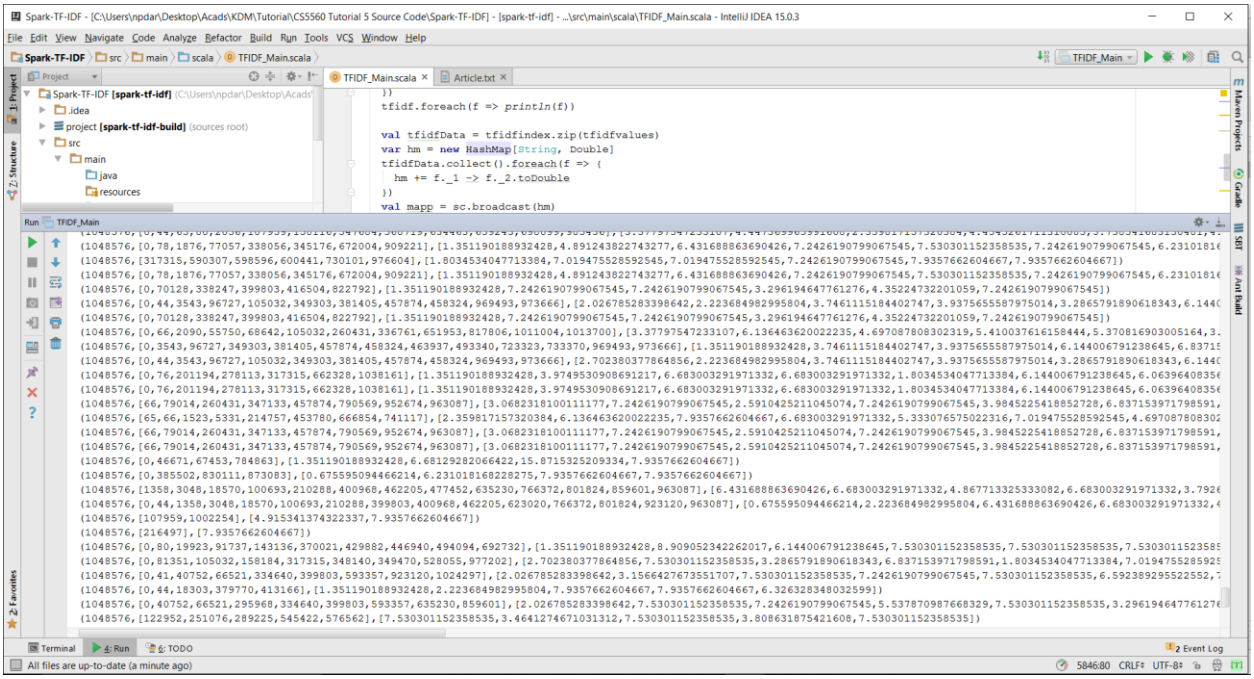
7.4 Feature Implemented:

TF-IDF: We have used TF-IDF to extract the top drugs that can be recommended to the user based on the symptoms and the criteria of the overall system. It reflects how important the word is to a document or to a dataset. Based on the provided statistics, we consider the similarities with the training data and extract the relevant top 10 drugs that can be recommended.

