**CS5560**

**Knowledge Discovery and Management**

**Project Team: 8**

**Phase-3 Report**

**Team Members:**

**Megha Nagabhushan**

**Rohith Kumar Nagulapati**

**Motivation:**

In the present data centric world, about 80% of the total data is semi structured and is difficult to derive information from these sources in its raw form. This fact motivated us in developing a dynamic question and answering system which is the most welcoming model in all the fields as this can be one’s personal assistant in responding to user queries. Also, this model finds worthy in every use case where we require some help from computer.

**Objective:**

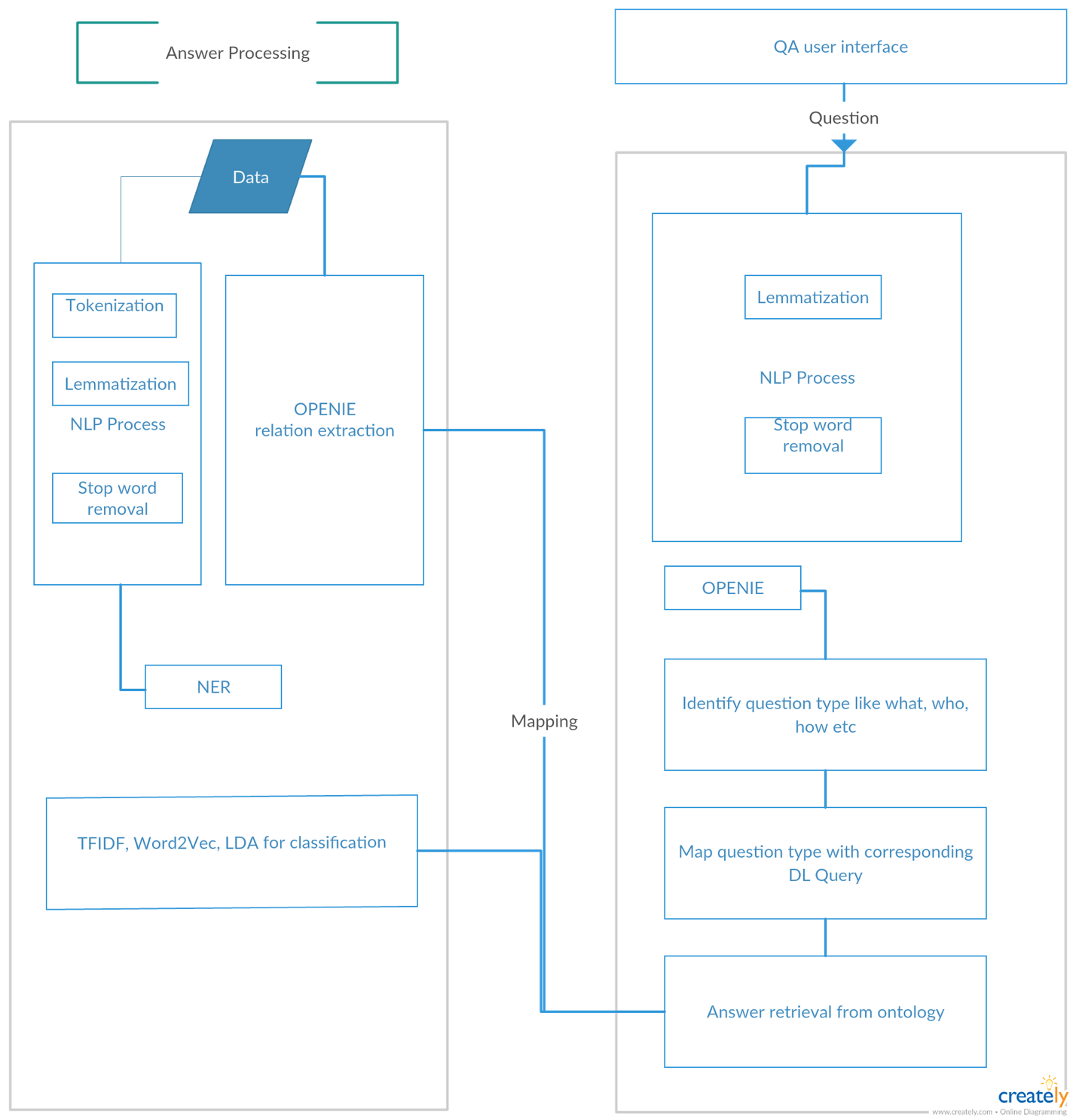
The main objective of this project is to extract information from a huge collection of medical data sources using latest techniques of natural language processing like lemmatization, named entity recognition and using OpenIE, Word2Vec, WordNet for extracting relationships, using clustering and classification techniques for topic discovery. Finally, to represent the obtained information in the form of ontology which then becomes available to question and answering system to query the results from the system.

