Name: Lakshmi Korrapati

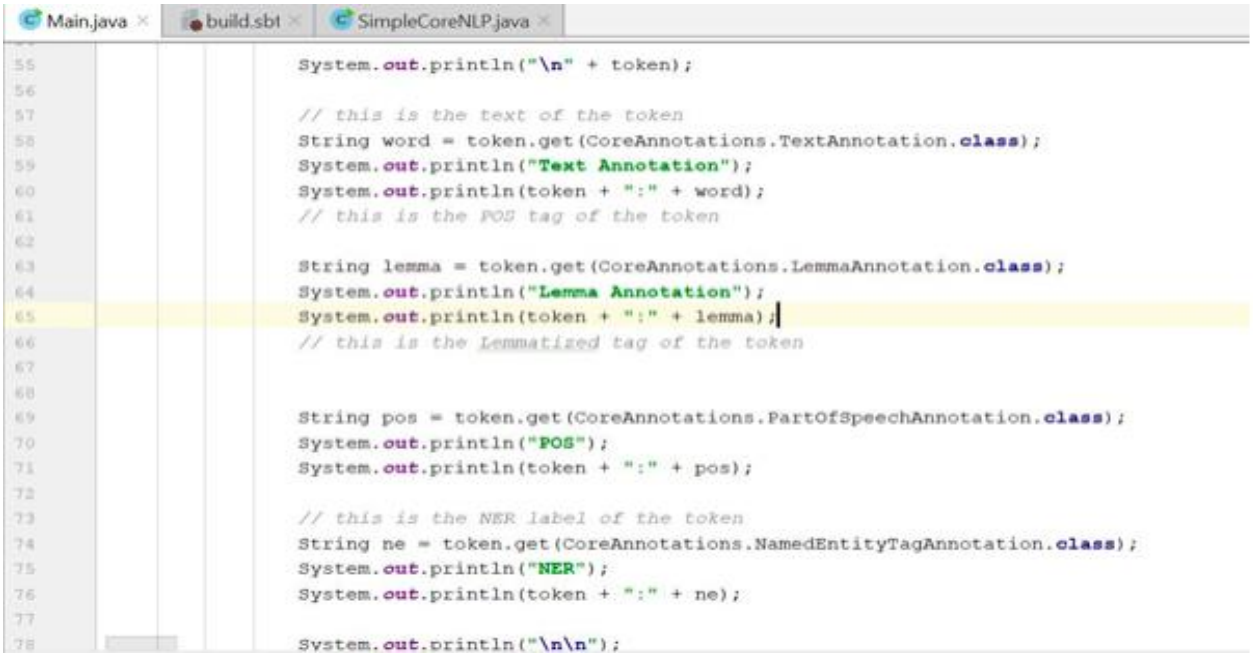
Class ID: 4

Use the program RetrieveAbstract3.java to obtain the abstracts in XML format and convert to textdata. After this, we combine the 5 abstracts data into one file which is one.txt.

Now we will be performing the below tasks

- Part-of-speech (POS) tagger
- Named entity recognizer (NER)
- Co-reference resolution system
- Word Detection
- Triplet Extraction

Core NLP: Source Code



OpenIE:
We will use the OpenIE code to perform the triplet extraction as shown below

```scala
WordNetSpark.scala | SparkOpenIE.scala | 1.txt

 */
object SparkOpenIE {

  def main(args: Array[String]) : Unit = {
    // Configuration
    val sparkConf = new SparkConf().setAppName("SparkWordCount").setMaster("

    val sc = new SparkContext(sparkConf)

    // For Windows Users
    System.setProperty("hadoop.home.dir", "C:\\winutils")


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

    val input = sc.textFile( path = "data/pubmed/1.txt").map(line => {
      //Getting OpenIE Form of the word using lda.CoreNLP

      val t=CoreNLP.returnTriplets(line)
      t
    })

    //println(CoreNLP.returnTriplets("The dog has a lifespan of upto 10-12 y
  println(input.collect().mkString("\n"))
```

Triplets extraction:

```
Run: SparkOpenIE

[(Children,learning from,their parents,1.0)]
[(We,model,good health,1.0), (We,model,health,1.0), (We,model,good environmental health,1.0), (We,model,environmental health,1.0)][(We,model,good health,1.0), (We,model,health,1.0), (We,mode

[(Mold,can form easily in,two rooms,1.0), (Mold,can form in,two rooms,1.0), (Mold,can form very easily in,two rooms,1.0), (germs,can form easily in,two rooms,1.0), (germs,can form in,two roo
[(You,are in,environment,1.0), (You,are in,environment right now,1.0), (You,are in,environment now,1.0)][(You,are in,environment,1.0), (You,are in,environment right now,1.0), (You,are in,env
[(you,is,anywhere are,1.0), (you,is,are,1.0), (It,is,are,1.0), (It,is,anywhere are,1.0)][(you,is,anywhere are,1.0), (you,is,are,1.0), (It,is,are,1.0), (It,is,anywhere are,1.0)][(you,is,anywh
[(work,place of are,environments,1.0), (Your home,are,environments,1.0), (place,are,environments,1.0)][(work,place of are,environments,1.0), (Your home,are,environments,1.0), (place,are,envi
[(your environment,Is,healthy,1.0)]
[(Pollution,can,can found anywhere,1.0), (Pollution,can,can found,1.0)][(Pollution,can,can found anywhere,1.0), (Pollution,can,can found,1.0)]
[(Pollution,is,bad,1.0), (Pollution,is bad for,your health,1.0)][(Pollution,is,bad,1.0), (Pollution,is bad for,your health,1.0)]
[(dirt,are,bad for your health,1.0), (Garbage,are bad for,your health,1.0), (mold,are,bad,1.0), (dust,are bad for,your health,1.0), (dust,are,bad,1.0), (Garbage,are,bad for your health,1.0),
[(They,can cause,allergies,1.0), (They,can cause,sickness,1.0), (They,can all cause,allergies,1.0), (They,can all cause,sickness,1.0)][(They,can cause,allergies,1.0), (They,can cause,sicknes
[(They,can also attract,pests,1.0), (They,can attract,pests,1.0)][(They,can also attract,pests,1.0), (They,can attract,pests,1.0)]

4: Run | 6: TODO | SBT Console | sbt shell | Terminal | Java Enterprise | 0: Messages | Event Log
```