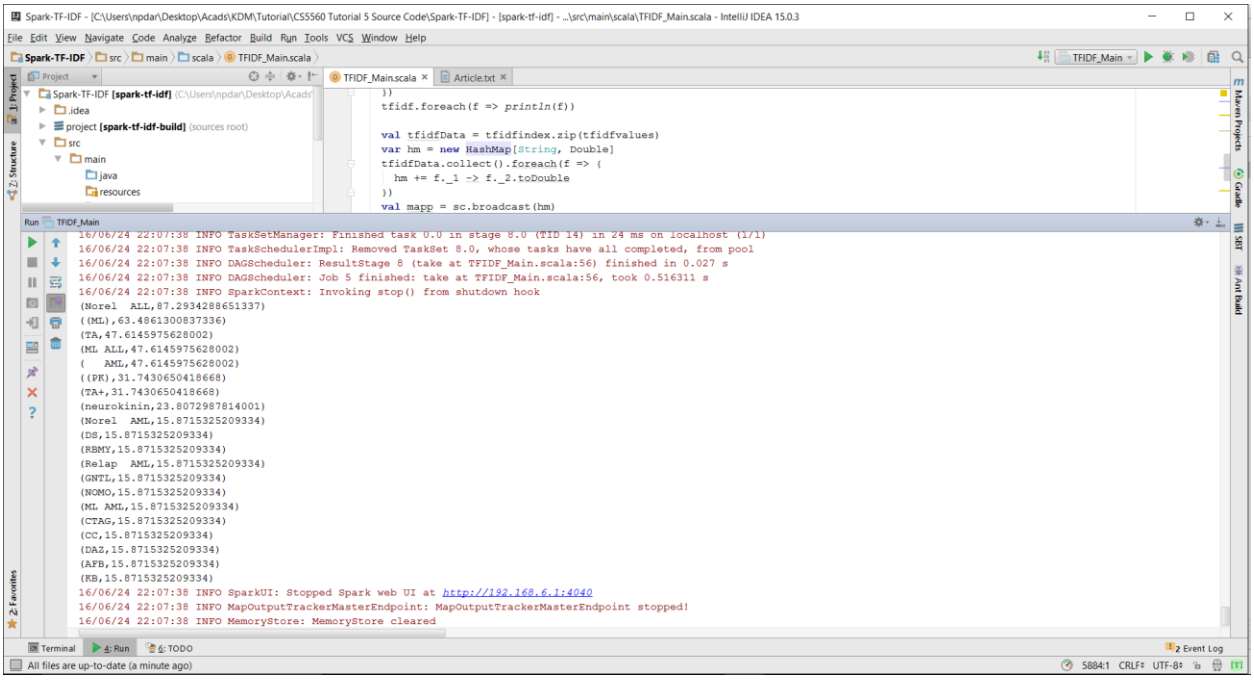
TF-IDF – Dataset:



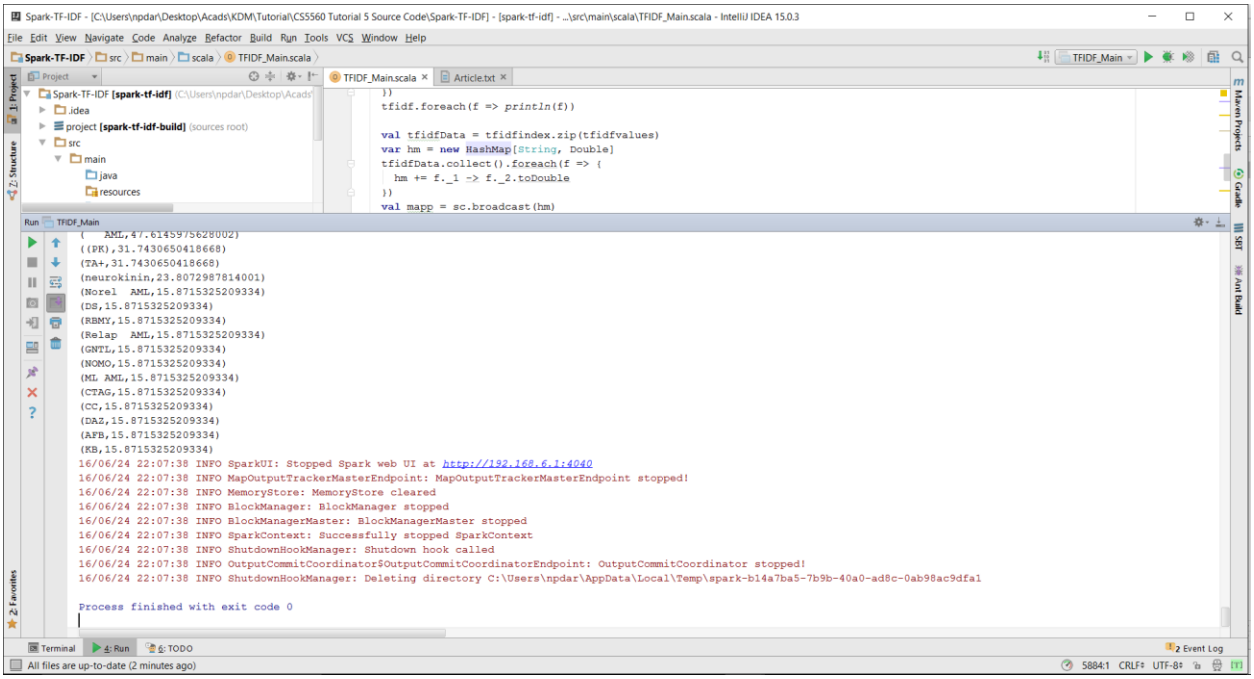
Output:



Top 10 terms:

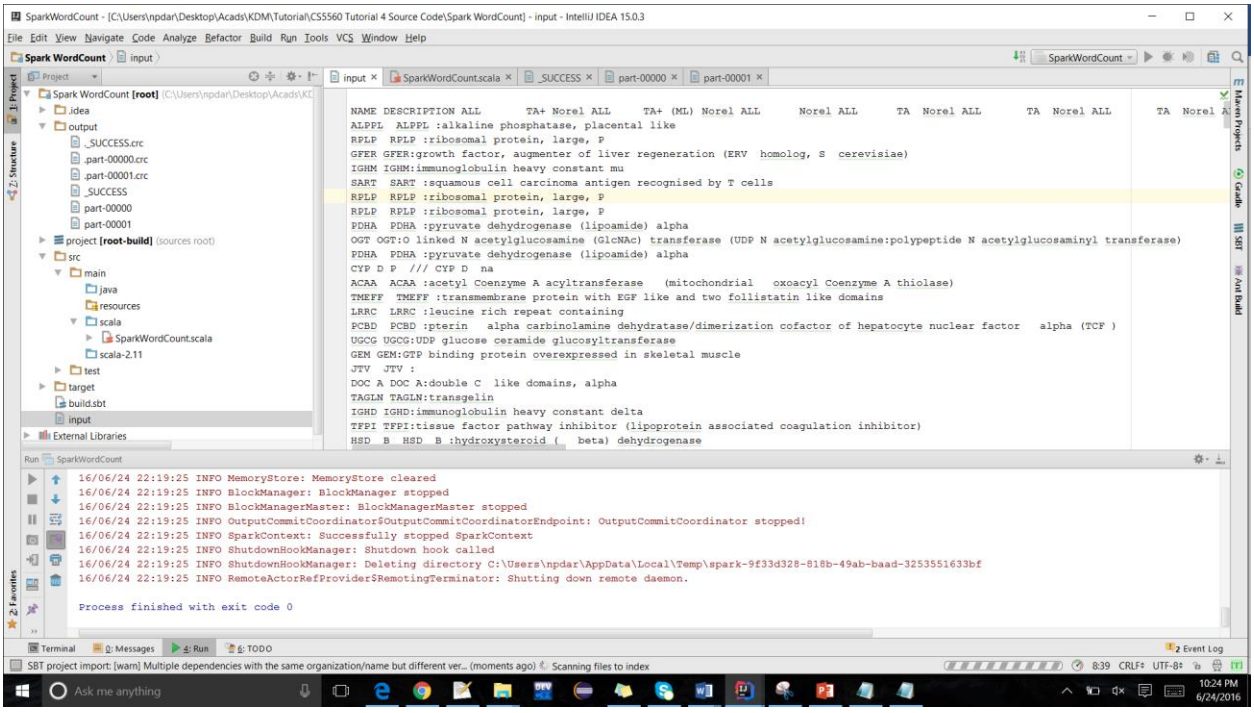


Process Success:

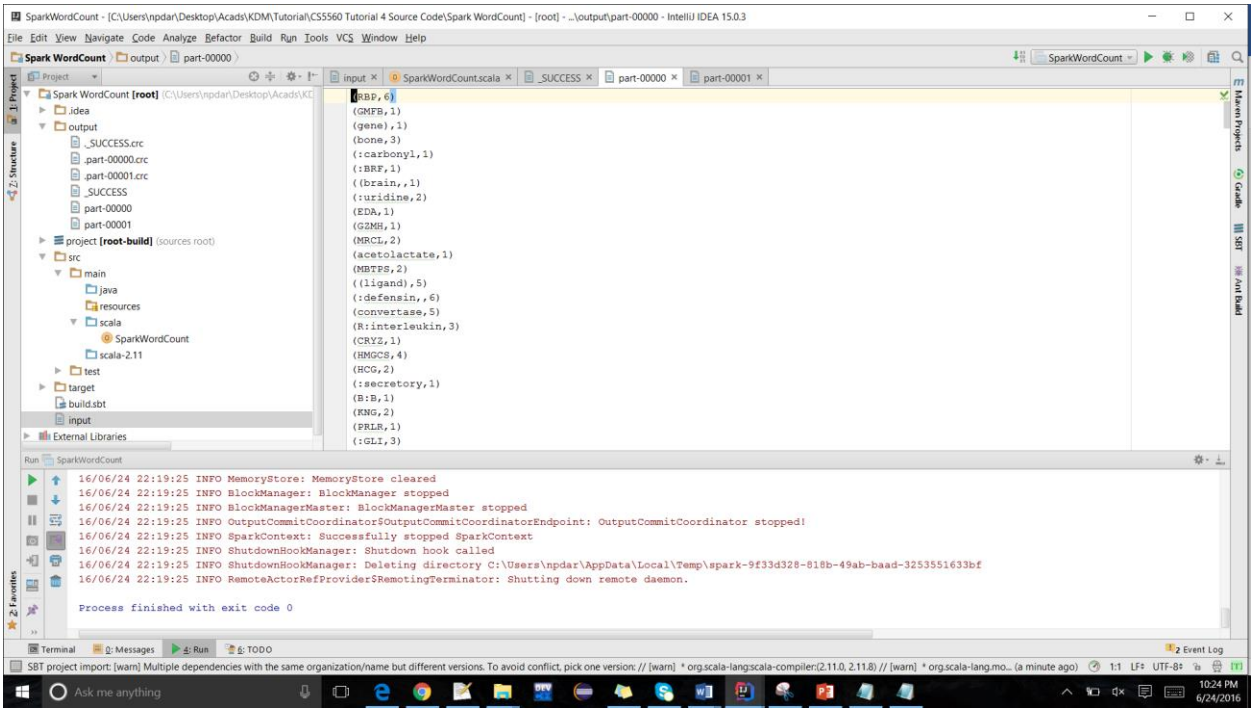


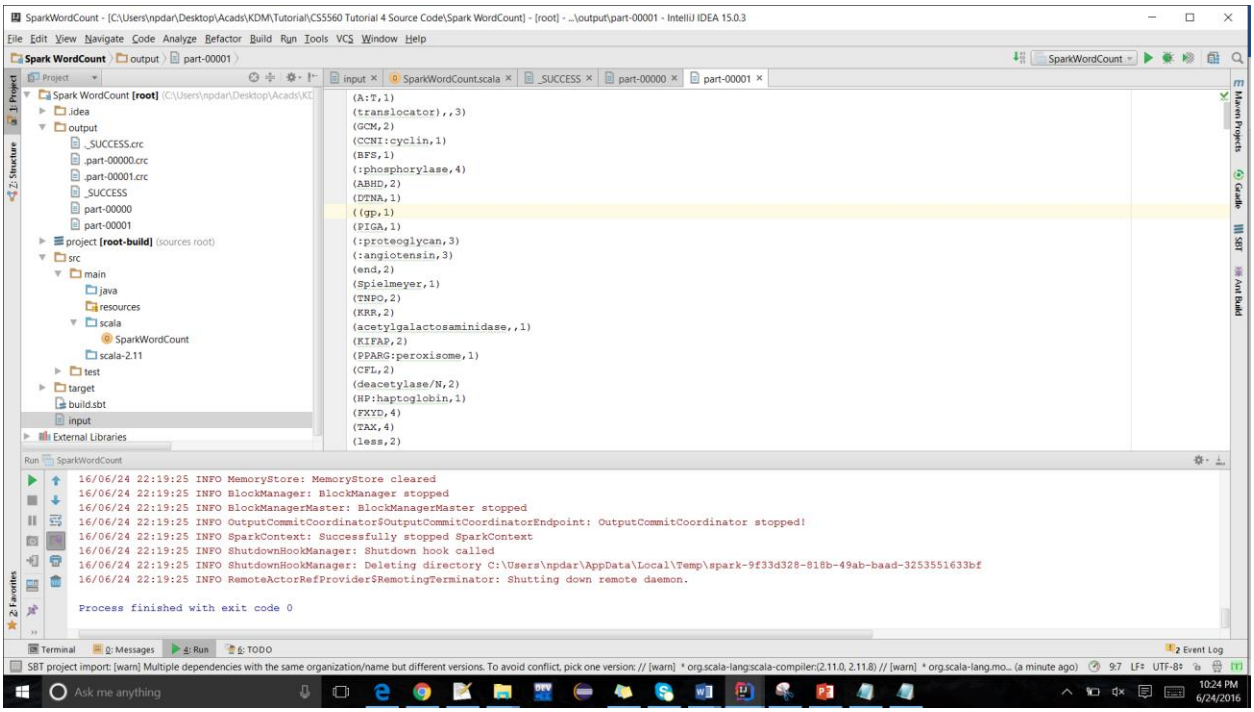
Word Count:

Input file:

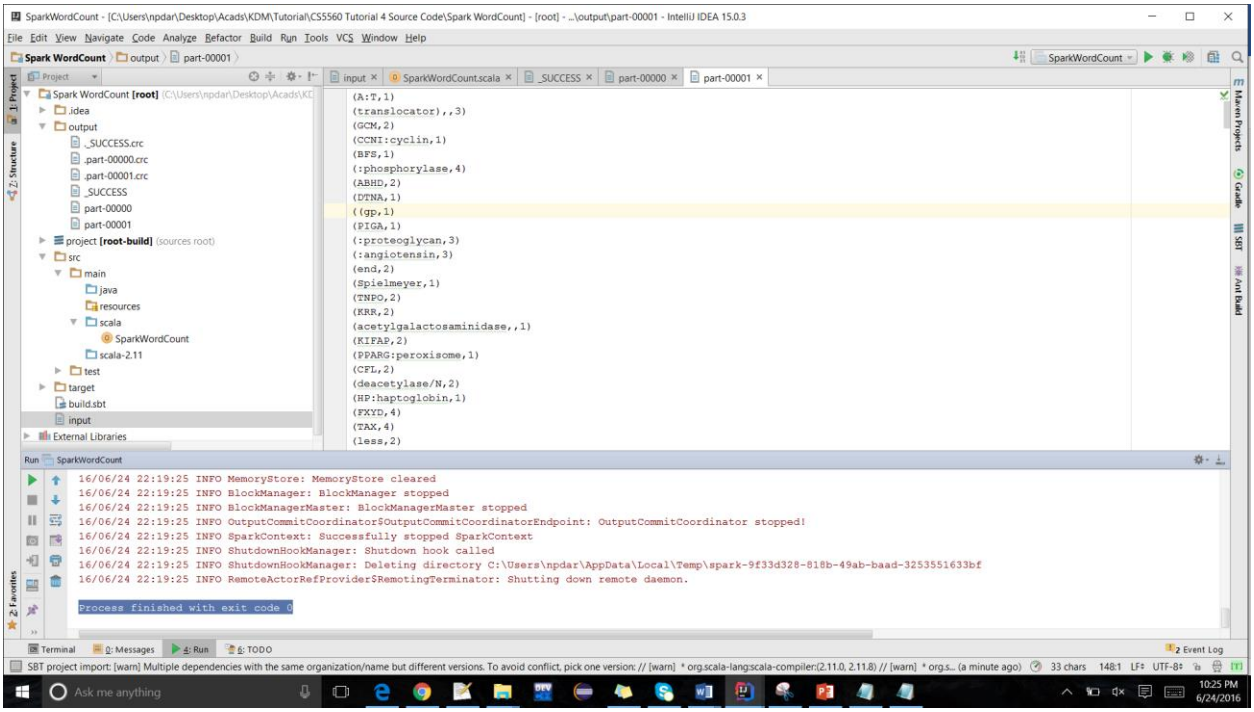


Output file:

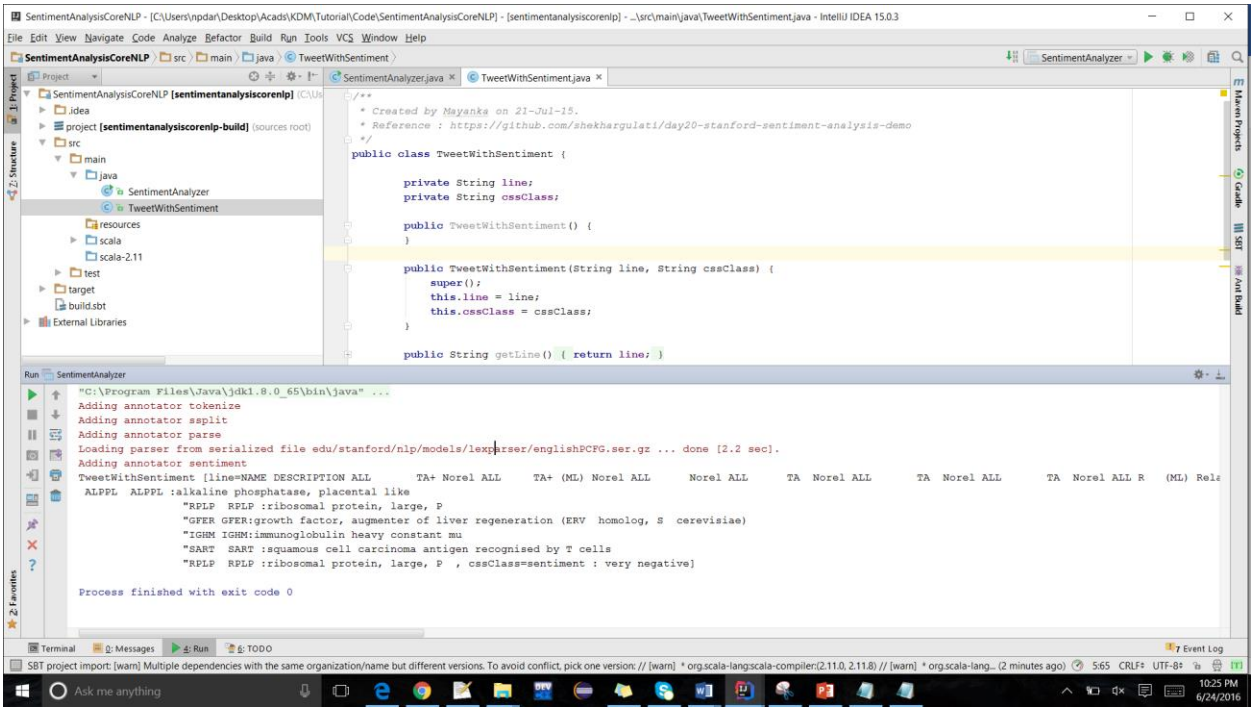




Process execution:



Sentimental Analysis:



8 PROJECT MANAGEMENT

8.1 CONTRIBUTION:

Overall – 100%

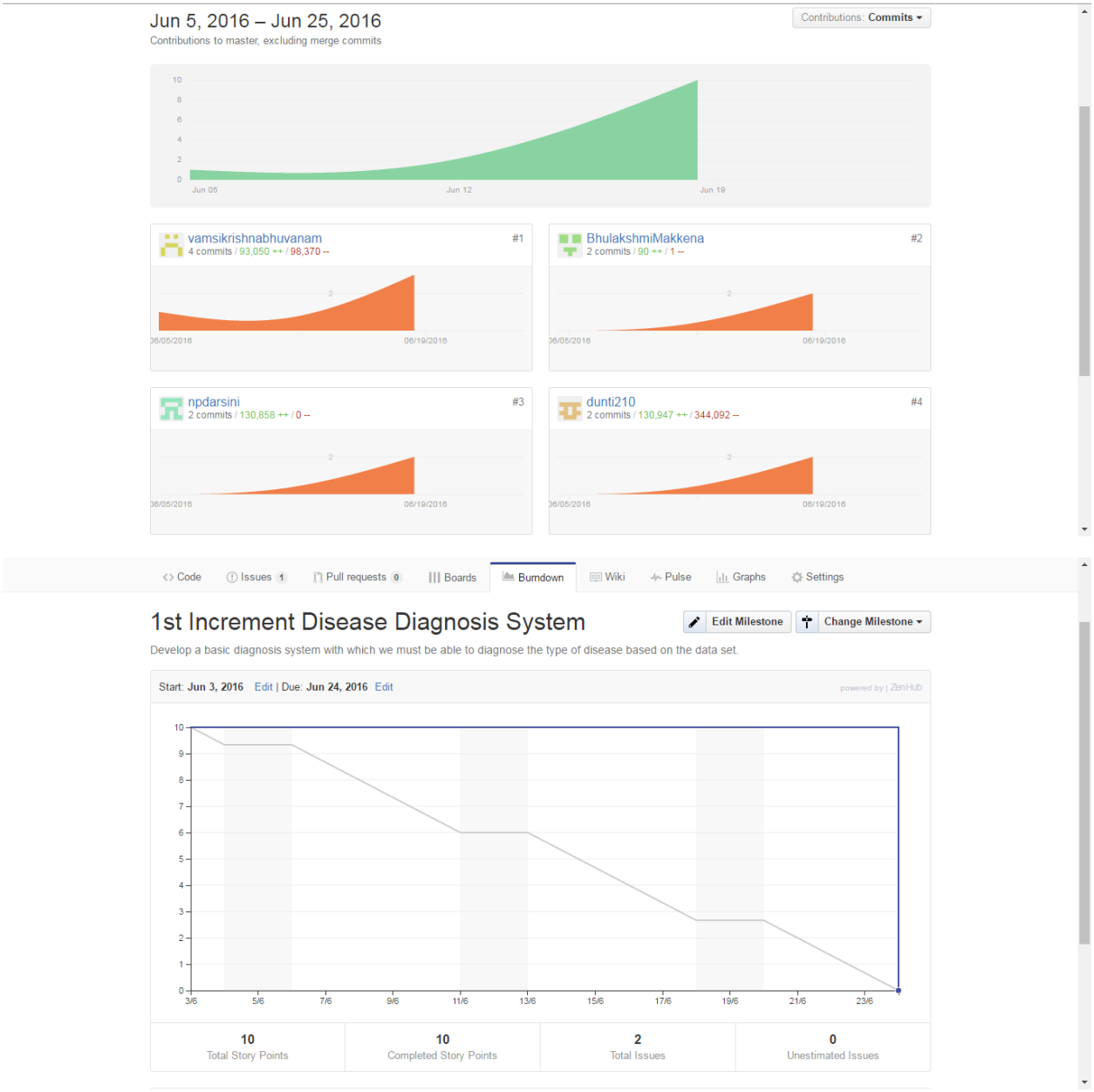
Venkata Vamsi Krishna Bhuvanam – 25%

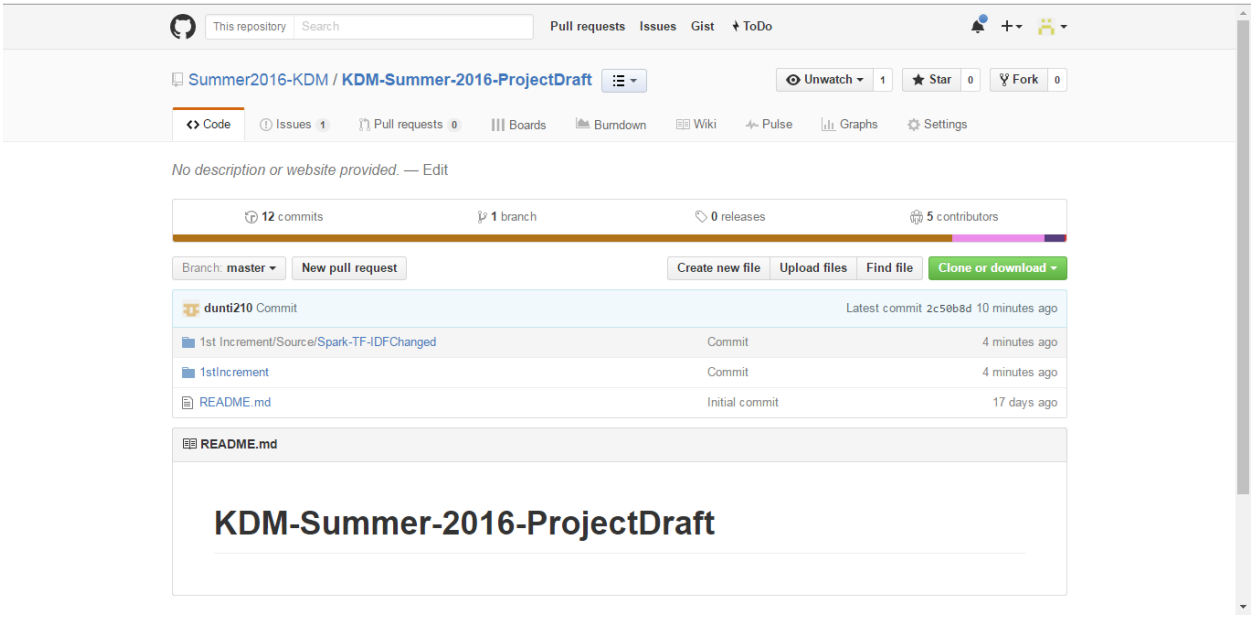
Priyadarsini Nidadavolu – 25%

Bhulakshmi Makkena – 25%

Tej Kumar Yentrapragada – 25%

8.2 ZENHUB AND GITHUB:




This repository
Search
Pull requests
Issues
Gist
ToDo
Summer2016-KDM / KDM-Summer-2016-ProjectDraft
Unwatch 1
Star 0
Fork 0
Code
Issues 1
Pull requests 0
Boards
Burndown
Wiki
Pulse
Graphs
Settings
No description or website provided. — Edit
12 commits
1 branch
0 releases
5 contributors
Branch: master
New pull request
Create new file
Upload files
Find file
Clone or download
dunti210 Commit
Latest commit 2c50b8d 10 minutes ago
1st Increment/Source/Spark-TF-IDFChanged
Commit
4 minutes ago
1stIncrement
Commit
4 minutes ago
README.md
Initial commit
17 days ago
README.md
KDM-Summer-2016-ProjectDraft

8.3 CONCERNS/ISSUES:

NA

8.4 FUTURE WORK:

Concentrating to continue to work on Drug Dataset collection and continue to build a recommendation system which will recommend drug to the user.