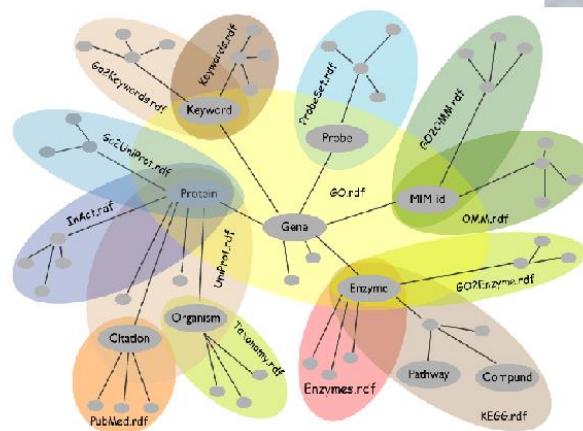


# eDrug



by Team 4 (StrikerZ):

Venkata Vamsi Krishna Bhuvanam

Priyadarsini Nidadavolu

Bhulakshmi Makkenna

Tej Kumar Yentrpragada

---

## TABLE OF CONTENTS

1	MOTIVATION .....	3
2	OBJECTIVES.....	4
3	EXPECTED OUTCOME .....	4
4	DOMAIN CHOSEN .....	5
5	DATA COLLECTION.....	6
6	TASKS AND FEATURES .....	7
7	IMPLEMENTATION.....	8
7.1	ARCHITECTURE & UML DIAGRAMS:.....	8
7.1.1	Sequence Diagram: .....	9
7.1.2	Class Diagram:.....	10
7.2	WORKFLOW: .....	10
7.3	EXISTING SERVICES:.....	11
7.3.1	First Increment:.....	11
7.3.2	Second Increment:.....	13
7.3.3	Third Increment: .....	26
7.4	FEATURE IMPLEMENTED: .....	35
8	PROJECT MANAGEMENT .....	40
8.1	CONTRIBUTION: .....	40
8.2	ZENHUB AND GITHUB:.....	41
8.2.1	First Increment:.....	41
8.2.2	SECOND INCREMENT: .....	43
8.2.3	THIRD INCREMENT.....	44
8.3	CONCERNS/ISSUES:.....	44

---

---

8.4	FUTURE WORK: .....	45
8.5	REFERENCES: .....	45

## **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/97k6zzx3>

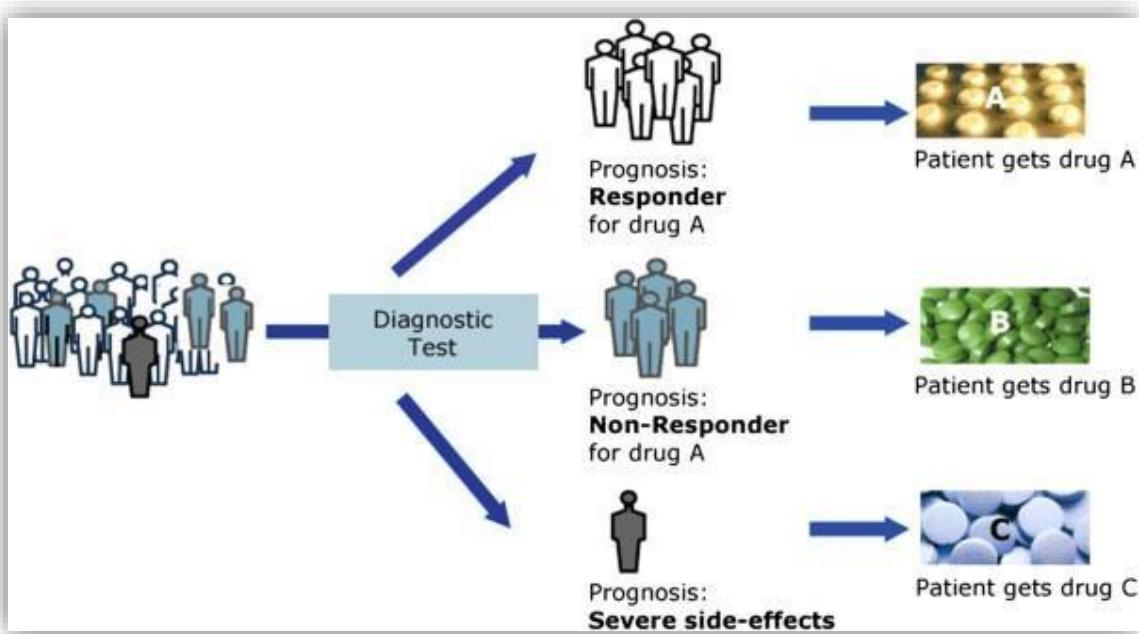
## **6 TASKS AND FEATURES**

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.

---

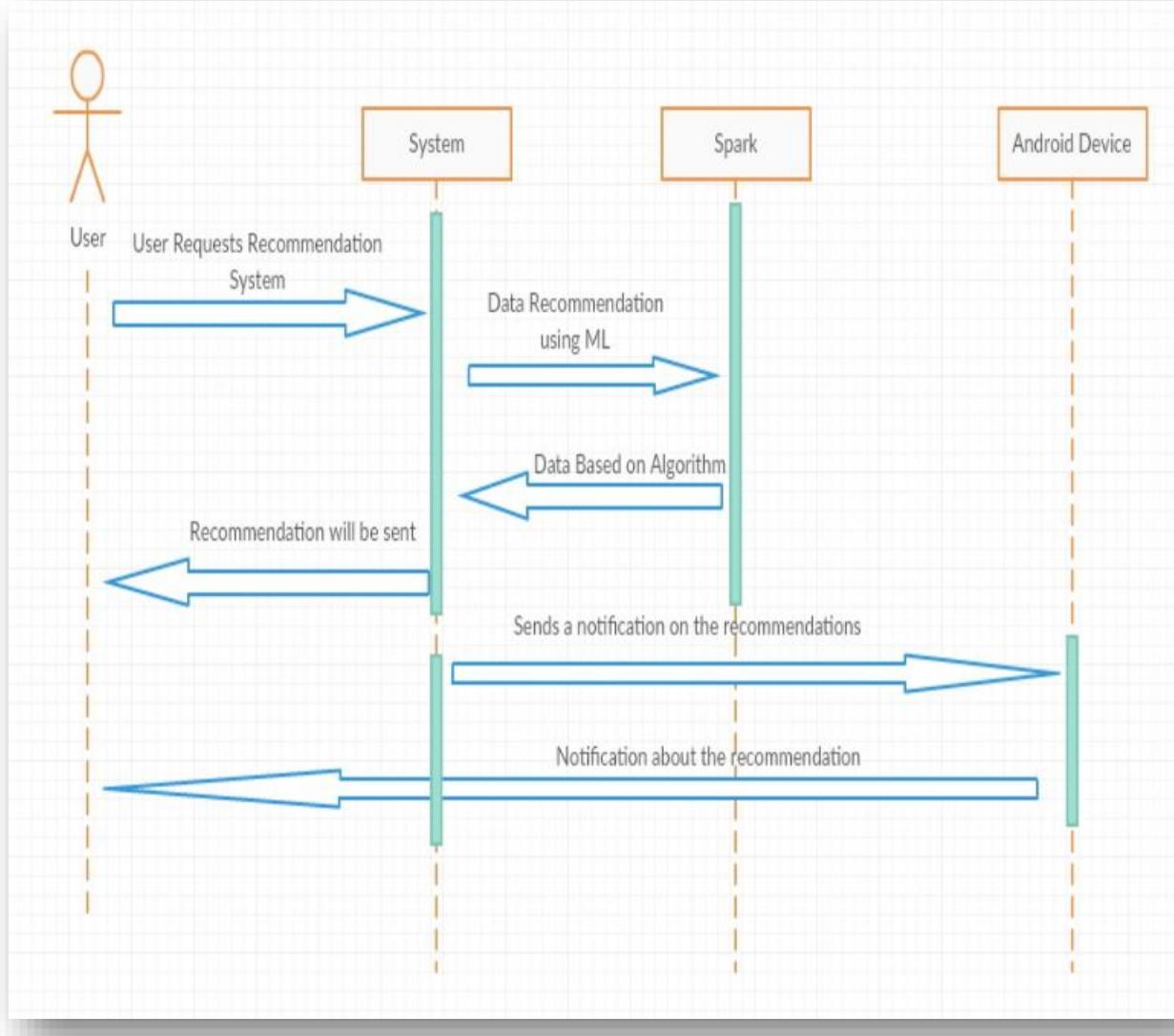
## 7 IMPLEMENTATION

### 7.1 ARCHITECTURE & UML DIAGRAMS:

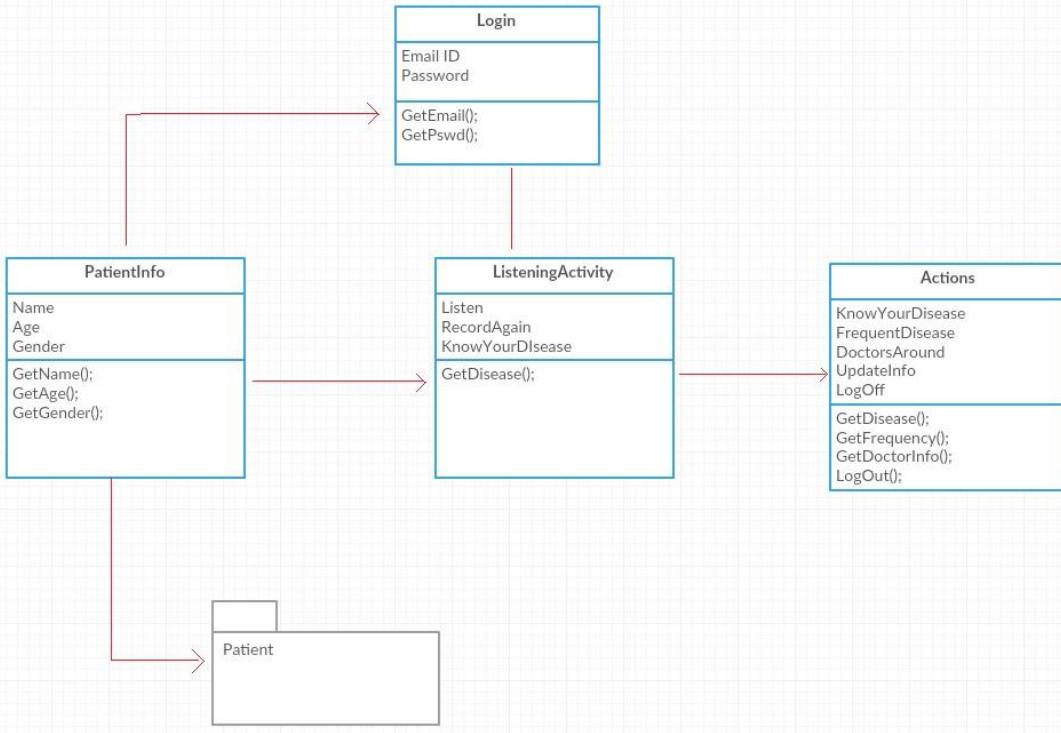


In traditional way, diagnostic test will be done first and then a drug will be used to a person where if he responds to it, then doctor will recommend to continue that drug A for better outcome, if by any reason the patient will not respond to that drug A, then doctor will recommend drug B. At-last if 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.

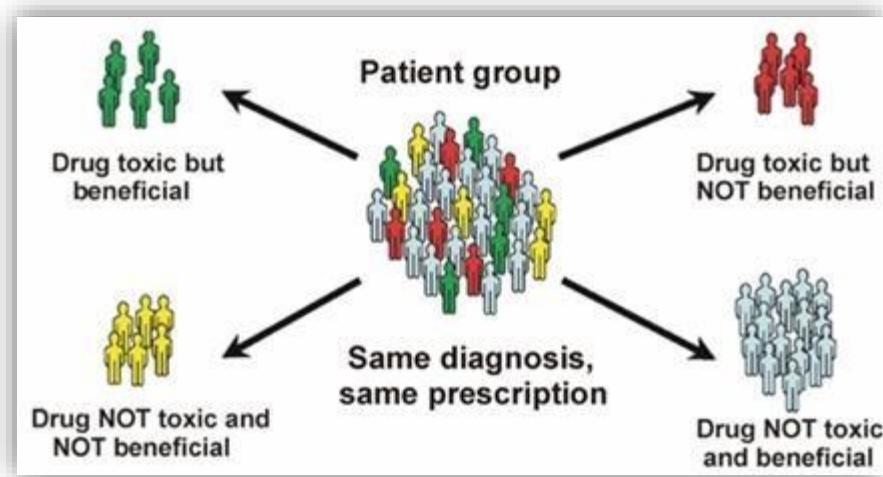
## 7.1.1 Sequence Diagram:



### 7.1.2 Class Diagram:



### 7.2 WORKFLOW:



Here we have a data set with group of people. Later this machine should segregate 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:

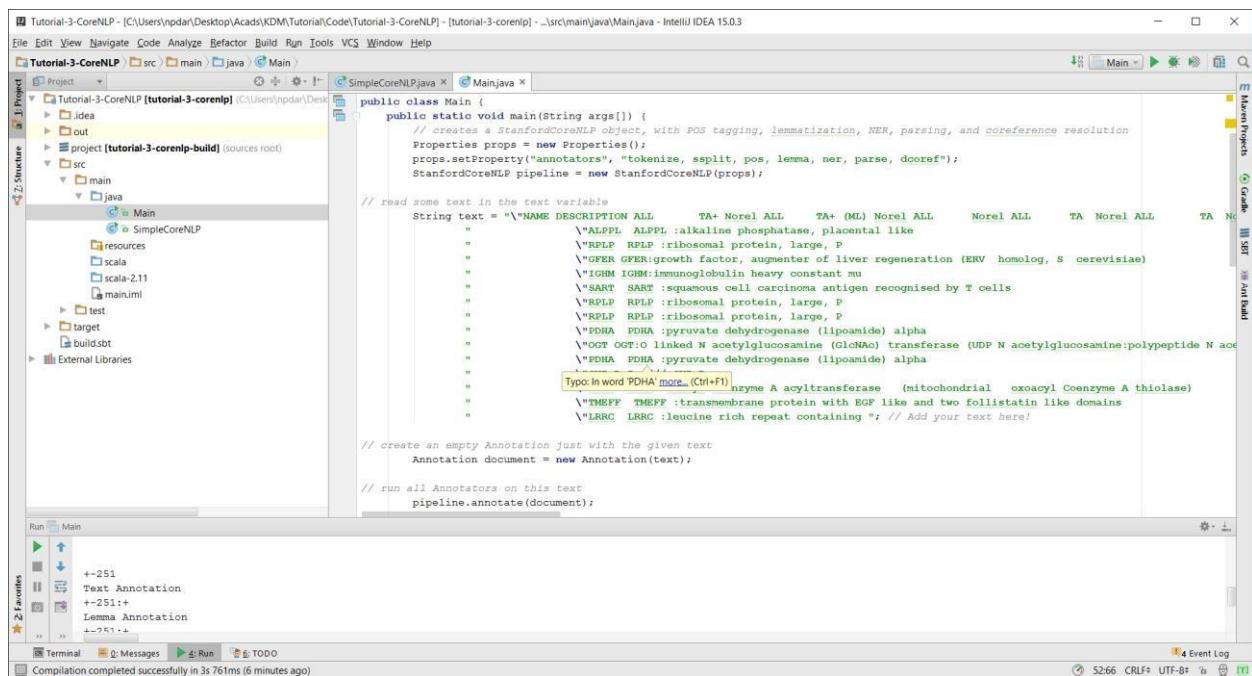
### 7.3.1 First Increment:

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:



The screenshot shows the IntelliJ IDEA 15.0.3 interface with the following details:

- File Path:** Tutorial-3-CoreNLP - [C:\Users\ndan\Desktop\Acads\KDM\Tutorial\Code\Tutorial-3-CoreNLP] - [tutorial-3-corenlp] - \src\main\java\Main.java
- Code Editor:** Shows the Main.java file with Java code for Stanford CoreNLP annotation. The code reads text from a string and uses a StanfordCoreNLP pipeline to annotate it. A tooltip appears over the word "PDHA", suggesting it is a mitochondrial enzyme.
- Project Structure:** Shows the project structure with modules like idea, out, src, and main containing Java, resources, scala, and build.sbt files.
- Toolbars and Menus:** Standard IntelliJ menus like File, Edit, View, Navigate, Code, Analyze, Refactor, Build, Run, Tools, VCS, Window, Help.
- Bottom Status Bar:** Shows compilation status: "Compilation completed successfully in 3s 761ms (6 minutes ago)" and file encoding: "5266 CRLF# UTF-8".