WordNet:

WordNetSpark.scala | WordNetMain.java | CoreNLP.java | SparkOpenIE.scala | build.sbt

```scala
/**
  * Created by Mayanka on 26-06-2017.
  */
object WordNetSpark {
  def main(args: Array[String]): Unit = {
    System.setProperty("hadoop.home.dir", "D:\\winutils")
    val conf = new SparkConf().setAppName("WordNetSpark").setMaster("local[*]").set("spark.driver.memory", "4g").set("spark.executor.memory
    val sc = new SparkContext(conf)


    val data=sc.textFile( path = "data/sample")

    val dd=data.map(line=>{
      val wordnet = new RiWordNet("/Users/lakshmikorrapati/Downloads/WordNet-3.0")
      val wordSet=line.split( regex = " ")
      val synarr=wordSet.map(word=>{
        if(wordnet.exists(word))
          (word,getSynoymns(wordnet,word))
        else
          (word,null)
      })
      synarr
    })
    dd.collect().foreach(linesyn=>{
      linesyn.foreach(wordssyn=>{
        if(wordssyn._2 != null)
          println(wordssyn._1+":"+wordssyn._2.mkString(","))
      })
```

```
by:aside,away,past
alabama:AL,Alabama,Alabama River,Camellia State,Chahta,Chickasaw,Choctaw,Heart of Dixie,Hitchiti,Koasati
in:440 yards,880 yards,Al,Am,Ba,Be,Bi,Bk,Ca,Cd
one:1,ane,cardinal,combined,extraordinary,i,incomparable,indefinite,matchless,nonpareil
nine:0,1,2,3,4,5,6,7,8,9
six:0,1,2,3,4,5,6,7,8,9
eight:0,1,2,3,4,5,6,7,8,9
alabama:AL,Alabama,Alabama River,Camellia State,Chahta,Chickasaw,Choctaw,Heart of Dixie,Hitchiti,Koasati
supported:based,braced,buttressed,dependent,gimbaled,pendant,pendent,subsidised,subsidized,supernatant
native:aboriginal,autochthonal,autochthonic,autochthonous,connatural,domestic,endemic,homegrown,inborn,inbred
son:Jnr,Jr,Junior,Logos,Son,Word,boy,mama's boy,mamma's boy,mother's boy
american:AAVE,African American,African American English,African American Vernacular English,African-American,Afro-American,Alabaman,Alabamian,Alaskan,Alsatian
independent:autarkic,autarkical,autonomous,breakaway,case-by-case,commutative,fissiparous,free,free-living,indie
party:African,American Federalist Party,American Labor Party,American Party,Amerindian,Anti-Masonic Party,Aquarius,Archer,Aries,Balance
```

NCBI –BioNLP and Bioportal API and used only BioNLP to find extractions

build.sbt | ex.pmid | RESTClientGet.java

```java
import ...

public class RESTClientGet {
    public static void main(String[] args)
    {
        if(args.length<2)
        {
            System.out.println("\n$ java RESTClientGet [Bioconcept] [Inputfile] [Format]");
            System.out.println("\nBioconcept: We support five kinds of bioconcepts, i.e., Gene, Disease, Chemical, Spe
        }
        else
        {
            String Bioconcept=args[0];
            String Inputfile=args[1];
            String Format="PubTator";
            if(args.length > 2)
            {
                Format=args[2];
            }

            try {

                //pmids
                BufferedReader fr= new BufferedReader(new FileReader(Inputfile));
                String pmid = "";
                while((pmid = fr.readLine()) != null)
                {
                    URL url_Submit;
                    url_Submit = new URL( spec: "https://www.ncbi.nlm.nih.gov/CBBresearch/Lu/Demo/RESTful/tmTool.cgi/"
                    HttpURLConnection conn_Submit = (HttpURLConnection) url_Submit.openConnection();
                    conn_Submit.setDoOutput(true);
                    BufferedReader br_Submit = new BufferedReader(new InputStreamReader(conn_Submit.getInputStream()))
                    String line="";
                    while((line = br_Submit.readLine()) != null)
                    {
                        System.out.println(line);
                    }
                    conn_Submit.disconnect();
                }
```