**Dataset:**

## For this project, we have taken the NCI-PID-PubMed Genomics Knowledge Base Completion Dataset. From this Data set we try to construct knowledge graph and making system dynamic to answer all possible questions on medical domain related questions.

**Workflow and Implementation:**

The complete system architecture is as shown in the below figure.

****

1. **Natural Language Processing(NLP)**

**NLP** in the field of computer science is helpful in giving computer the ability to understand human speech as it is spoken. The most common tasks include tokenization, Lemmatization, Parts-of-Speech Tagging, Named Entity Recognition and breaking documents to sentences. It is helpful in pre-processing raw text data.

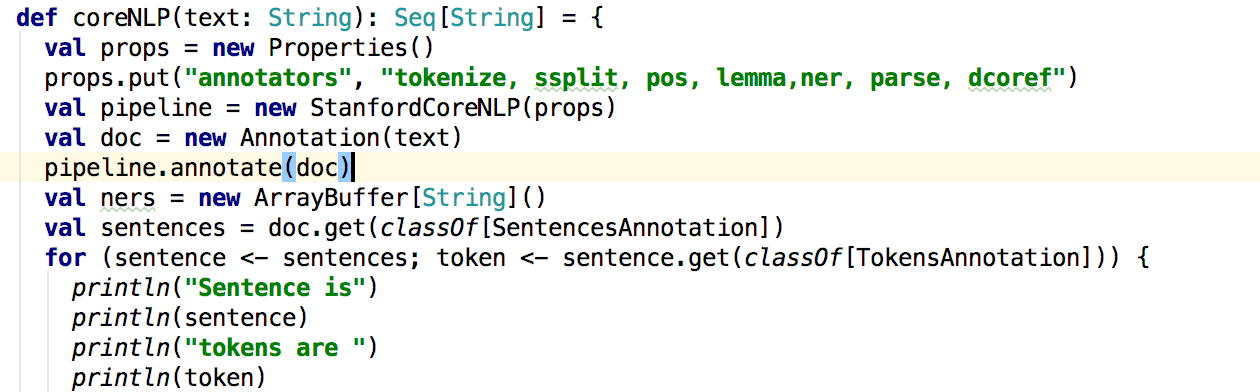
* 1. **Tokenization**

Tokenization is the process of breaking sentences in to tokens which are the smallest constructs of a huge text data.

Sentence: South Africa far too strong again

Tokens: South, Africa, far, too, strong, again

Code snippet:



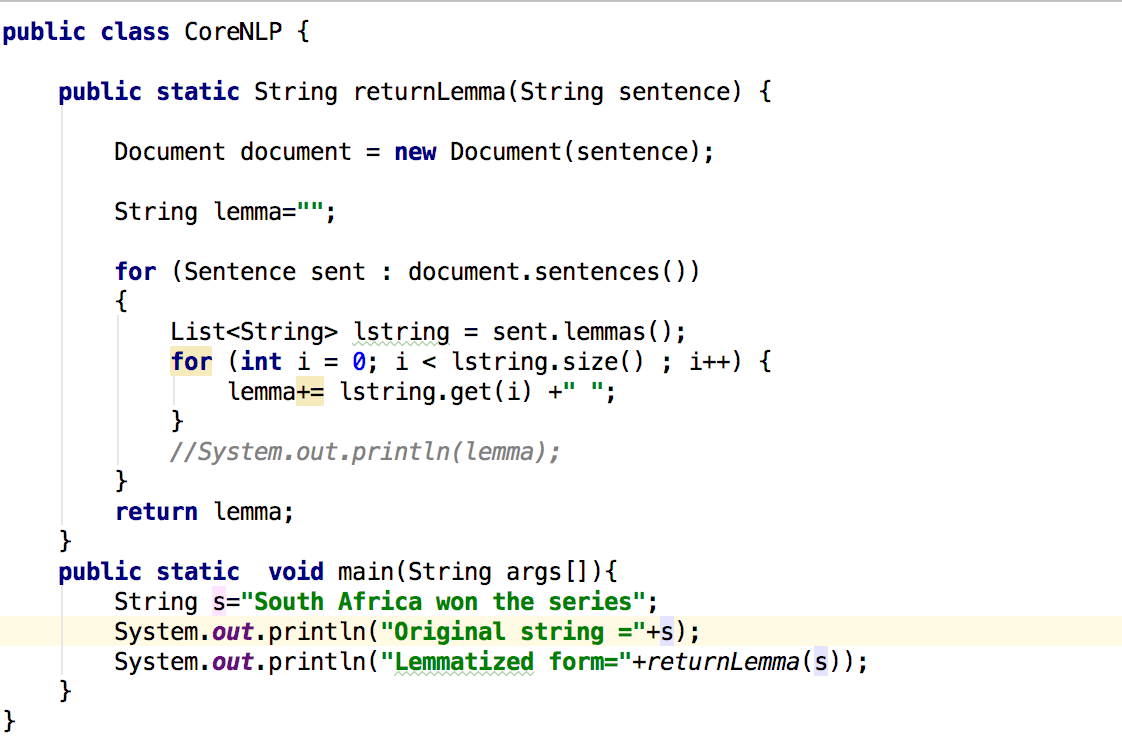
* 1. **Lemmatization**

Lemmatization is the process of separating words into individual [morphemes](https://en.wikipedia.org/wiki/Morpheme) and identify the class of the morphemes.

Original string =South Africa won the series

Lemmatized form=South Africa win the series

Code Snippet



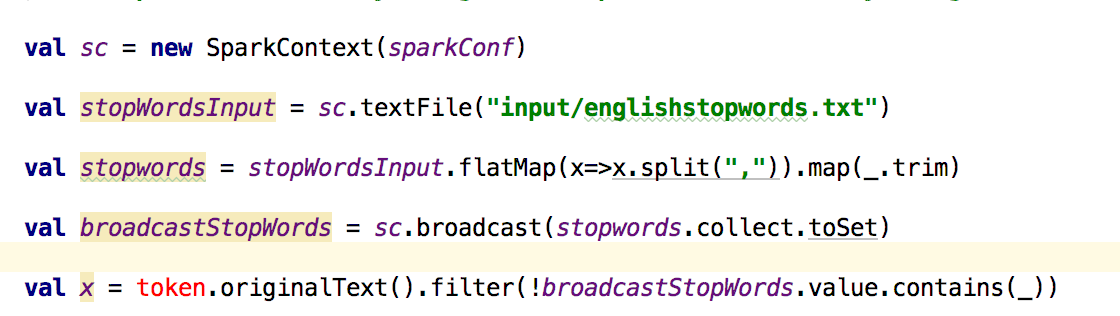
* 1. **Stop word Removal**

Stop word Removal is the process of removing stop words from the data.

**For example,** the stop words in English be:

able, about, above, according, accordingly, across, actually, after, afterwards, again.

Code snippet:



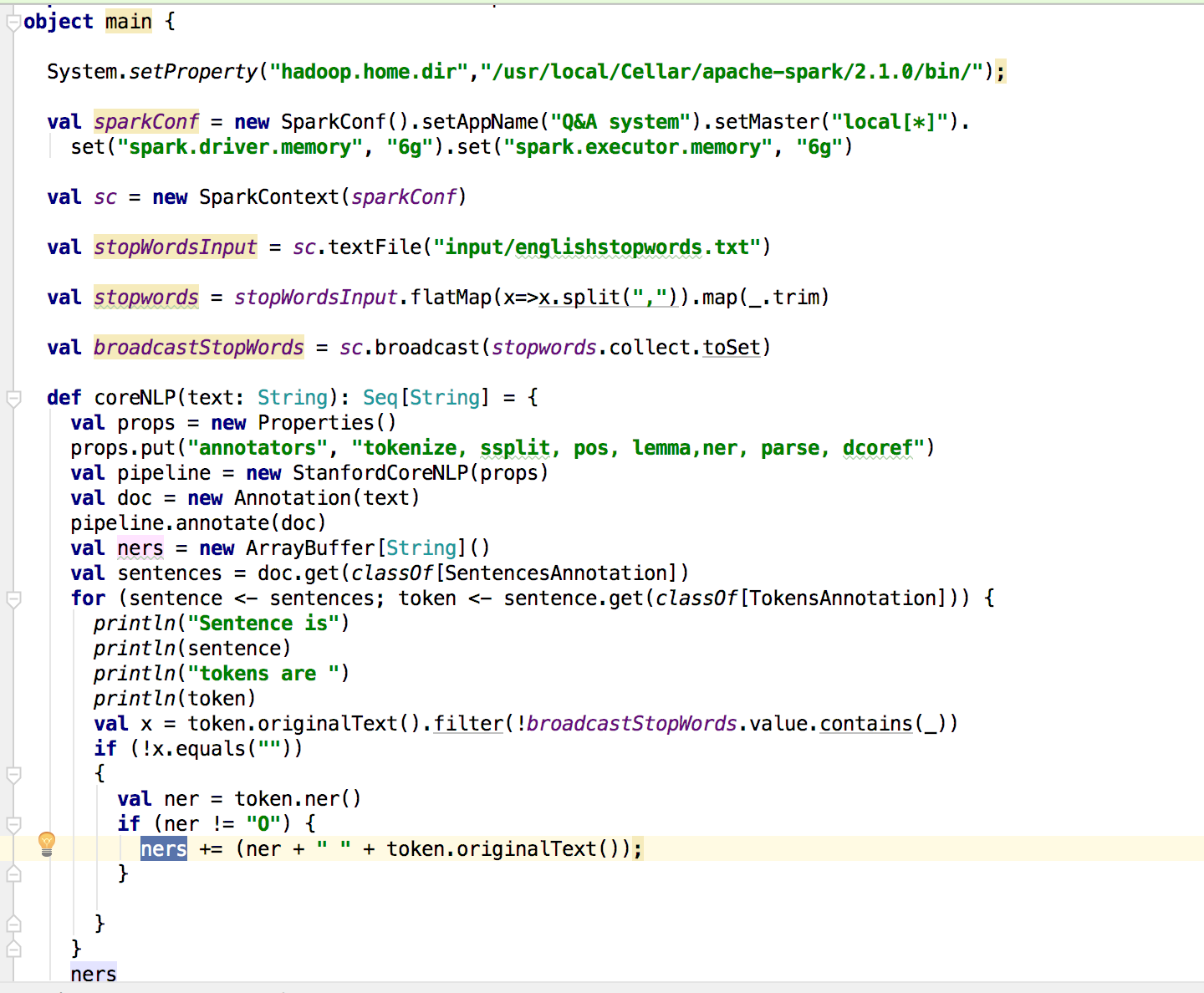
* 1. **Named Entity Recognition**

Given a collection of text data, NER determines which items in the text map to proper names, such as people or places, and what the type of each such name is e.g. person, location, organization etc.

PERSON LOCATION NUMBERS

Hayden South Africa 281-9  
 Ricky Australia 2-0  
 Ponting New Zealand 60

Code Snippet :



1. **OpenIE relation Extraction**

An Open IE system runs over sentences and creates extractions that represent relations in text.

For example, consider the following sentence:

The U.S. president Barack Obama gave his speech on Tuesday.

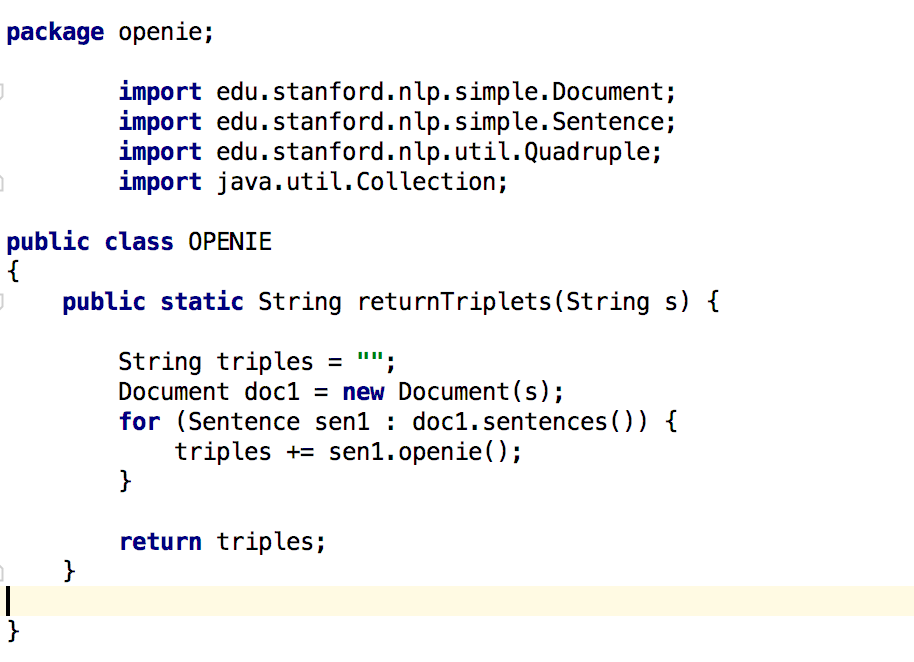
There are many binary relations in this sentence that can be expressed as a triple (A, B, C) where A and B are arguments, and C is the relation between those arguments.