## Project Report Increment 2

The screenshot shows the IntelliJ IDEA interface with the following details:

- Project Structure:** The project is named "Tutorial-3-CoreNLP" located at "C:\Users\vpndan\Desktop\Acads\KDM\Tutorial\Code\Tutorial-3-CoreNLP".
- Code Editor:** The file "SimpleCoreNLP.java" is open, containing Java code for a "Main" class that uses the StanfordCoreNLP pipeline.
- Output Window:** The bottom-left window displays the processed text for two input examples. The first example ("alkaline") shows annotations for Text Annotation, Lemma Annotation, POS, NER, and coreference resolution. The second example ("phosphatase") shows similar annotations.
- Terminal:** The bottom-left terminal window shows the compilation message: "Compilation completed successfully in 3s 761ms (7 minutes ago)".
- Event Log:** The bottom-right event log window shows the message "32789 CRLF# UTF-8# 'b'".

## Parse tree:

The screenshot shows the IntelliJ IDEA interface with the following details:

- Project Structure:** The project is named "Tutorial-3-CoreNLP". It contains a "src" directory with a "main" package, which in turn contains a "java" folder and a "Main" class.
- Main.java Content:** The code prints two types of annotations:
  - A parse tree using `TreeCoreAnnotations.TreeAnnotation`.
  - A Stanford dependency graph using `SemanticGraphCoreAnnotations.CollapsedCCProcessedDependenciesAnnotation`.
- Run Tab:** The "Main" class is selected for execution.
- Toolbars and Menus:** Standard IntelliJ IDEA menus like File, Edit, View, Navigate, Code, Analyze, Refactor, Build, Run, Tools, VCS, Window, Help are visible at the top.
- Bottom Status Bar:** Shows compilation status: "Compilation completed successfully in 3s 761ms (8 minutes ago)".
- Event Log:** An "Event Log" tab is open in the bottom right corner.

## Process execution:

The screenshot shows the IntelliJ IDEA interface with the following details:

- Project Structure:** The project is named "Tutorial-3-CoreNLP". It contains a "src" directory with "main" and "java" sub-directories. The "java" directory contains a file named "Main.java".
- Code Editor:** The "Main.java" file is open, displaying Java code that prints parse trees and dependency graphs for a sentence. The code includes imports for `TreeAnnotations`, `SemanticGraphCoreAnnotations`, and `CollapsedCCProcessedDependenciesAnnotation` from the Stanford NLP library.
- Terminal Output:** The terminal shows the execution of the application. The output includes a list of dependencies (e.g., `:/ (punct)`, `transmembrane/NN (dep)`), followed by the message "[CHAIN16-[Coenzyme" in sentence 1], CHAIN1-[Norel AML" in sentence 1, "Norel AML" in sentence 1], CHAIN18-[two" in sentence 1], CHAIN3-[Relap AML" in sentence 1], CHAIN4-[` and the message "Process finished with exit code 0".
- Event Log:** The event log shows a single entry: "Compilation completed successfully in 3s 761ms (9 minutes ago)".

### 7.3.2 Second Increment: WordNet on top TFIDF words:

## Project Report Increment 2

SparkOpenE - [C:\Users\ndpda\Desktop\Acads\KDM\Tutorial\CS5560 - Tutorial 7 Source Code\SparkOpenE] - [sparkopenie] - \src\main\scala\wordnet\WordNetMain.java - IntelliJ IDEA 15.0.3

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

SparkOpenE src main scala wordnet WordNetMain

Project Structure

```

String start = "medicine";
String end = "drug";
pos = wordnet.getBestPos(start);

// Wordnet can find relationships between words
System.out.println("\n\nRelationship between: " + start + " and " + end);
float dist = wordnet.getDistance(start,end,pos);
String[] parents = wordnet.getCommonParents(start, end, pos);
System.out.println(start + " and " + end + " are related by a distance of: " + dist);

// These words have common parents (hyponyms in this case)
System.out.println("Common parents: ");
if (parents != null) {
    for (int i = 0; i < parents.length; i++) {
        System.out.println(parents[i]);
    }
}

//System.out.println("\n\nHypernym Tree for " + start);
//Int[] ids = wordnet.getSenseIds(start,wordnet.NOUN);
//wordnet.printHypernymtree(ids[0]);

```

Run WordNetMain

```

Hyponyms for million:
|_
|   Hypernyms for million:
|   |_
|   |   large integer
|   |_
|   |   large indefinite quantity
|   |_
|   |   large indefinite amount

Relationship between: medicine and drug
medicine and drug are related by a distance of: 0.16666667
Common parents:
entity

Process finished with exit code 0

```

Terminal Messages Run TODO

Compilation completed successfully in 6s 971ms (a minute ago)

Event Log 187:1 CRLF: UTF-8: 3:49 PM 7/8/2016

Ask me anything

SparkOpenE - [C:\Users\ndpda\Desktop\Acads\KDM\Tutorial\CS5560 - Tutorial 7 Source Code\SparkOpenE] - [sparkopenie] - \src\main\scala\wordnet\WordNetMain.java - IntelliJ IDEA 15.0.3

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

SparkOpenE src main scala wordnet WordNetMain

Project Structure

```

String start = "medicine";
String end = "drug";
pos = wordnet.getBestPos(start);

// Wordnet can find relationships between words
System.out.println("\n\nRelationship between: " + start + " and " + end);
float dist = wordnet.getDistance(start,end,pos);
String[] parents = wordnet.getCommonParents(start, end, pos);

```

Run WordNetMain

```

"C:\Program Files\Java\jdk1.8.0_65\bin\java" ...
Finding parts of speech for page.
page
pagev

Definitions for patent:
patentobtain a patent for
patentgrant rights to; grant a patent for
patentmake open to sight or notice

Synonyms for next (pos: a)
next - adjacent
next - close
next - following
next - future
next - incoming
next - side by side
next - succeeding

Synonyms for next (pos: r)

Synonyms for page (pos: n)
page - Page
page - Sir Frederick Handley Page
page - Thomas Nelson Page
page - arborist
page - baggageman

```

Terminal Messages Run TODO

Compilation completed successfully in 6s 971ms (a minute ago)

Event Log 189:14 CRLF: UTF-8: 3:50 PM 7/8/2016

Ask me anything

## Project Report Increment 2

The screenshot shows the IntelliJ IDEA interface with the project 'SparkOpenE' open. The code editor displays Java code related to wordnet. A list of word synonyms is shown in the 'Run' tool window:

```
String start = "medicine";
String end = "drug";
pos = wordnet.getBestPos(start);

// Wordnet can find relationships between words
```

Synonyms for page (pos: v)

- page - assist
- page - beaver
- page - beaver away
- page - beacon
- page - boondoggle
- page - busy
- page - buzz
- page - carpenter
- page - clerk
- page - collaborate

Synonyms for doctor (pos: n)

- doctor - Arabist
- doctor - Cabalist
- doctor - Church Father
- doctor - Doctor
- doctor - Doctor of the Church
- doctor - Dr.
- doctor - Father
- doctor - Father of the Church
- doctor - GP
- doctor - Islamist

Synonyms for doctor (pos: v)

- doctor - administer
- doctor - advance
- doctor - aid
- doctor - alleviate
- doctor - analyse

The screenshot shows the IntelliJ IDEA interface with the project 'SparkOpenE' open. The code editor displays Java code related to wordnet. A list of word synonyms is shown in the 'Run' tool window:

```
String start = "medicine";
String end = "drug";
pos = wordnet.getBestPos(start);

// Wordnet can find relationships between words
```

Synonyms for currently (pos: r)

- currently - presently

Synonyms for change (pos: n)

- change - Coriolis effect
- change - Maundy money
- change - about-face
- change - acceleration
- change - accenting
- change - accentuation
- change - accessory
- change - accident
- change - accompaniment
- change - accomplishment

Synonyms for change (pos: v)

- change - Americanise
- change - Americanize
- change - Europeanise
- change - Europeanize
- change - Frenchify
- change - Islamise
- change - Islamize
- change - accelerate
- change - accommodate
- change - accompany

Synonyms for rising (pos: a)

- rising - acclivitous

## Project Report Increment 2

```
String start = "medicine";
String end = "drug";
pos = wordnet.getBestPos(start);

// Wordnet can find relationships between words
Synonyms for brand (pos: v)
brand - address
brand - asterisk
brand - badge
brand - bespot
brand - brandmark
brand - calibrate
brand - call
brand - code
brand - crisscross
brand - delineate

Synonyms for mg (pos: n)
mg - Al
mg - Am
mg - Ba
mg - Be
mg - Bi
mg - Bk
mg - Ca
mg - Cd
mg - Ce
mg - Cf

Synonyms for percentage (pos: n)
percentage - absentee rate
percentage - accounts receivable
percentage - allocation
percentage - allotment

Synonyms for mg (pos: n)
mg - Cf

Synonyms for percentage (pos: n)
percentage - absentee rate
percentage - accounts receivable
percentage - allocation
percentage - allotment
percentage - allowance
percentage - amount
percentage - amount of money
percentage - batting average
percentage - capital
percentage - case-fatality proportion

Hyponyms for million:

Hypernyms for million:
large integer
large indefinite quantity
large indefinite amount

Relationship between: medicine and drug
medicine and drug are related by a distance of: 0.16666667
Common parents:
entity

Process finished with exit code 0
```

```
String start = "medicine";
String end = "drug";
pos = wordnet.getBestPos(start);

// Wordnet can find relationships between words
Synonyms for brand (pos: v)
brand - address
brand - asterisk
brand - badge
brand - bespot
brand - brandmark
brand - calibrate
brand - call
brand - code
brand - crisscross
brand - delineate

Synonyms for mg (pos: n)
mg - Al
mg - Am
mg - Ba
mg - Be
mg - Bi
mg - Bk
mg - Ca
mg - Cd
mg - Ce
mg - Cf

Synonyms for percentage (pos: n)
percentage - absentee rate
percentage - accounts receivable
percentage - allocation
percentage - allotment
percentage - allowance
percentage - amount
percentage - amount of money
percentage - batting average
percentage - capital
percentage - case-fatality proportion

Synonyms for mg (pos: n)
mg - Cf

Synonyms for percentage (pos: n)
percentage - absentee rate
percentage - accounts receivable
percentage - allocation
percentage - allotment
percentage - allowance
percentage - amount
percentage - amount of money
percentage - batting average
percentage - capital
percentage - case-fatality proportion

Hyponyms for million:

Hypernyms for million:
large integer
large indefinite quantity
large indefinite amount

Relationship between: medicine and drug
medicine and drug are related by a distance of: 0.16666667
Common parents:
entity

Process finished with exit code 0
```

### Word to Vector using MLLIB:

Word 2 Vec takes input as a text/Json corpus and gives output in the form of feature vectors for the words in that text corpus. Feature vector is a vector that is formed after doing Lemmatization, stop word

removal. Word2vec trains words against other words that neighbor them in the input corpus. Similarities are done between words based on the associations.

We gave `medicaments.json` as input file which contains data about existing drugs and their composition, manufactured date, number of them available etc.

```

// Turn off Info Logger for Console
Logger.getLogger("org").setLevel(Level.OFF);
Logger.getLogger("akka").setLevel(Level.OFF);

// Read the file into RDD[String]
val input = sc.textFile("data/medicaments").map(line => {
    // Getting Lemmatized Form of the word using CoreNLP
    val lemma = CoreNLP.returnLemma(line)
    (0, lemma)
})

//Creating DataFrame from RDD

val sentenceData = spark.createDataFrame(input).toDF("labels", "sentence")

//Tokenizer
val tokenizer = new Tokenizer().setInputCol("sentence").setOutputCol("words")
val wordsData = tokenizer.transform(sentenceData)

//Stop Word Remover
val remover = new StopWordsRemover()
.setInputCol("words")
.setOutputCol("filteredWords")
val processedWordData = remover.transform(wordsData)

```

The screenshot shows three instances of the IntelliJ IDEA interface. The top instance displays the `SparkW2VML.scala` file with code related to reading a text file and performing NLP operations. The middle instance shows the project structure with the `SparkW2VML` module selected. The bottom instance shows the same code as the top one, indicating a step-by-step process or a comparison.

**Output snippet is shown below**

```
bloc :  
[0,8,0.999991693342083]  
[séborrhéique,0.9999624167176876]  
[trafic,0.999951455955272]  
  
power :  
[10/07/2013,0.9995441497647819]  
[2003-08-18,0.9993619630221745]  
[sous-cutanées,0.9993440385325913]  
  
décurarisation :  
[nanocolloïdes,0.9999969329010768]  
[marcaine,0.9999789195759184]  
[954-5,0.9999456291830718]  
  
neuromusculaire :  
[64975424,0.9999907636165579]  
[18,2,0.9999839484435472]  
[suspicion,0.9999696840965253]  
  
induit :  
[prévention,0.9999922947274583]  
[theostat,0.9998890689560397]  
[rhumatismes,0.9998063200484436]  
  
rocuronium :  
[cytotec,0.9999636647599903]  
[213-5,0.9998232878230584]  
[fluocyne,0.9998151128176668]  
  
cymevan :  
[aas,0.9999503875533897]  
[membrane,0.9999468208849455]  
[antigène,0.9999024447530916]
```

**K-MEANS:**

In K Means we can cluster the whole bunch of documents and extract some most common key words to represent the topic of the document.