Here is a possible list of the binary relations in the above sentence:

(Barack Obama, is the president of, the U.S.)

(Barack Obama, gave, his speech)

(Barack Obama, gave his speech, on Tuesday)

****

1. **TF-IDF(TermFrequency- Inverse Document Frequency)**

In [information retrieval](https://en.wikipedia.org/wiki/Information_retrieval) system, TF-IDF, is a numerical statistic and a weighing factor that is used to reflect how important a word is to a [document](https://en.wikipedia.org/wiki/Document) in a collection or [corpus](https://en.wikipedia.org/wiki/Text_corpus).

The TF-IDF is the product of Term Frequency and Inverse Document Frequency.

Where

1. Term Frequency is the number of times a word in a document.
2. Inverse Document Frequency is the number of documents that contains this word.

**TF-IDF is calculated as tf-idf (t, d, D) = tf (t, d). idf (t, D)**

**Code Snippet**

****

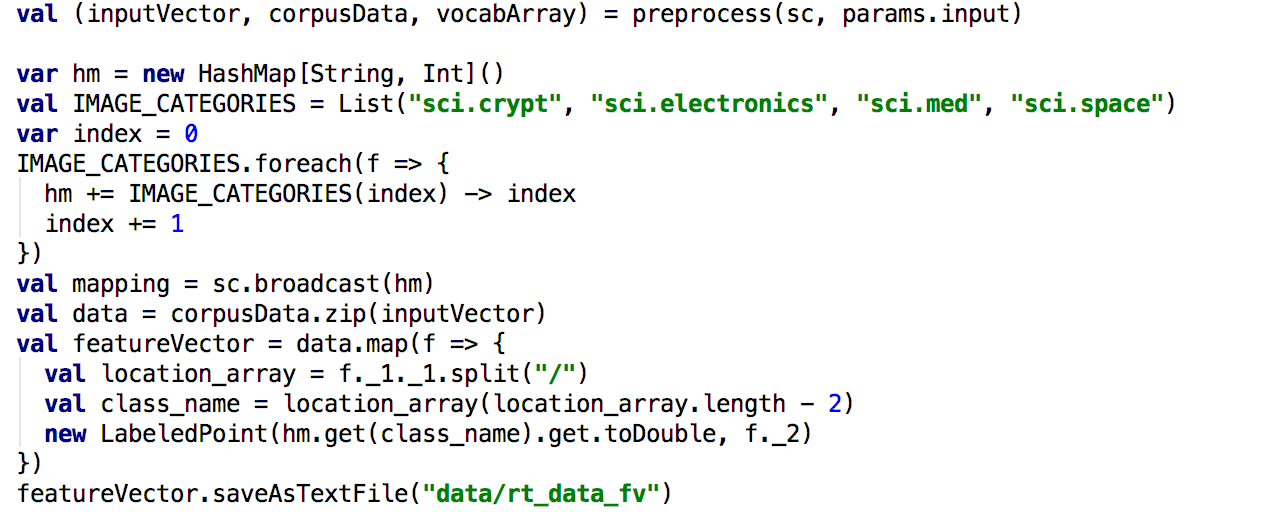
1. **Word2Vec**

Word2Vec is a shallow two-layer neural network that are trained to reconstruct linguistic contexts of words. The Word2Vec constructs a vector space of n dimensions taking a large corpus of text as input where each unique word in the corpus is assigned a corresponding vector in the space.

1. **Feature Vector**

In machine learning feature vector is a n-dimensional vector of numerical features that represent some object used for processing and performing statistical procedures such as linear regression. They are often combined with weights using a dot product to construct a linear prediction function for making a better prediction.

Code Snippet:

****

1. **K-means clustering**

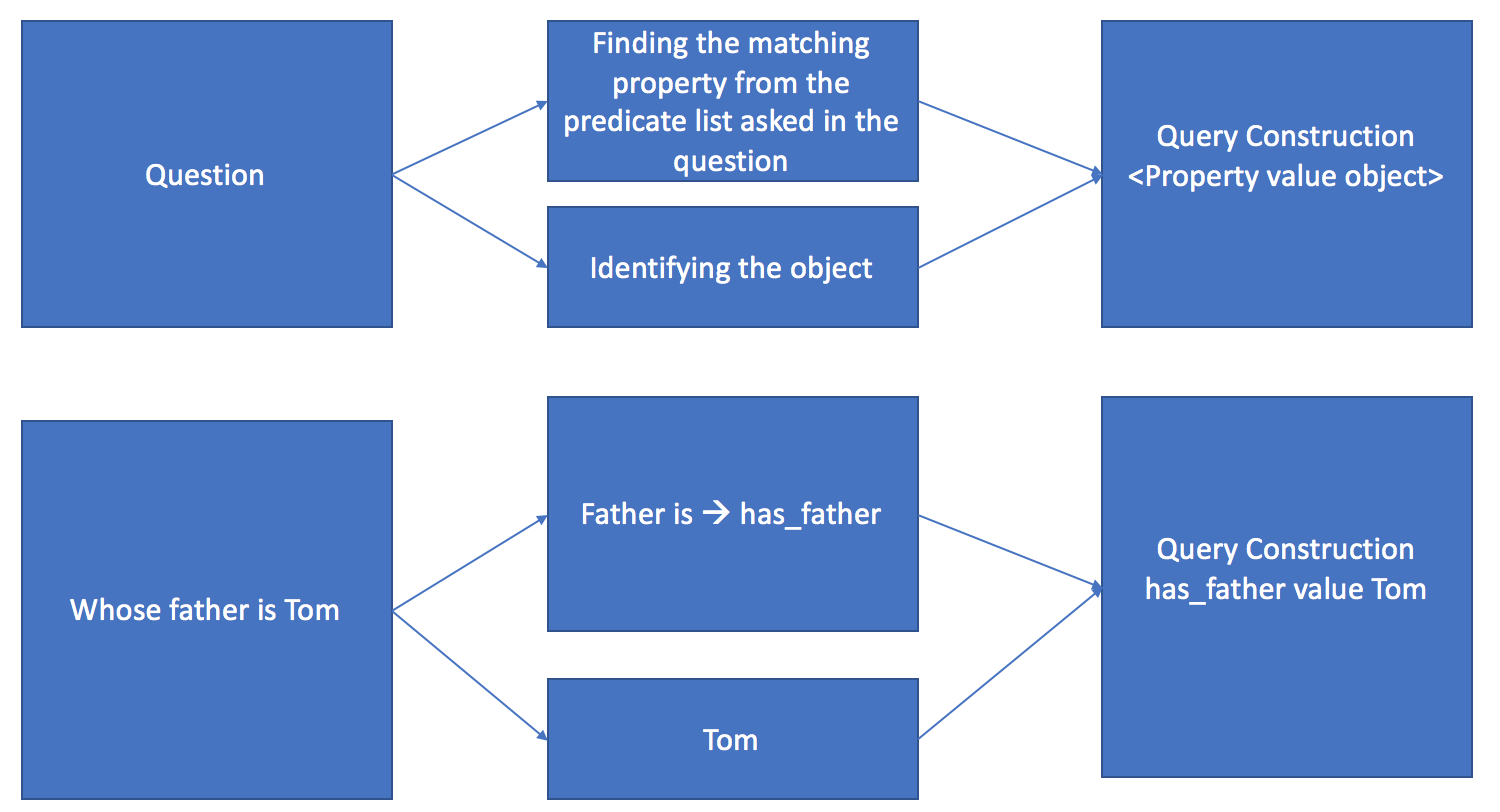
k-means clustering is a popular clustering technique that aims to partition ‘n’ observations into ‘k’ clusters in which each observation belongs to the cluster with the nearest mean, serving as being the centroid of the cluster.

Code Snippet:



1. **Descriptive Logic query (DL query)**

DL query is an owl querying language and is based on the Manchester OWL syntax that represents the ontology’s class, property, and individuals into a single construct, called frame. It is possible to use DL query only on a classified ontology.





1. **Spark Play framework**

Play is a high-productivity Java and Scala web application framework that integrates the components and APIs you need for modern web application development. Also, this is a lightweight, stateless, web-friendly architecture and features predictable and minimal resource consumption (CPU, memory, threads) for highly-scalable applications.