# **Project Report Increment 2**

```
Reading POS tagger model from edu/stanford/nlp/models/pos-tagger/english-left3words/english-left3words-distsim.tagger ... done [2.8 sec].
root
|--- location: string (nullable = true)
|--- docs: string (nullable = true)
|--- rawTokens: array (nullable = true)
|   |--- element: string (containsNull = true)
|--- tokens: array (nullable = true)
|   |--- element: string (containsNull = true)
|--- features: vector (nullable = true)

()

[file:///E:/#UMKC/KDM/Tutorial/9/CS5560 - Tutorial 9 Source Code/Spark_KMeans_FV/data/drugs/med.Insulin/1400.txt,RDF datum ospa lipoprotein Resource URI http wifo informatik unus

Corpus summary:
  Training set size: 12 documents
  Vocabulary size: 968 terms
  Preprocessing time: 27.49186883 sec

16/07/06 21:18:25 WARN BLAS: Failed to load implementation from: com.github.fommil.netlib.NativeSystemBLAS
16/07/06 21:18:25 WARN BLAS: Failed to load implementation from: com.github.fommil.netlib.NativeRefBLAS
Finished training KMeans model.  Summary:
  Training time: 1.951532782 sec

Process finished with exit code 0
```

LDA:

## Project Report Increment 2

SparkLDA - [C:\Users\ndar\Desktop\Acads\CS5560 - Tutorial 8 Source Code\SparkLDA] - [sparkida] - ..\data\ner\domainBasedWords - IntelliJ IDEA 15.0.3

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

SparkLDA > data > ner > domainBasedWords

Project Structure

```

SparkLDA (sparkida) (C:\Users\ndar\Desktop\Acads\CS5560
  +--.idea
  +-- data
    +-- 20_news_group
      +-- sc_crypttxt
      +-- sc_electronics.txt
      +-- sc_medtxt
      +-- sc_space.txt
    +-- ner
      +-- domainBasedWords
        +-- sample
        +-- Results.txt
  +-- sample
  +-- domainBasedWords
  +-- Results.txt

```

Run SparkLDA

```

18 5.325394948025587E-4
2016 5.325394947943757E-4
fewer 5.20165091160667E-4
expiry 5.201650911038421E-4
; 5.201650910887035E-4
lastly 5.2016509107417E-4
justify 5.201650910729724E-4
period 5.201650910465048E-4
capture 5.201650910420657E-4
y 5.201650910376502E-4
' 5.201650910265634E-4
lifesaving 5.201650910202453E-4
competitor 5.20165091014611E-4
al 5.201650910118556E-4
2.6 5.201650910107151E-4
hinder 5.201650910090387E-4
develop 5.20165090998469E-4
generic 5.201650909979478E-4
70 5.20165090977437E-4

```

Process finished with exit code 0

Terminal Run TODO

All files are up-to-date (a minute ago)

Event Log 14985:1 CRLF: UTF-8: 4:05 PM 7/8/2016

Ask me anything

SparkLDA - [C:\Users\ndar\Desktop\Acads\CS5560 - Tutorial 8 Source Code\SparkLDA] - [sparkida] - ..\data\Results.txt - IntelliJ IDEA 15.0.3

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

SparkLDA > data > Results.txt

Project Structure

```

SparkLDA (sparkida) (C:\Users\ndar\Desktop\Acads\CS5560
  +--.idea
  +-- data
    +-- 20_news_group
      +-- sc_crypttxt
      +-- sc_electronics.txt
      +-- sc_medtxt
      +-- sc_space.txt
    +-- ner
      +-- domainBasedWords
        +-- sample
        +-- Results.txt
  +-- project (sparkida-build) (sources root)
  +-- spark-warehouse
  +-- src
    +-- main
      +-- java
  +-- Results.txt

```

Run SparkLDA

Corpus summary:

- Training set size: 97 documents
- Vocabulary size: 741 terms
- Training set size: 1494 tokens
- Preprocessing time: 15.763828637 sec

Finished training LDA model. Summary:

- Training time: 11.916366596 sec
- Training data average log likelihood: -110.6917543173391

20 topics:

- TOPIC\_0:, 0.08755849653882213
- TOPIC\_0:, 0.05620948269393406
- TOPIC\_0:, 0.04620890538997851
- TOPIC\_0:, 0.04620890538997851
- TOPIC\_0:, 0.012336901228243068
- TOPIC\_0:, 0.01070585550805288
- TOPIC\_0:, 0.009146675743181202
- TOPIC\_0:, 0.008824877931782904
- TOPIC\_0:, 0.008313260817210013
- S\_0.007579735772978709
- cancer\_0.0075009512730609545
- \_0.00701399514512827
- spending\_0.0069025923049940326

Training data average log likelihood: -110.6917543173391

20 topics:

- TOPIC\_0:, 0.08755849653882213
- TOPIC\_0:, 0.05620948269393406
- TOPIC\_0:, 0.04620890538997851
- TOPIC\_0:, 0.04620890538997851
- TOPIC\_0:, 0.012336901228243068
- TOPIC\_0:, 0.01070585550805288
- TOPIC\_0:, 0.009146675743181202
- TOPIC\_0:, 0.008824877931782904
- TOPIC\_0:, 0.008313260817210013
- S\_0.007579735772978709
- cancer\_0.0075009512730609545
- \_0.00701399514512827
- spending\_0.0069025923049940326

Terminal Run TODO

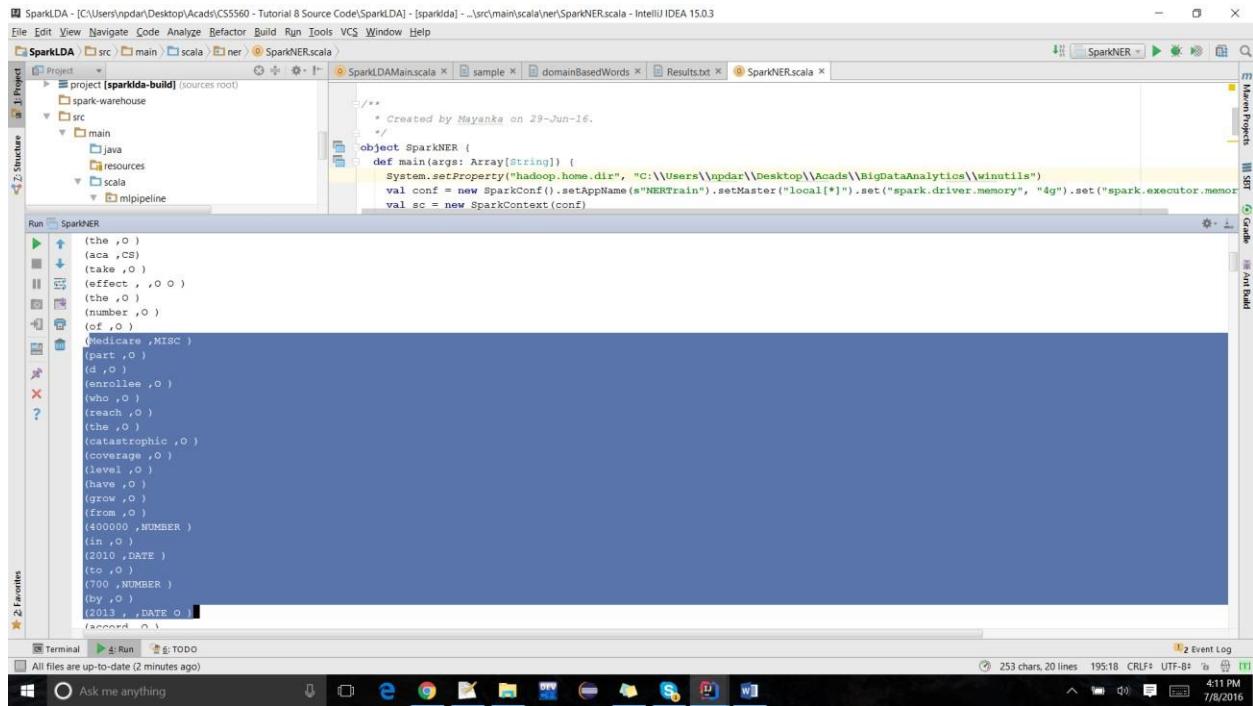
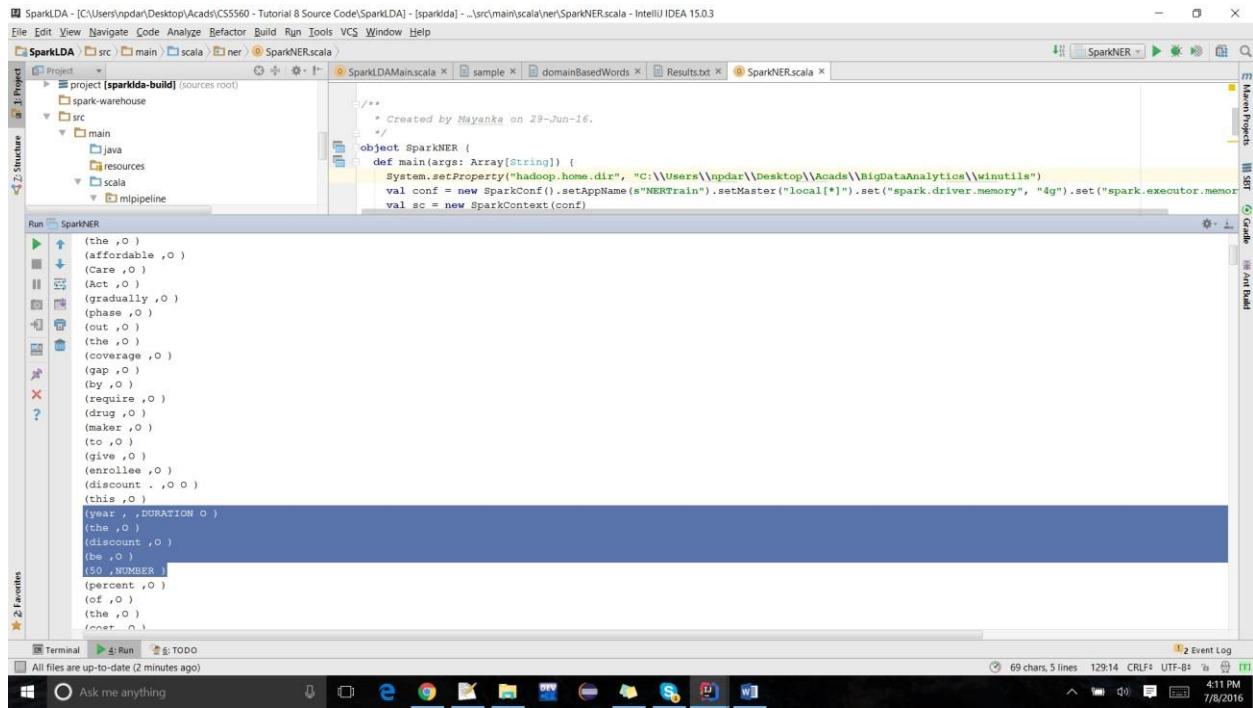
All files are up-to-date (2 minutes ago)

Event Log 15:33 CRLF: UTF-8: 4:05 PM 7/8/2016

Ask me anything

Named Entity Relation results based on the domain based words:

## Project Report Increment 2



## Project Report Increment 2

SparkLDA - [C:\Users\ndar\Desktop\Acads\CS5560 - Tutorial 8 Source Code\SparkLDA] - [sparkida] - ...\\src\\main\\scala\\ner\\SparkNER.scala - IntelliJ IDEA 15.0.3

```

/* Created by Mayanka on 29-Jun-16.
*/
object SparkNER {
  def main(args: Array[String]) {
    System.setProperty("hadoop.home.dir", "C:\\Users\\ndar\\Desktop\\Acads\\BigDataAnalytics\\winutils")
    val conf = new SparkConf().setAppName("NERTrain").setMaster("local[*]").set("spark.driver.memory", "4g").set("spark.executor.memory", "4g")
    val sc = new SparkContext(conf)
  }
}

(this, o)
(will, o)
(actually, o)
(increase, o)
(spending, ' ', o o o)
(say, o)
(Carolina, PERSON)
(Person, PERSON)
(of, o)
(consult, o)
(firm, o)
(Avalere, o)
(in, o)
(Washington, LOCATION o)
(D.C., LOCATION o)
(which, o)
(be, o)
(in, o)
(the, o)
(process, o)
(of, o)
(analyze, o)
(the, o)
(financial, o)
(impact, o)
(of, o)
(the, o)
(marinac, o)

```

Run SparkNER

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

SparkLDA > src > main > scala > ner > SparkNER.scala

Project Structure

Run Terminal Event Log

All files are up-to-date (2 minutes ago)

4:12 PM 7/8/2016

SparkLDA - [C:\Users\ndar\Desktop\Acads\CS5560 - Tutorial 8 Source Code\SparkLDA] - [sparkida] - ...\\src\\main\\scala\\ner\\SparkNER.scala - IntelliJ IDEA 15.0.3

```

/* Created by Mayanka on 29-Jun-16.
*/
object SparkNER {
  def main(args: Array[String]) {
    System.setProperty("hadoop.home.dir", "C:\\Users\\ndar\\Desktop\\Acads\\BigDataAnalytics\\winutils")
    val conf = new SparkConf().setAppName("NERTrain").setMaster("local[*]").set("spark.driver.memory", "4g").set("spark.executor.memory", "4g")
    val sc = new SparkContext(conf)
  }
}

(factor, o)
(help, o)
(drive, CS)
(spending, o)
(in, o)
(the, o)
(program, o)
(to, o)
($ 75, MONEY MONEY)
(billion, NUMBER)
(in, o)
(2014, DATE)
(-, o)
(and, o)
(more, o)
(high-cost, o)
(drug, o)
(will, o)
(hit, o)
(the, o)
(market, o)
(soon, o o)
(without, o)
(action, o o)
(the, o)
(rise, CS)
(cost, o)
(could, o)

```

Run SparkNER

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

SparkLDA > src > main > scala > ner > SparkNER.scala

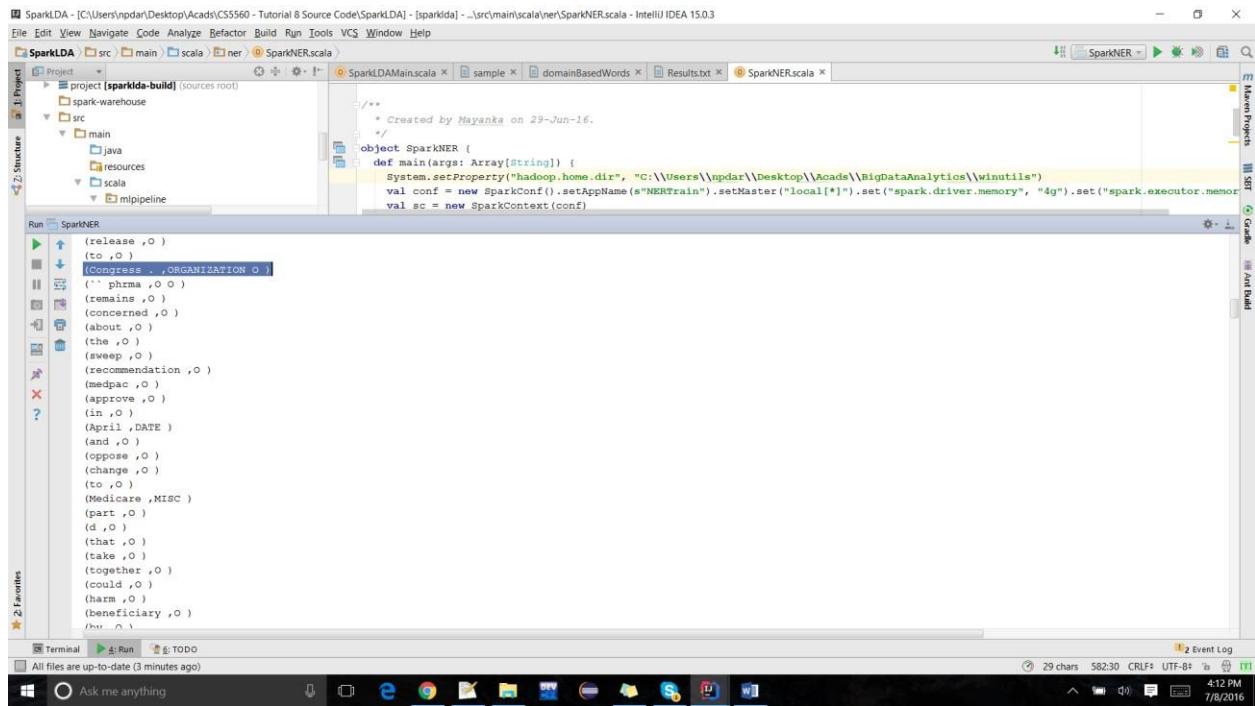
Project Structure

Run Terminal Event Log

All files are up-to-date (3 minutes ago)

4:12 PM 7/8/2016

## Project Report Increment 2



## OPEN IE:

The screenshot shows the IntelliJ IDEA interface with the 'sentenceSample' project selected. The 'Run' tab contains the following NLP code:

```

[(It,'s Like,Have ADHDRecognize Withdrawal Symptoms,0.08771302436841699)
[(Tips,Avoid,HangoverDrug Treatments,1.0)]
[(Street Drugs,Know,FactsCrystal Meth 101,1.0)

[(WebMD Feature,Reviewed by,Louise Chang,1.0)]

[(parents,worried In,1970s,1.0), (their longhaired,getting,drunk marijuana,1.0), (their,longhaired,bell-bottomed teenagers,1.0), (their longhaired,getting,marijuana,1.0)][(pare
[(rise,is in,country,1.0), (Prescription drug abuse,be on,rise,1.0), (Prescription drug abuse,appears,rise,1.0)][(rise,is in,country,1.0), (Prescription drug abuse,be on,rise,1
[(opportunity,increasing numbers of,prescriptions written for certain drugs,0.8551475344953919), (greater opportunity,numbers of,prescriptions written,0.8551475344953919), (opp

Process finished with exit code 0

```

The terminal at the bottom shows:

```

Compilation completed successfully with 3 warnings in 14s 41ms (a minute ago)

```

The screenshot shows the IntelliJ IDEA interface with the 'sentenceSample' project selected. The 'Run' tab contains the following NLP code:

```

[(parents,worried In,1970s,1.0), (their longhaired,getting,drunk marijuana,1.0), (their,longhaired,bell-bottomed teenagers,1.0), (their longhaired,getting,marijuana,1.0)][(pare
[(abused drugs,is in,U.S. Over-the-counter,1.0)]
[(abused drugs,is in,U.S. Over-the-counter,1.0)][(some,can,can addictive,1.0), (they,'re used,wrong way,1.0), (they,'re used,way,1.0), (they,'re,used,0.08525081604622908)][(som
[(rise,is in,country,1.0), (Prescription drug abuse,be on,rise,1.0), (Prescription drug abuse,appears,rise,1.0)][(rise,is in,country,1.0), (Prescription drug abuse,be on,rise,1
[(rise,is in,country,1.0), (Prescription drug abuse,be on,rise,1.0), (Prescription drug abuse,appears,rise,1.0)][(rise,is in,country,1.0), (Prescription drug abuse,be on,rise,1

[(you,spot,misused medicines,1.0), (you,spot,commonly misused medicines,1.0)][(you,spot,misused medicines,1.0), (you,spot,medicines,1.0), (you,spot,co
[(you,spot,misused medicines,1.0), (you,spot,medicines,1.0), (you,spot,commonly misused medicines,1.0)][(you,spot,misused medicines,1.0), (you,spot,medicines,1.0), (you,spot,co
[(you,spot,misused medicines,1.0), (you,spot,medicines,1.0), (you,spot,commonly misused medicines,1.0)][(you,spot,misused medicines,1.0), (you,spot,medicines,1.0), (you,spot,co

[(opportunity,increasing numbers of,prescriptions written for certain drugs,0.8551475344953919), (greater opportunity,numbers of,prescriptions written,0.8551475344953919), (opp
[(opportunity,increasing numbers of,prescriptions written for certain drugs,0.8551475344953919), (greater opportunity,numbers of,prescriptions written,0.8551475344953919), (opp

[(It,'s Like,Have ADHDRecognize Withdrawal Symptoms,0.08771302436841699)

[(Tips,Avoid,HangoverDrug Treatments,1.0)]
[(Street Drugs,Know,FactsCrystal Meth 101,1.0)

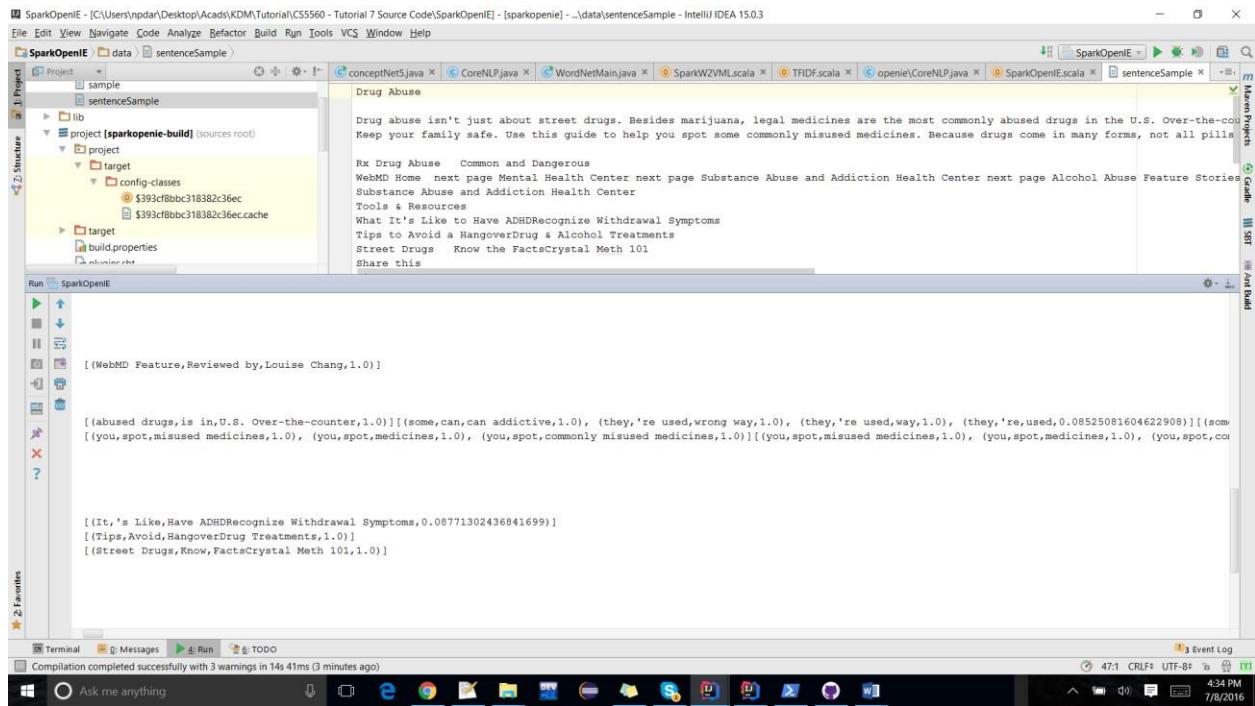

```

The terminal at the bottom shows:

```

Compilation completed successfully with 3 warnings in 14s 41ms (3 minutes ago)

```



### Comparison:

Both K-Means and LDA are unsupervised learning, also both will have parameters like K value, number of clusters and number of topics. For suppose if we are K topics in set of N documents, K-Means will N number of documents into K disjoint clusters (topics). But in LDA, it assigns a document of many different topics which are mixed, because of which we are getting more than one topic for each document. Therefore, LDA will be more realistic than K-Means in topic assignment.

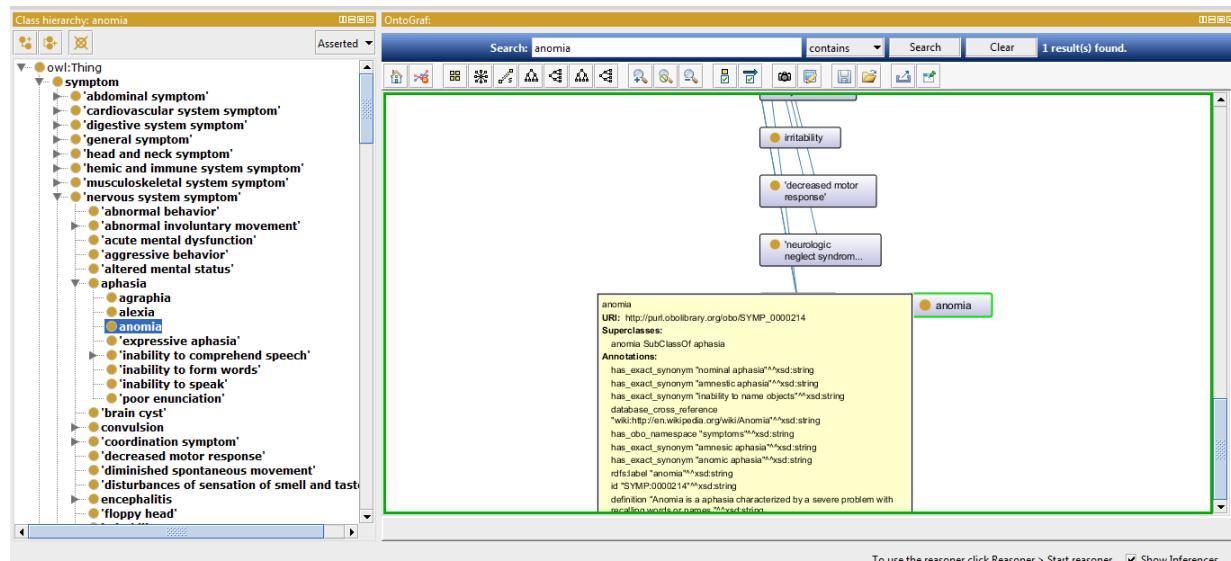
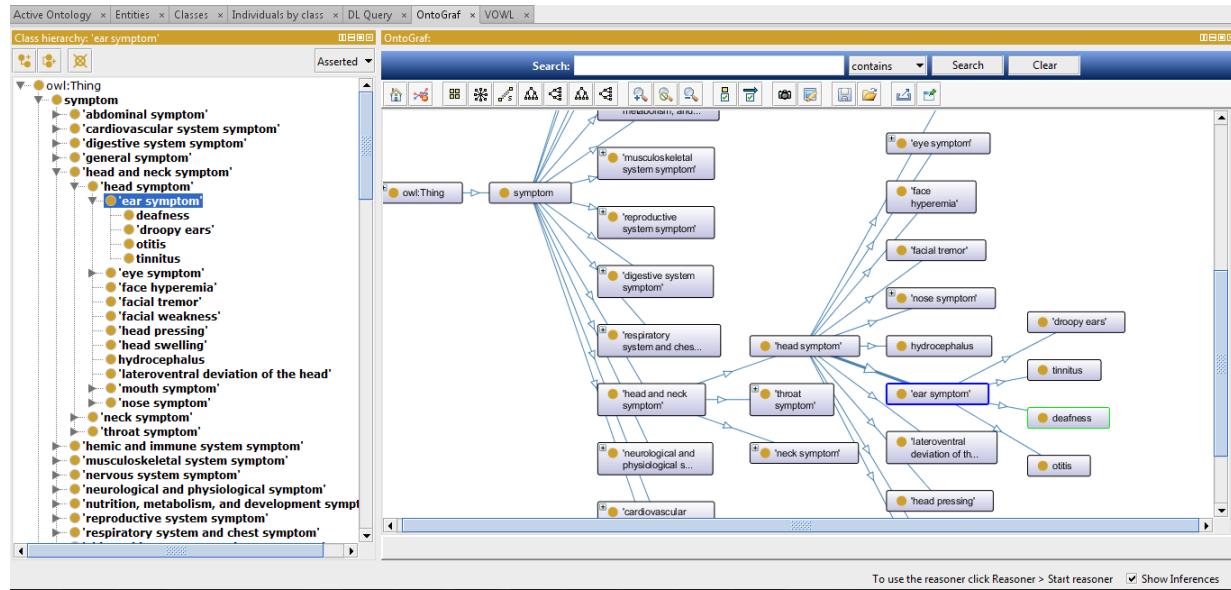
### 7.3.3 Third Increment:

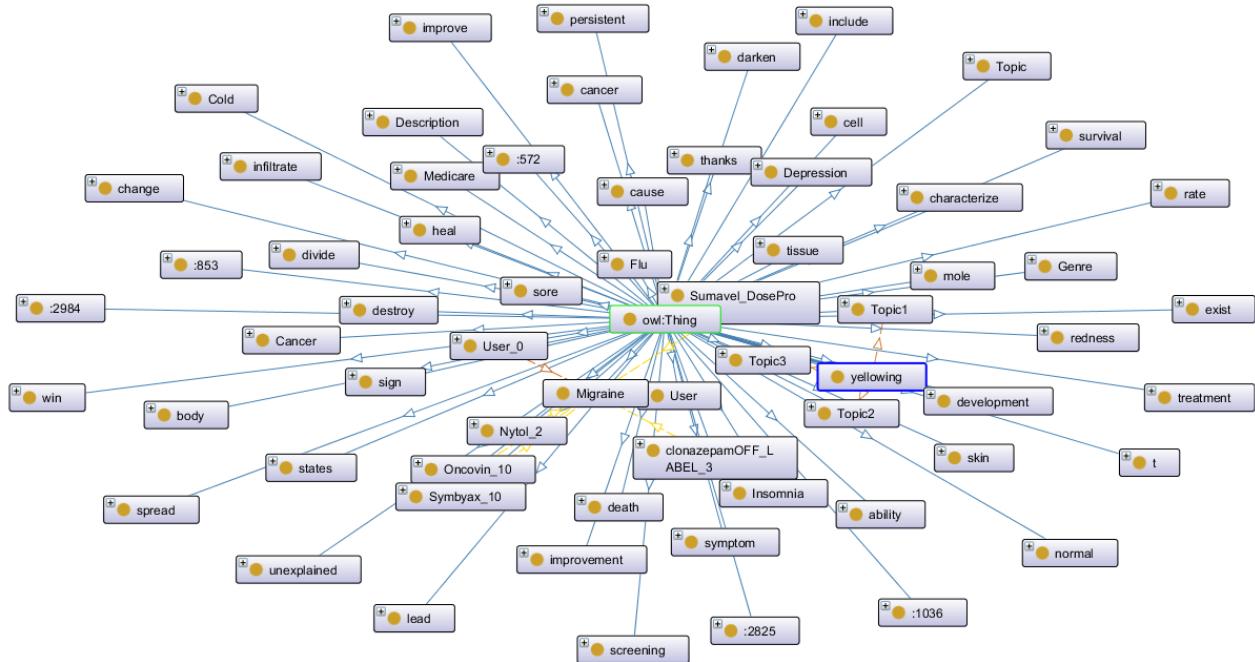
```

2792   <rdf:datatype="http://www.w3.org/2001/XMLSchema#string">SYMP:0000213</oboInOwl:id>
2793   <ExactSynonym rdf:datatype="http://www.w3.org/2001/XMLSchema#string">inability to write</oboInOwl:hasExactSynonym>
2794   <OBONamespace rdf:datatype="http://www.w3.org/2001/XMLSchema#string">symptoms</oboInOwl:hasOBONamespace>
2795
2796   <dTarget rdf:datatype="http://www.w3.org/2001/XMLSchema#string">Agraphia is a aphasia where there is a pathological loss of the ability to write.</o
2797   <DbXref rdf:datatype="http://www.w3.org/2001/XMLSchema#string">URL: http://www2.merriam-webster.com/cgi-bin/mwmednlm?book=Medical&va=agraphia</ob
2798   <dProperty rdf:resource="http://purl.obolibrary.org/obo/IAO_0000115"/>
2799   <dSource rdf:resource="http://purl.obolibrary.org/obo/SYMP_0000213"/>
2800
2801
2802
2803
2804
2805   obolibrary.org/obo/SYMP_0000214 -->
2806
2807   <out="http://purl.obolibrary.org/obo/SYMP_0000214">
2808   <df:datatype="http://www.w3.org/2001/XMLSchema#string">anomia</rdfs:label>
2809   <sof rdf:resource="http://purl.obolibrary.org/obo/SYMP_0000508"/>
2810   <i115 rdf:datatype="http://www.w3.org/2001/XMLSchema#string">Anomia is a aphasia characterized by a severe problem with recalling words or names.</ob
2811   <ExactSynonym rdf:datatype="http://www.w3.org/2001/XMLSchema#string">SYMP:0000214</oboInOwl:id>
2812   <ExactSynonym rdf:datatype="http://www.w3.org/2001/XMLSchema#string">amnesic aphasia</oboInOwl:hasExactSynonym>
2813   <ExactSynonym rdf:datatype="http://www.w3.org/2001/XMLSchema#string">amnestic aphasia</oboInOwl:hasExactSynonym>
2814   <ExactSynonym rdf:datatype="http://www.w3.org/2001/XMLSchema#string">anomic aphasia</oboInOwl:hasExactSynonym>
2815   <ExactSynonym rdf:datatype="http://www.w3.org/2001/XMLSchema#string">inability to name objects</oboInOwl:hasExactSynonym>
2816   <ExactSynonym rdf:datatype="http://www.w3.org/2001/XMLSchema#string">nominal aphasia</oboInOwl:hasExactSynonym>
2817   <OBONamespace rdf:datatype="http://www.w3.org/2001/XMLSchema#string">symptoms</oboInOwl:hasOBONamespace>
2818
2819   <dTarget rdf:datatype="http://www.w3.org/2001/XMLSchema#string">Anomia is a aphasia characterized by a severe problem with recalling words or names..<
2820   <DbXref rdf:datatype="http://www.w3.org/2001/XMLSchema#string">wiki: http://en.wikipedia.org/wiki/Anomia</oboInOwl:hasDbXref>
2821   <dProperty rdf:resource="http://purl.obolibrary.org/obo/IAO_0000115"/>
2822   <dSource rdf:resource="http://purl.obolibrary.org/obo/SYMP_0000214"/>
2823
2824
2825
2826
2827
2828   <!-- http://purl.obolibrary.org/obo/SYMP_0000215 -->
2829
2830   <owl:Class rdf:about="http://purl.obolibrary.org/obo/SYMP_0000215">
2831     <rdfs:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string">inability to comprehend speech</rdfs:label>
2832     <rdfs:subClassOf rdf:resource="http://purl.obolibrary.org/obo/SYMP_0000508"/>
2833     <oboInOwl:id rdf:datatype="http://www.w3.org/2001/XMLSchema#string">SYMP:0000215</oboInOwl:id>
2834     <oboInOwl:hasOBONamespace rdf:datatype="http://www.w3.org/2001/XMLSchema#string">symptoms</oboInOwl:hasOBONamespace>
2835
2836
2837
2838
2839   <!-- http://purl.obolibrary.org/obo/SYMP_0000216 -->
2840
2841   <owl:Class rdf:about="http://purl.obolibrary.org/obo/SYMP_0000216">
2842     <rdfs:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string">inability to speak</rdfs:label>
2843     <rdfs:subClassOf rdf:resource="http://purl.obolibrary.org/obo/SYMP_0000508"/>
2844     <oboInOwl:id rdf:datatype="http://www.w3.org/2001/XMLSchema#string">SYMP:0000216</oboInOwl:id>
2845     <oboInOwl:hasOBONamespace rdf:datatype="http://www.w3.org/2001/XMLSchema#string">symptoms</oboInOwl:hasOBONamespace>
2846
2847
2848
2849
2850   <!-- http://purl.obolibrary.org/obo/SYMP_0000217 -->
2851
2852   <owl:Class rdf:about="http://purl.obolibrary.org/obo/SYMP_0000217">
2853     <rdfs:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string">inability to form words</rdfs:label>
2854     <rdfs:subClassOf rdf:resource="http://purl.obolibrary.org/obo/SYMP_0000508"/>
2855     <oboInOwl:id rdf:datatype="http://www.w3.org/2001/XMLSchema#string">SYMP:0000217</oboInOwl:id>
2856     <oboInOwl:hasOBONamespace rdf:datatype="http://www.w3.org/2001/XMLSchema#string">symptoms</oboInOwl:hasOBONamespace>
2857
2858
2859
2860
2861   <!-- http://purl.obolibrary.org/obo/SYMP_0000218 -->
2862
2863   <owl:Class rdf:about="http://purl.obolibrary.org/obo/SYMP_0000218">
2864     <rdfs:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string">poor enunciation</rdfs:label>
2865
2866   !!!

```

## Visualization:





## Project Report Increment 2

### SPARQL:

The screenshot shows the Apache Jena Fuseki interface for a dataset named '/drugDataset'. The main page displays various service endpoints and statistics. Under 'Available services', it lists:

- File Upload: http://localhost:3030/drugDataset/upload
- Graph Store Protocol: http://localhost:3030/drugDataset/data
- Graph Store Protocol (Read): http://localhost:3030/drugDataset/get
- HTTP Quads: http://localhost:3030/drugDataset/
- SPARQL Query: http://localhost:3030/drugDataset/query
- SPARQL Update: http://localhost:3030/drugDataset/sparql
- SPARQL Update: http://localhost:3030/drugDataset/update

The 'Statistics' section shows the following data for '/drugDataset':

Name	Overall	Overall good	Overall bad	SPARQL Update	SPARQL Query	SPARQL Query	Graph Store Protocol	Graph Store Protocol (Read)	HTTP Quads	File Upload
/drugDataset	7	7	0	0	6 (0 bad)	0	1 (0 bad)	0	0	0

The 'Dataset size' section indicates there are 1142 triples in the default graph. The 'Ongoing operations' section notes that backups are pending.

The screenshot shows the Apache Jena Fuseki interface for a dataset named '/drugDataset' in the 'SPARQL query' section. The user has selected 'Selection of triples' from the example queries. The query editor contains the following SPARQL code:

```
1
2
3 SELECT ?subject ?predicate ?object
4 WHERE {
5   ?subject ?predicate ?object
6 }
7 LIMIT 25
```

The configuration for the SPARQL endpoint is set to 'http://localhost:3030/drugDataset/query' with 'CONTENT TYPE (SELECT)' as 'JSON' and 'CONTENT TYPE (GRAPH)' as 'Turtle'.

## Project Report Increment 2

QUERY RESULTS		
subject	predicate	object
1 <https://www.kdm.com/OWL/drugs#no>	<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	<https://www.kdm.com/OWL/drugs#Cold_Description>
2 <https://www.kdm.com/OWL/drugs#no>	<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	<http://www.w3.org/2002/07/owl#NamedIndividual>
3 <https://www.kdm.com/OWL/drugs#853>	<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	<http://www.w3.org/2002/07/owl#NamedIndividual>
4 <https://www.kdm.com/OWL/drugs#853>	<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	<https://www.kdm.com/OWL/drugs#DrugID>
5 <https://www.kdm.com/OWL/drugs#provide>	<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	<http://www.w3.org/2002/07/owl#NamedIndividual>
6 <https://www.kdm.com/OWL/drugs#provide>	<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	<https://www.kdm.com/OWL/drugs#Cancer_Description>
7 <https://www.kdm.com/OWL/drugs#Symbanax_10>	<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	<http://www.w3.org/2002/07/owl#NamedIndividual>
8 <https://www.kdm.com/OWL/drugs#Symbanax_10>	<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	<https://www.kdm.com/OWL/drugs#DrugIII>
9 <https://www.kdm.com/OWL/drugs#Symbanax_10>	<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	<http://www.w3.org/2002/07/owl#Class>
10 <https://www.kdm.com/OWL/drugs#within>	<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	<https://www.kdm.com/OWL/drugs#Cold_Description>
11 <https://www.kdm.com/OWL/drugs#within>	<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	<http://www.w3.org/2002/07/owl#NamedIndividual>
12 <https://www.kdm.com/OWL/drugs#abnormally>	<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	<http://www.w3.org/2002/07/owl#NamedIndividual>
13 <https://www.kdm.com/OWL/drugs#abnormally>	<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	<https://www.kdm.com/OWL/drugs#Cancer_Description>
14 <https://www.kdm.com/OWL/drugs#evaluation>	<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	<http://www.w3.org/2002/07/owl#NamedIndividual>
15 <https://www.kdm.com/OWL/drugs#evaluation>	<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	<https://www.kdm.com/OWL/drugs#Cancer_Description>
16 <https://www.kdm.com/OWL/drugs#:1036>	<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	<http://www.w3.org/2002/07/owl#Class>
Showing 1 to 25 of 25 entries		
8 <https://www.kdm.com/OWL/drugs#Symbanax_10>		
9 <https://www.kdm.com/OWL/drugs#Symbanax_10>	<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	<https://www.kdm.com/OWL/drugs#DrugIII>
10 <https://www.kdm.com/OWL/drugs#within>	<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	<http://www.w3.org/2002/07/owl#Class>
11 <https://www.kdm.com/OWL/drugs#within>	<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	<https://www.kdm.com/OWL/drugs#Cold_Description>
12 <https://www.kdm.com/OWL/drugs#abnormally>	<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	<http://www.w3.org/2002/07/owl#NamedIndividual>
13 <https://www.kdm.com/OWL/drugs#abnormally>	<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	<https://www.kdm.com/OWL/drugs#Cancer_Description>
14 <https://www.kdm.com/OWL/drugs#evaluation>	<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	<http://www.w3.org/2002/07/owl#NamedIndividual>
15 <https://www.kdm.com/OWL/drugs#evaluation>	<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	<https://www.kdm.com/OWL/drugs#Cancer_Description>
16 <https://www.kdm.com/OWL/drugs#:1036>	<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	<http://www.w3.org/2002/07/owl#Class>
17 <https://www.kdm.com/OWL/drugs#talk>	<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	<https://www.kdm.com/OWL/drugs#Cold_Description>
18 <https://www.kdm.com/OWL/drugs#talk>	<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	<http://www.w3.org/2002/07/owl#NamedIndividual>
19 <https://www.kdm.com/OWL/drugs#possible>	<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	<https://www.kdm.com/OWL/drugs#Cold_Description>
20 <https://www.kdm.com/OWL/drugs#possible>	<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	<http://www.w3.org/2002/07/owl#NamedIndividual>
21 <https://www.kdm.com/OWL/drugs#infiltrate>	<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	<http://www.w3.org/2002/07/owl#NamedIndividual>
22 <https://www.kdm.com/OWL/drugs#infiltrate>	<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	<https://www.kdm.com/OWL/drugs#Cancer_Description>
23 <https://www.kdm.com/OWL/drugs#infiltrate>	<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	<http://www.w3.org/2002/07/owl#Class>
24 <https://www.kdm.com/OWL/drugs#vaccine>	<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	<https://www.kdm.com/OWL/drugs#Cold_Description>
25 <https://www.kdm.com/OWL/drugs#vaccine>	<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	<http://www.w3.org/2002/07/owl#NamedIndividual>
Showing 1 to 25 of 25 entries		

## Project Report Increment 2

The screenshot shows the Apache Jena Fuseki interface running at [localhost:3030/dataset.html?tab=upload&ds=/Drug\\_Dataset](http://localhost:3030/dataset.html?tab=upload&ds=/Drug_Dataset). The top navigation bar includes links for Apps, BlueMix, MapReduce, Algorithms, dominicklim/dec, BigData Project, Interns, Camunda, npdarsini (Priyad), and CS5560 - 2016 S. A server status icon is visible in the top right.

The main area displays a SPARQL query editor for the dataset `/Drug_Dataset`. The query pane contains the following code:

```
1
2
3 SELECT ?subject ?predicate ?object
4 WHERE {
5   ?subject ?predicate ?object
6 }
7 LIMIT 25
```

The results pane shows 15 entries, with the first few rows being:

subject	predicate	object
http://www.w3.org/ns/csvw#row	http://www.w3.org/ns/csvw#rowNumber	1
http://www.w3.org/ns/csvw#row	http://www.w3.org/ns/csvw#rowNumber	2
http://www.w3.org/ns/csvw#row	http://www.w3.org/ns/csvw#rowNumber	3
http://www.w3.org/ns/csvw#row	http://www.w3.org/ns/csvw#rowNumber	4
http://www.w3.org/ns/csvw#row	http://www.w3.org/ns/csvw#rowNumber	5
http://www.w3.org/ns/csvw#row	http://www.w3.org/ns/csvw#rowNumber	6
http://www.w3.org/ns/csvw#row	http://www.w3.org/ns/csvw#rowNumber	7
http://www.w3.org/ns/csvw#row	http://www.w3.org/ns/csvw#rowNumber	8
http://www.w3.org/ns/csvw#row	http://www.w3.org/ns/csvw#rowNumber	9
http://www.w3.org/ns/csvw#row	http://www.w3.org/ns/csvw#rowNumber	10
http://www.w3.org/ns/csvw#row	http://www.w3.org/ns/csvw#rowNumber	11
http://www.w3.org/ns/csvw#row	http://www.w3.org/ns/csvw#rowNumber	12
http://www.w3.org/ns/csvw#row	http://www.w3.org/ns/csvw#rowNumber	13
http://www.w3.org/ns/csvw#row	http://www.w3.org/ns/csvw#rowNumber	14
http://www.w3.org/ns/csvw#row	http://www.w3.org/ns/csvw#rowNumber	15

QUERY RESULTS			
	subject	predicate	object
Showing 1 to 15 of 15 entries	Search: <input type="text"/> Show 50 entries		
1	<http://purl.oclibrary.org/obo/drone.owl>	<http://www.w3.org/2002/07/owl#imports>	<http://purl.oclibrary.org/obo/drone-upper.owl>
2	<http://purl.oclibrary.org/obo/drone.owl>	<http://www.w3.org/2002/07/owl#imports>	<http://purl.oclibrary.org/obo/drone-nodc.owl>
3	<http://purl.oclibrary.org/obo/drone.owl>	<http://www.w3.org/2002/07/owl#imports>	<http://purl.oclibrary.org/obo/drone-rxnorm.owl>
4	<http://purl.oclibrary.org/obo/drone.owl>	<http://www.w3.org/2002/07/owl#imports>	<http://purl.oclibrary.org/obo/drone-pro.owl>
5	<http://purl.oclibrary.org/obo/drone.owl>	<http://www.w3.org/2002/07/owl#imports>	<http://purl.oclibrary.org/obo/drone-ingredient.owl>
6	<http://purl.oclibrary.org/obo/drone.owl>	<http://www.w3.org/2002/07/owl#imports>	<http://purl.oclibrary.org/obo/drone-chebi.owl>
7	<http://purl.oclibrary.org/obo/drone.owl>	<http://www.w3.org/2000/01/rdf-schema#comment>	"DrOn contains content developed by the National Library of Medicine in RxNorm. In creating DrOn, we have used RxNorm content only with SAB = RXNORM."
8	<http://purl.oclibrary.org/obo/drone.owl>	<http://www.w3.org/2000/01/rdf-schema#comment>	"An ontology of drugs."
9	<http://purl.oclibrary.org/obo/drone.owl>	<http://www.w3.org/2000/01/rdf-schema#comment>	"When citing DrOn, use the permanent URL of the ontology: http://purl.oclibrary.org/obo/drone.owl. When referencing a specific component of the DrOn, such as a class, object property, annotation property, or individual, use the Uniform Resource Identifier (URI) of that component." @en
10	<http://purl.oclibrary.org/obo/drone.owl>	<http://purl.org/dc/elements/1.1/license>	"https://creativecommons.org/licenses/by/3.0/"@en
11	<http://purl.oclibrary.org/obo/drone.owl>	<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	<http://www.w3.org/2002/07/owl#Ontology>
12	<http://purl.oclibrary.org/obo/drone.owl>	<http://purl.org/dc/elements/1.1/creator>	"Mathias Brochhausen"
13	<http://purl.oclibrary.org/obo/drone.owl>	<http://purl.org/dc/elements/1.1/creator>	"Josh Hanna"
14	<http://purl.oclibrary.org/obo/drone.owl>	<http://purl.org/dc/elements/1.1/creator>	"William Hogan"
15	<http://purl.oclibrary.org/obo/drone.owl>	<http://purl.org/dc/elements/1.1/creator>	"Eric Joseph"

## Project Report Increment 2

**SWRL:**

File Edit View Reasoner Tools Refactor Window Ontop Help

drugs (https://www.kdm.com/OWL/drugs) : [C:\Users\lakshmi\Downloads\CS5560-T14-SourceCode\W14 - Source Code\RecommendationWorkflow\data\drugRecommendation.owl]

Active Ontology Entities Individuals by class DL Query OntoGraf SWRLTab SPARQL Tab SPARQL Query

Name	Rule	Comment
S1	cause(cancerous) -> lead(death)	
S2	ability(recognize) -> ability(cancerous)	

Control Rules Asserted Axioms Inferred Axioms OWL 2 RL

OWL axioms successfully transferred to rule engine.  
Number of SWRL rules exported to rule engine: 2  
Number of OWL class declarations exported to rule engine: 66  
Number of OWL individual declarations exported to rule engine: 454  
Number of OWL object property declarations exported to rule engine: 6  
Number of OWL data property declarations exported to rule engine: 0  
Total number of OWL axioms exported to rule engine: 1572  
The transfer took 450 milliseconds(s).  
Press the 'Run Drools' button to run the rule engine.

OWL+SWRL->Drools Run Drools Drools->OWL

Control Rules Asserted Axioms Inferred Axioms OWL 2 RL

OWL axioms successfully transferred to rule engine.  
Number of SWRL rules exported to rule engine: 2  
Number of OWL class declarations exported to rule engine: 66  
Number of OWL individual declarations exported to rule engine: 454  
Number of OWL object property declarations exported to rule engine: 6  
Number of OWL data property declarations exported to rule engine: 0  
Total number of OWL axioms exported to rule engine: 1572  
The transfer took 221 milliseconds(s).  
Press the 'Run Drools' button to run the rule engine.  
Successful execution of rule engine.  
Number of inferred axioms: 749  
The process took 856 millisecond(s).  
Look at the 'Inferred Axioms' tab to see the inferred axioms.  
Press the 'Drools->OWL' button to translate the inferred axioms to OWL knowledge.  
Press the 'Run Drools' button to run the rule engine.

OWL+SWRL->Drools Run Drools Drools->OWL

File Edit View Reasoner Tools Refactor Window Ontop Help

drugs (https://www.kdm.com/OWL/drugs) : [C:\Users\lakshmi\Downloads\CS5560-T14-SourceCode\W14 - Source Code\RecommendationWorkflow\data\drugRecommendation.owl]

Active Ontology Entities Individuals by class DL Query OntoGraf SWRLTab SPARQL Tab SPARQL Query

Name	Rule	Comment
S1	cause(cancerous) -> lead(death)	
S2	ability(recognize) -> ability(cancerous)	

Control Rules Asserted Axioms Inferred Axioms OWL 2 RL

OWL axioms successfully transferred to rule engine.  
Number of SWRL rules exported to rule engine: 2  
Number of OWL class declarations exported to rule engine: 66  
Number of OWL individual declarations exported to rule engine: 454  
Number of OWL object property declarations exported to rule engine: 6  
Number of OWL data property declarations exported to rule engine: 0  
Total number of OWL axioms exported to rule engine: 1572  
The transfer took 221 milliseconds(s).  
Press the 'Run Drools' button to run the rule engine.  
Successful execution of rule engine.  
Number of inferred axioms: 749  
The process took 856 millisecond(s).  
Look at the 'Inferred Axioms' tab to see the inferred axioms.  
Press the 'Drools->OWL' button to translate the inferred axioms to OWL knowledge.  
Successfully transferred inferred axioms to OWL model.  
The process took 26 millisecond(s).

OWL+SWRL->Drools Run Drools Drools->OWL

## Project Report Increment 2

The screenshot shows the KDM interface with the 'Individuals by class' tab selected. The interface has a menu bar with File, Edit, View, Reasoner, Tools, Refactor, Window, Ontop, Help. Below the menu is a toolbar with buttons for New, Edit, Clone, Delete. The main area displays a table with columns Name, Rule, and Comment. Two rules are listed:

Name	Rule	Comment
S1	cancer(abdomen) -> lead(death)	
S2	cause(weakness) -> sameAs(feaver, symptom)	

At the bottom, there is a status bar with the message "No Reasoner set. Select a reasoner from the Reasoner menu" and a checkbox for "Show Inferences".

The screenshot shows the KDM interface with the 'Individuals by class' tab selected. The interface has a menu bar with File, Edit, View, Reasoner, Tools, Refactor, Window, Ontop, Help. Below the menu is a toolbar with buttons for New, Edit, Clone, Delete. The main area displays a table with columns Name, Rule, and Comment. Two rules are listed:

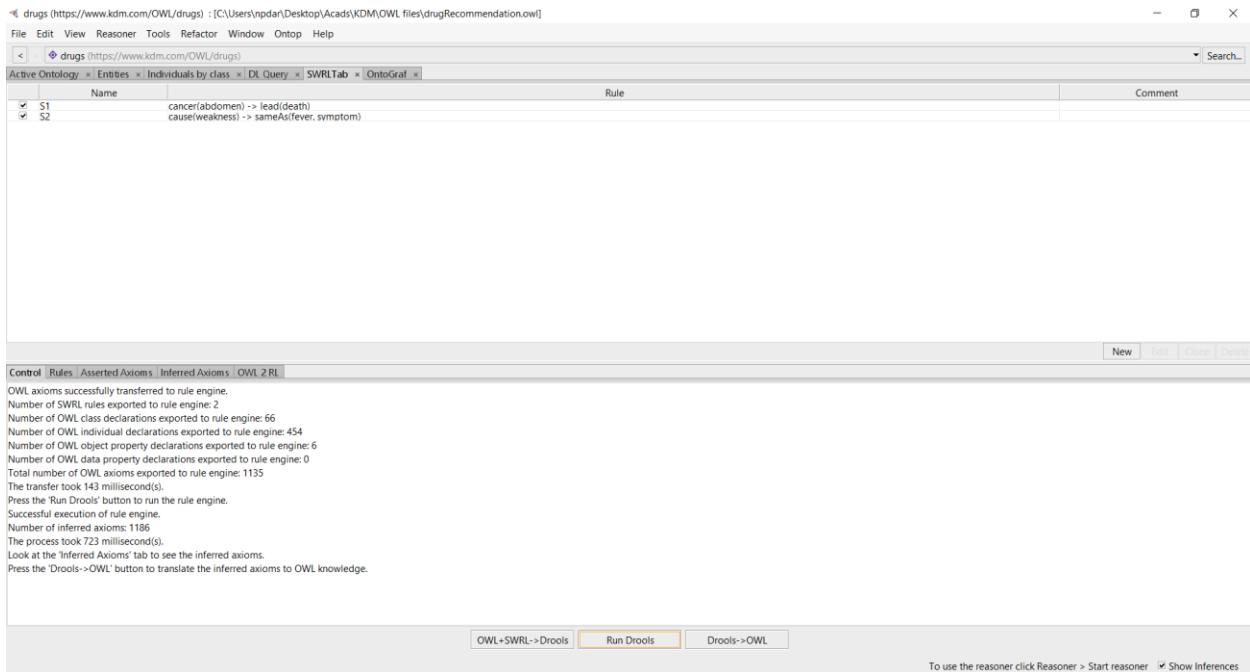
Name	Rule	Comment
S1	cancer(abdomen) -> lead(death)	
S2	cause(weakness) -> sameAs(feaver, symptom)	

Below the table, there is a detailed status message:

OWL axioms successfully transferred to rule engine.  
Number of SWRL rules exported to rule engine: 2  
Number of OWL class declarations exported to rule engine: 66  
Number of OWL individual declarations exported to rule engine: 454  
Number of OWL object property declarations exported to rule engine: 6  
Number of OWL data property declarations exported to rule engine: 0  
Total number of OWL axioms exported to rule engine: 1135  
The transfer took 143 milliseconds(s).  
Press the 'Run Drools' button to run the rule engine.

At the bottom, there is a status bar with the message "To use the reasoner click Reasoner > Start reasoner" and a checkbox for "Show Inferences".

## Project Report Increment 2



drugs (https://www.kdm.com/OWL/drugs) : [C:\Users\npda\Desktop\Acads\KDM\OWL files\drugRecommendation.owl]

File Edit View Reasoner Tools Refactor Window Ontop Help

Active Ontology Entities Individuals by class DL Query SWRLTab OntoGraf

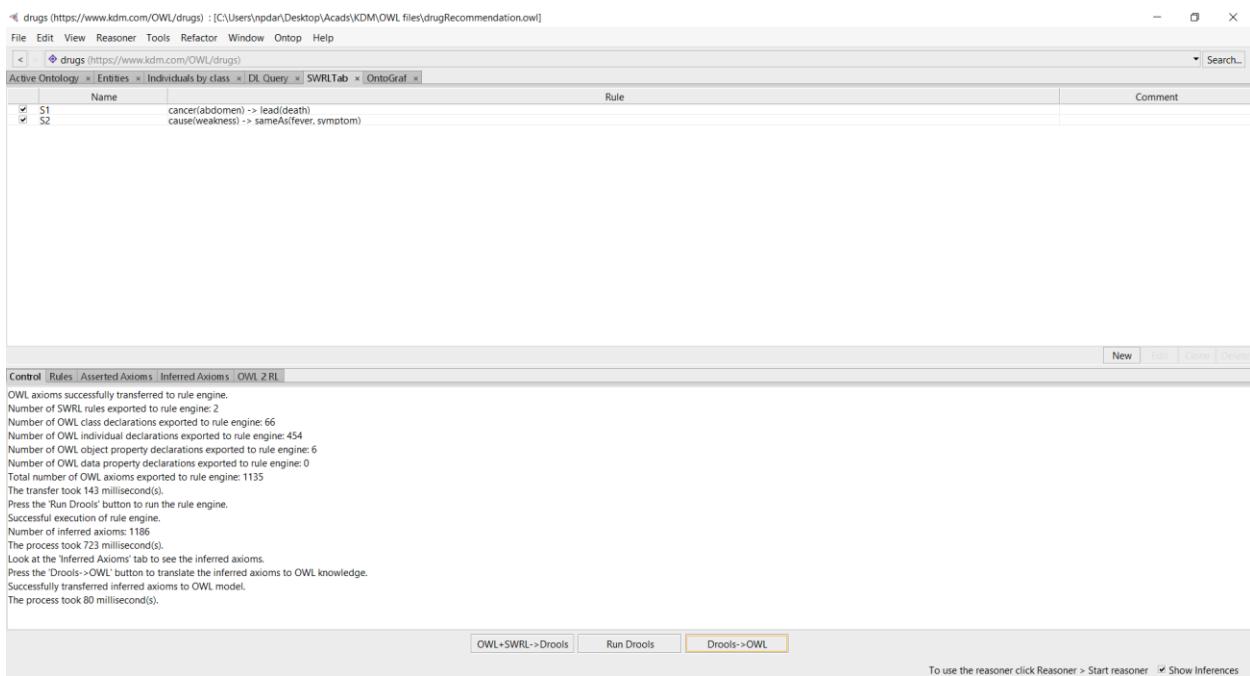
Name	Rule	Comment
S1	cancer(abdomen) -> lead(death)	
S2	cause(weakness) -> sameAs(fever, symptom)	

Control Rules Asserted Axioms Inferred Axioms OWL 2 RL

OWL axioms successfully transferred to rule engine.  
Number of SWRL rules exported to rule engine: 2  
Number of OWL class declarations exported to rule engine: 66  
Number of OWL individual declarations exported to rule engine: 454  
Number of OWL object property declarations exported to rule engine: 6  
Number of OWL data property declarations exported to rule engine: 0  
Total number of OWL axioms exported to rule engine: 1135  
The transfer took 141 milliseconds(s).  
Press the 'Run Drools' button to run the rule engine.  
Successful execution of rule engine.  
Number of inferred axioms: 1186  
The process took 723 milliseconds(s).  
Look at the 'Inferred Axioms' tab to see the inferred axioms.  
Press the 'Drools->OWL' button to translate the inferred axioms to OWL knowledge.

OWL+SWRL->Drools Run Drools Drools->OWL

To use the reasoner click Reasoner > Start reasoner  Show Inferences



drugs (https://www.kdm.com/OWL/drugs) : [C:\Users\npda\Desktop\Acads\KDM\OWL files\drugRecommendation.owl]

File Edit View Reasoner Tools Refactor Window Ontop Help

Active Ontology Entities Individuals by class DL Query SWRLTab OntoGraf

Name	Rule	Comment
S1	cancer(abdomen) -> lead(death)	
S2	cause(weakness) -> sameAs(fever, symptom)	

Control Rules Asserted Axioms Inferred Axioms OWL 2 RL

OWL axioms successfully transferred to rule engine.  
Number of SWRL rules exported to rule engine: 2  
Number of OWL class declarations exported to rule engine: 66  
Number of OWL individual declarations exported to rule engine: 454  
Number of OWL object property declarations exported to rule engine: 6  
Number of OWL data property declarations exported to rule engine: 0  
Total number of OWL axioms exported to rule engine: 1135  
The transfer took 143 milliseconds(s).  
Press the 'Run Drools' button to run the rule engine.  
Successful execution of rule engine.  
Number of inferred axioms: 1186  
The process took 723 milliseconds(s).  
Look at the 'Inferred Axioms' tab to see the inferred axioms.  
Press the 'Drools->OWL' button to translate the inferred axioms to OWL knowledge.  
Successfully transferred inferred axioms to OWL model.  
The process took 80 milliseconds(s).

OWL+SWRL->Drools Run Drools Drools->OWL

To use the reasoner click Reasoner > Start reasoner  Show Inferences

## 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:

```

Spark-TF-IDF - [C:\Users\npdar\Desktop\Acads\KDM\Tutorial\CS5560 Tutorial 5 Source Code\Spark-TF-IDF - [spark-tf-idf - ...\\Article.txt - IntelliJ] IDEA 15.0.3
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Spark-TF-IDF > Article.txt
Project Structure Article.txt
① TFIIDF_Mainscala | ② Article.txt
IRAK :interleukin receptor associated kinase
UBE A :ubiquitin conjugating enzyme E A (RAD homolog)
PRKC1:protein kinase C, iota
COX I :cytochrome c oxidase subunit IV isoform
PRRC1:protein kinase C, eta
UBE I :ubiquitin conjugating enzyme E I (UBC homolog, yeast)
PRRCG:protein kinase C, gamma
UBE H :ubiquitin conjugating enzyme E H (UBC homolog, yeast)
PRKCE:protein kinase C, epsilon
UBE C :ubiquitin conjugating enzyme E C
PRKCD:protein kinase C, delta
UBE B :ubiquitin conjugating enzyme E B (RAD homolog)
ENO B :ENo B;
PROC :protein C (inactivator of coagulation factors Va and VIIa)
PRKD :protein kinase D
UBE N :ubiquitin conjugating enzyme E N (UBC homolog, yeast)
PRKQ:protein kinase C, theta
GHRH :growth hormone releasing hormone
ARCN :archain
COMMD :copper metabolism (Murr ) domain containing
NGFR :nerve growth factor receptor (TNFR superfamily, member )
PRNP :prion protein (p ) (Creutzfeldt Jakob disease, Gerstmann Straussler Scheinker syndrome, fatal familial insomnia)
AOC :amine oxidase, copper containing (retina specific)
UBE S :ubiquitin conjugating enzyme E S
MATR :matrin
AOC :amine oxidase, copper containing (vascular adhesion protein )
MBL :mannose binding lectin (protein C , soluble (opsonic defect)
PRKZ:protein kinase C, zeta
UBE A :ubiquitin protein ligase E A (human papilloma virus E associated protein, Angelman syndrome)
TRA A :TRA A;
GPR :G protein coupled receptor
PRKDC:protein kinase, DNA activated, catalytic polypeptide
HCFC :host cell factor C (VP accessory protein)
AKAP :AKAP :A kinase (PRKA) anchor protein (yotiao)
UBE C :ubiquitin protein ligase E C
SMA // SMA na
NGFB :nerve growth factor, beta polypeptide
GPR :G protein coupled receptor
Terminal Messages Run TODO Event Log
ideaVim: Using the Ctrl+H shortcut for Vim emulation. // You can redefine it as an IDE shortcut or configure its handler in Vim Emulation settings. (15 minutes ago)
55767 CRLF+ UTF-8+

```

### Output:

# **Project Report Increment 2**

The screenshot shows the IntelliJ IDEA interface with the following details:

- Project Structure:** The left sidebar shows the project structure under "spark-TF-IDF".
- Code Editor:** The main window displays the code for `TFIDF_Mainscala.scala`. The code implements a TF-IDF algorithm using Scala and Spark. It includes imports for `SparkSession`, `SQLContext`, `JavaRDD`, and `Map`. The logic involves reading a file, calculating word counts, creating a `tfidfindex` and `tfidfvalues` RDD, and performing a join operation.
- Run Configuration:** A "Run" configuration for "TFIDF\_Main" is visible at the bottom left.
- Toolbars:** Standard IntelliJ toolbars for file operations, navigation, and search are present.
- Status Bar:** The bottom right shows the status bar with "504680 CRLF UFT-8" and "Event Log".

## Top 10 terms:

Spark-TF-IDF - [C:\Users\inpda\Desktop\Acads\KDM\Tutorial\CS5560 Tutorial 5 Source Code\Spark-TF-IDF] - [spark-tf-idf] - ...src\main\scala\TFIDF\_Main.scala - IntelliJ IDEA 15.0.3

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

Spark-TF-IDF Project src main scala TFIDF\_Main.scala

Project Structure

src

Spark-TF-IDF [spark-tf-idf] (C:\Users\inpda\Desktop\Acads\KDM\Tutorial\CS5560 Tutorial 5 Source Code\Spark-TF-IDF) idea project [spark-tf-idf-build] (sources root)

src

main

java

resources

Article.txt

Run TFIDF Main

16/06/24 22:07:38 INFO TaskSetManager: Finished Task 0.0 in stage 8.0 (TID 14) in 24 ms on localhost (1/1)  
16/06/24 22:07:38 INFO TaskSchedulerImpl: Removed TaskSet 8.0, whose tasks have all completed, from pool  
16/06/24 22:07:38 INFO DAGScheduler: ResultStage 8 (take at TFIDF\_Main.scala:56) finished in 0.027 s  
16/06/24 22:07:38 INFO DAGScheduler: Job 5 finished: take at TFIDF\_Main.scala:56, took 0.516311 s  
16/06/24 22:07:38 INFO SparkContext: Invoking stop() from shutdown hook  
(Norel, ALL, 87.2934288651337)  
(ML, 63.4861300837336)  
(TA, 47.6145975628002)  
(ML, ALL, 47.6145975628002)  
(AMD, 47.6145975628002)  
(PRK, 31.7430650418668)  
(TA+, 31.7430650418668)  
(neurokininin, 23.807297814001)  
(Norel, AMD, 15.8715325209334)  
(DS, 15.8715325209334)  
(RBMY, 15.8715325209334)  
(Relax, AMD, 15.8715325209334)  
(GNTL, 15.8715325209334)  
(NOMO, 15.8715325209334)  
(ML, AMD, 15.8715325209334)  
(CTAG, 15.8715325209334)  
(CC, 15.8715325209334)  
(DAZ, 15.8715325209334)  
(AFB, 15.8715325209334)  
(KB, 15.8715325209334)  
16/06/24 22:07:38 INFO SparkUI: Stopped Spark web UI at <http://192.168.6.1:4040>  
16/06/24 22:07:38 INFO MapOutputTrackerMasterEndpoint: MapOutputTrackerMasterEndpoint stopped!  
16/06/24 22:07:38 INFO MemoryStore: MemoryStore cleared

Terminal Run TODO

All files are up-to-date (a minute ago).

Event Log

## Process Success:

## Project Report Increment 2

```

object TFIDF {
    def main(args: Array[String]): Unit = {
        val rdd = sc.textFile("Article.txt")
        val words = rdd.flatMap(_.split(" "))
        val wordCount = words.map((_, 1)).reduceByKey(_ + _)
        val totalWords = wordCount.sum
        val wordProbabilities = wordCount.map{ case (word, count) => (word, count / totalWords)}
        val tfidf = wordProbabilities.map{ case (word, probability) => (word, log((totalWords / wordCount.get(word)) * probability))}
        tfidf.foreach(println)
    }
}

val tfidfData = tfidfIndex.zip(tfidfvalues)
var hm = new HashMap[String, Double]
tfidfData.collect().foreach(f => {
    hm += f._1 -> f._2.toDouble
})
val mapp = sc.broadcast(hm)

```

Process finished with exit code 0

### Word Count:

#### Input file:

```

object SparkWordCount {
    def main(args: Array[String]): Unit = {
        val inputPath = "input/part-00000"
        val outputPath = "output/_SUCCESS"
        val wordCount = sc.textFile(inputPath).flatMap(_.split(" ")).map((_, 1)).reduceByKey(_ + _).collect()
        wordCount.foreach(println)
    }
}

```

NAME	DESCRIPTION	ALL	TA+	Norel	ALL	TA+	(ML)	Norel	ALL	Norel	ALL	TA	Norel	ALL	TA	Norel	ALL	TA	Norel	ALL
ALPP	ALPLP	alkaline phosphatase, placental like																		
GFER	GFER	growth factor, augmented of liver regeneration (ERV homolog, S cerevisiae)																		
IGHM	IGHM	immunoglobulin heavy constant mu																		
SART	SART	:squamous cell carcinoma antigen recognised by T cells																		
RPLP	RPLP	ribosomal protein, large, P																		
RPLD	RPLD	ribosomal protein, large, P																		
PDXA	PDXA	:pyruvate dehydrogenase (lipoyamide) alpha																		
OGT	OGT	:linked N acetylglucosamine (GlcNAc) transferase (UDP N acetylglucosamine:polypeptide N acetylglucosaminyl transferase)																		
PDXA	PDXA	:pyruvate dehydrogenase (lipoyamide) alpha																		
CYD	CYD	// CYP D na																		
ACAA	ACAA	:acetyl Coenzyme A acyltransferase (mitochondrial oxaoyl Coenzyme A thiolase)																		
TMEFF	TMEFF	:transmembrane protein with EGF like and two follistatin like domains																		
LRRK	LRRK	:leucine rich repeat containing																		
PCBD	PCBD	:pterin alpha carbonylamine dehydratase/dimerization cofactor of hepatocyte nuclear factor alpha (TCF)																		
UGCG	UGCG	:UDP glucose ceramide glucosyltransferase																		
GEN	GEN	:GTP binding protein overexpressed in skeletal muscle																		
JTV	JTV	:																		
DOC	DOC	A DOC A:double C like domains, alpha																		
TAGLN	TAGLN	:transgelin																		
IGHD	IGHD	:immunoglobulin heavy constant delta																		
TFPI	TFPI	:tissue factor pathway inhibitor (lipoprotein associated coagulation inhibitor)																		
HSD	HSD	B :hydroxysteroid ( beta) dehydrogenase																		

Process finished with exit code 0

#### Output file:

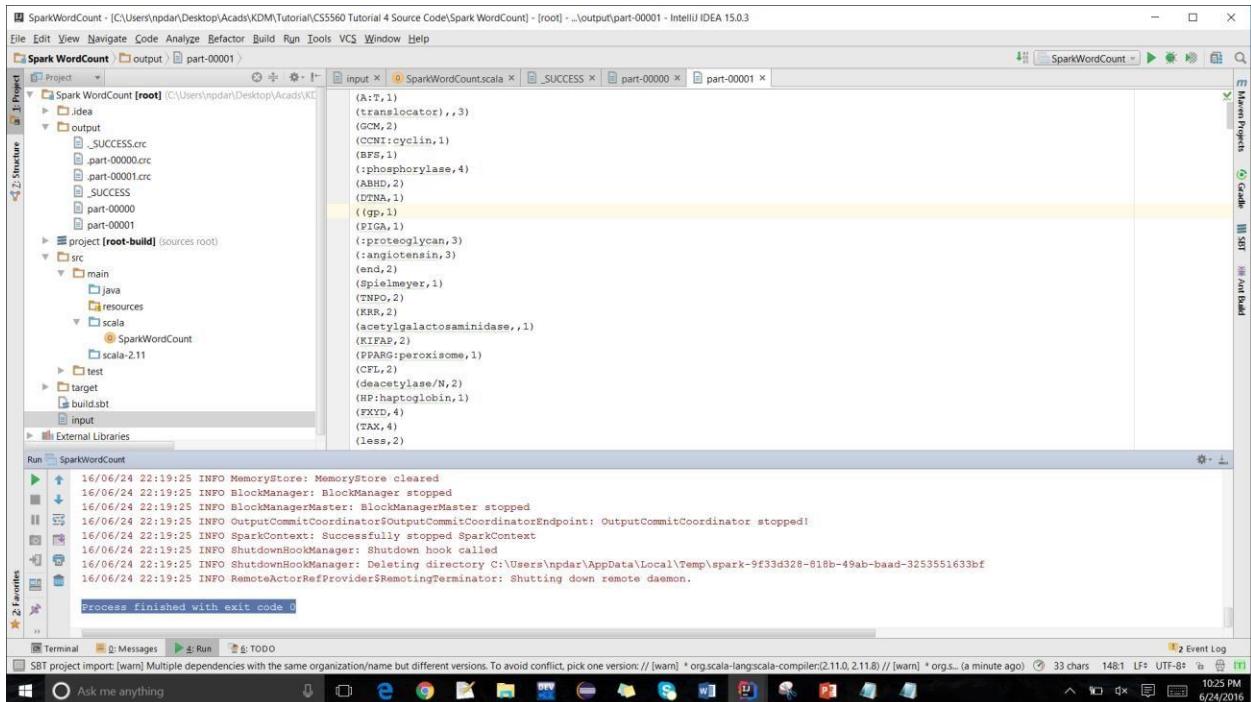
## Project Report Increment 2

The screenshot shows the IntelliJ IDEA interface with the following details:

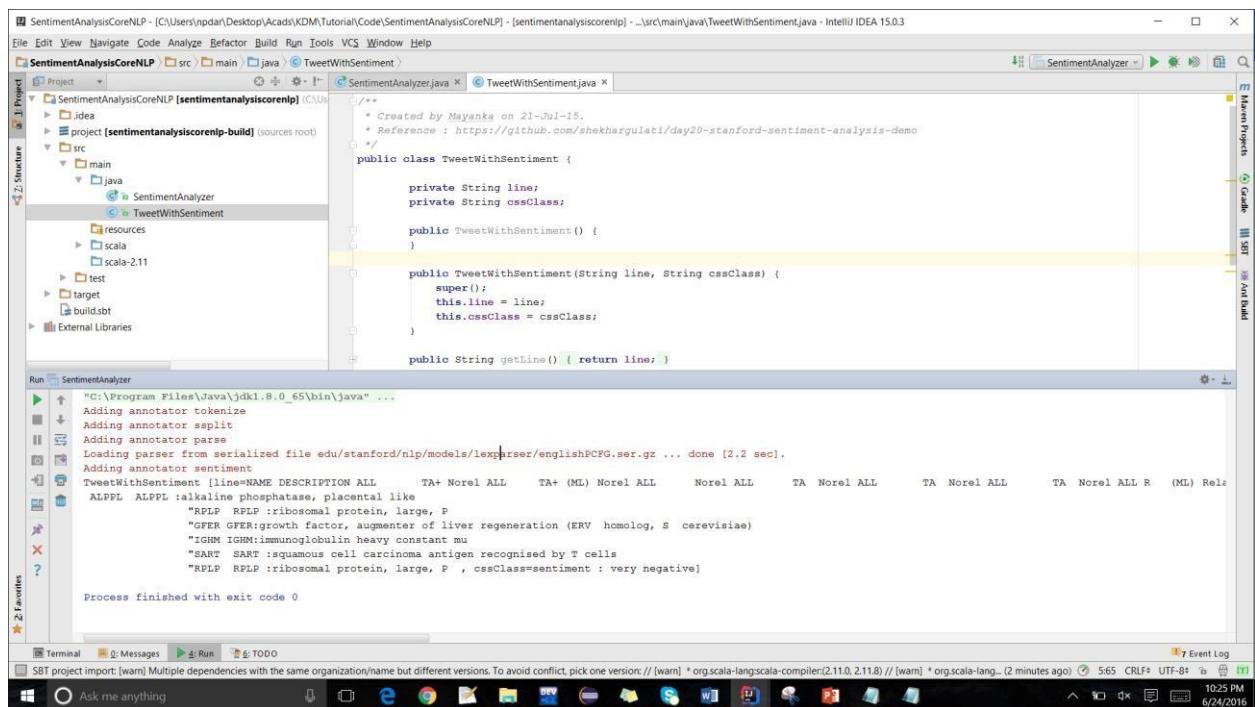
- Project Structure:** The project is named "Spark WordCount". It contains a "src" directory with "main", "java", "resources", and "scala" sub-directories. The "scala" directory contains a "SparkWordCount" object and a "scala-2.11" sub-directory.
- Run Output:** The terminal window shows the execution of the application. The logs indicate the application has cleared memory, stopped the BlockManager and BlockManagerMaster, and successfully stopped the SparkContext. It also shows the shutdown hook was called and the temporary directory was deleted.
- Event Log:** The event log shows a warning about multiple dependencies with the same organization/name but different versions.
- System Bar:** The taskbar at the bottom shows various application icons.

This screenshot is similar to the one above, but the run output shows a different set of log entries, indicating a successful execution of the application. The logs include the clearing of memory, stopping of the BlockManager and BlockManagerMaster, successful stopping of the SparkContext, and the deletion of the temporary directory.

### Process execution:



## Sentimental Analysis:



## **8 PROJECT MANAGEMENT**

---

### **8.1 CONTRIBUTION:**

Overall – 100%

Venkata Vamsi Krishna Bhuvanam – 25%

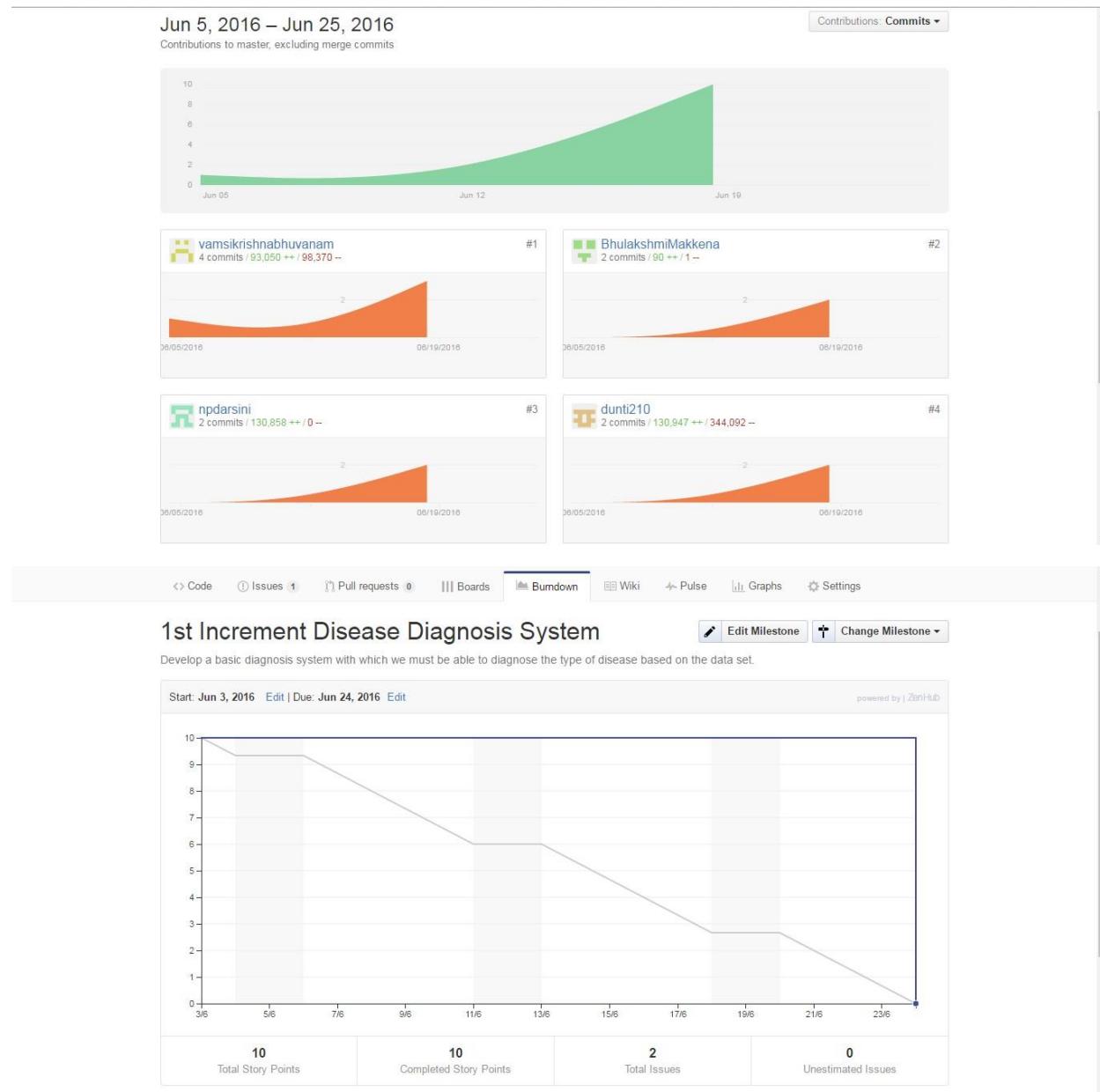
Priyadarsini Nidadavolu – 25%

Bhulakshmi Makkenna – 25%

Tej Kumar Yentrapragada – 25%

## 8.2 ZENHUB AND GITHUB:

### 8.2.1 First Increment:



## Project Report Increment 2

This repository | Search Pull requests Issues Gist ToDo

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

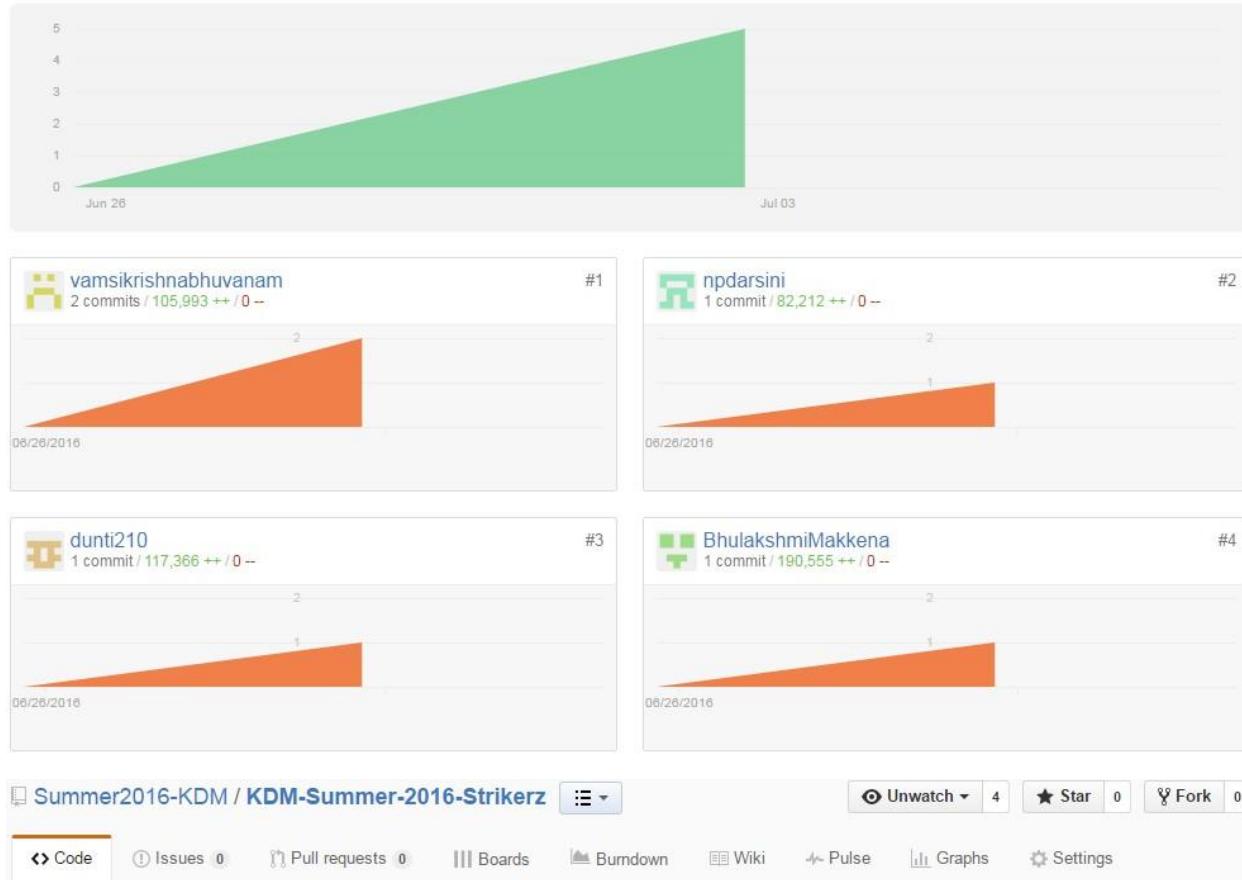
1stIncrement Commit 4 minutes ago

README.md Initial commit 17 days ago

**KDM-Summer-2016-ProjectDraft**

## 8.2.2 SECOND INCREMENT:

Contributions to master, excluding merge commits



Created a new Repository because of contribution issues in Earlier Repo. This repository consists from 2nd Increment. — Edit

The repository page shows the following information:

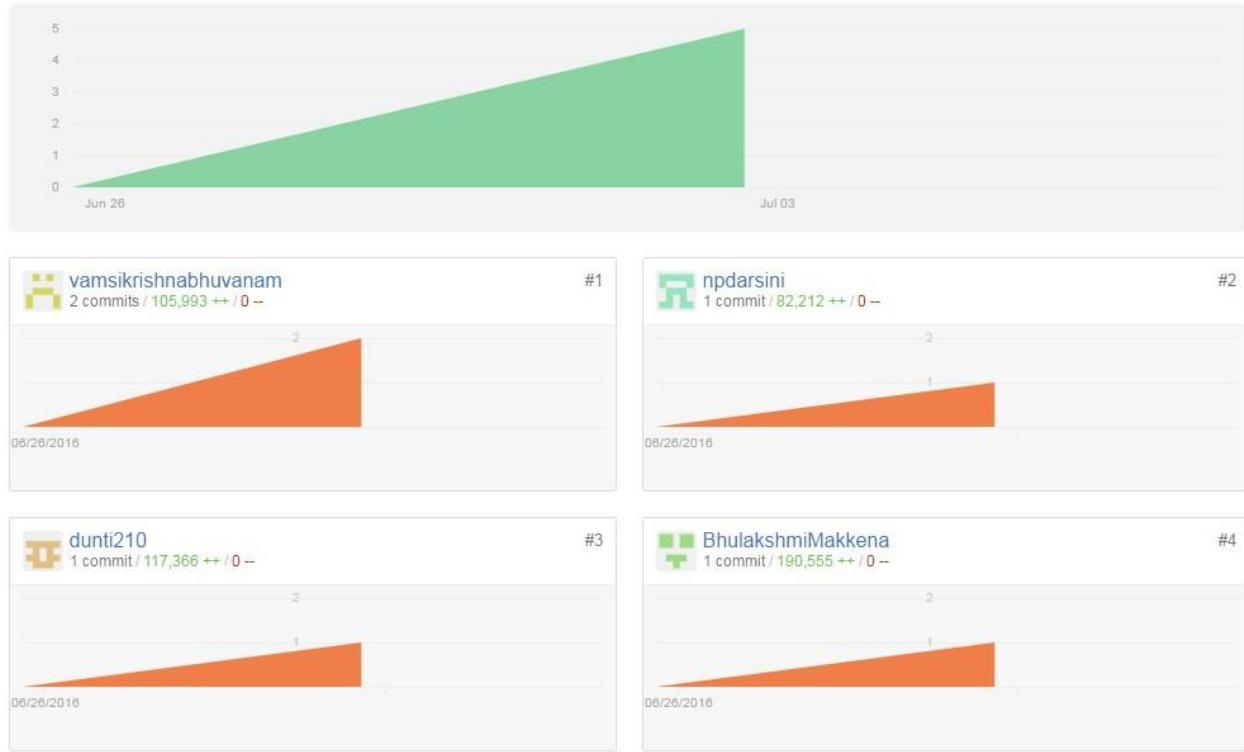
- Stats:** 5 commits, 1 branch, 0 releases, 1 contributor
- Branch:** master
- Files:**
  - npdarsini TFIDF->W2Vec-> WordNet dictionary (Latest commit e394bf0 6 minutes ago)
  - 2nd Increment/Source (TFIDF->W2Vec-> WordNet dictionary 6 minutes ago)
  - README.md (Initial commit an hour ago)
  - README.md (Current view)
- Actions:** Create new file, Upload files, Find file, Clone or download
- README Content:**

**KDM-Summer-2016-Strikerz**

Created a new Repository because of contribution issues in Earlier Repo. This repository consists from 2nd Increment.

### 8.2.3 THIRD INCREMENT

Contributions to master, excluding merge commits



### 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.

#### **8.5 REFERENCES:**

Dataset: Drugs.com Articles.