SESSION 2015-1016 (Minor project)

TOPIC MODELING USING BIG DATA ANALYTICS

PRESENTED BY:
Farheen Nilofer (12-CSS-21)
Sarah Masud(12-CSS-57)

BACKGROUND:

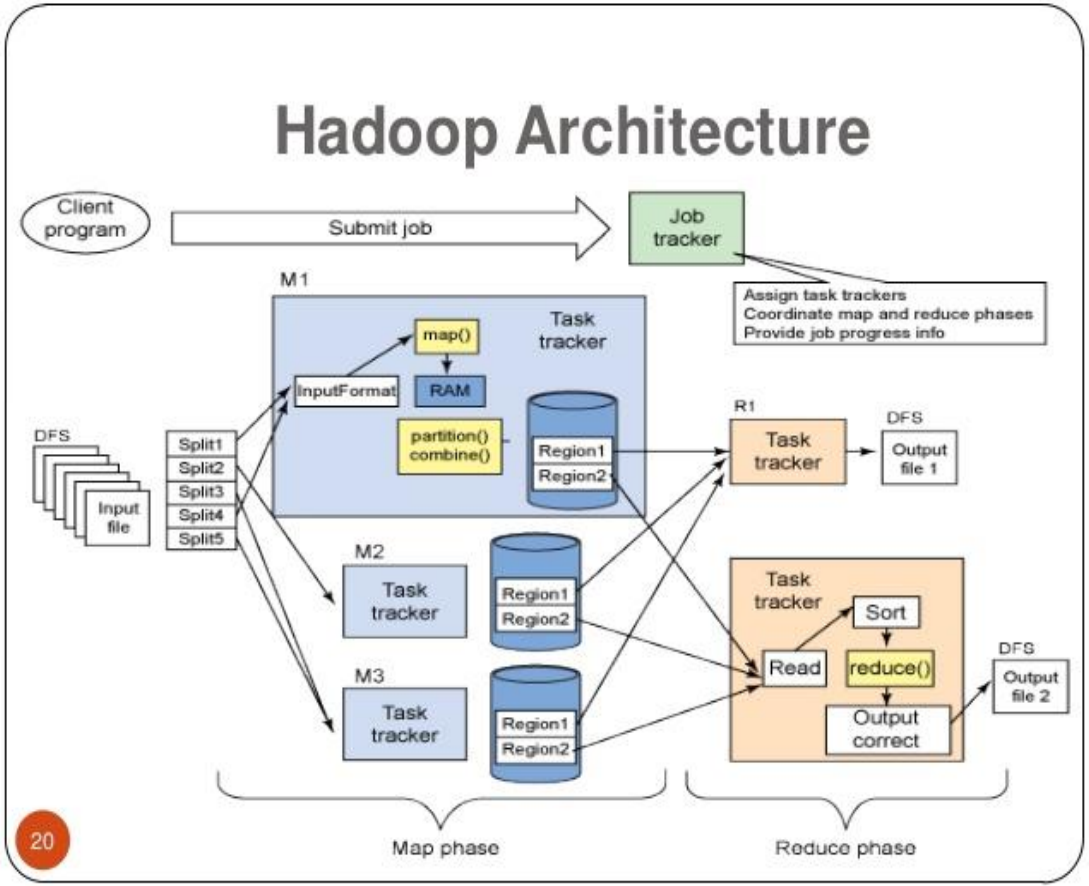
Hadoop: Hadoop is a framework that allows for the distributed processing of large datasets across cluster of computers using computing paradigm like MapReduce.

MapReduce: MapREduce is a programming model processing and generating large datasets with parallel and distributed algorithm on a cluster.

HDFS:A distributed Java-based filesystem for storing large volumes of data.

FEATURES:

- Highly Fault tolerant
- Deployed on low cost hardware
- High throughput for application having large data sets



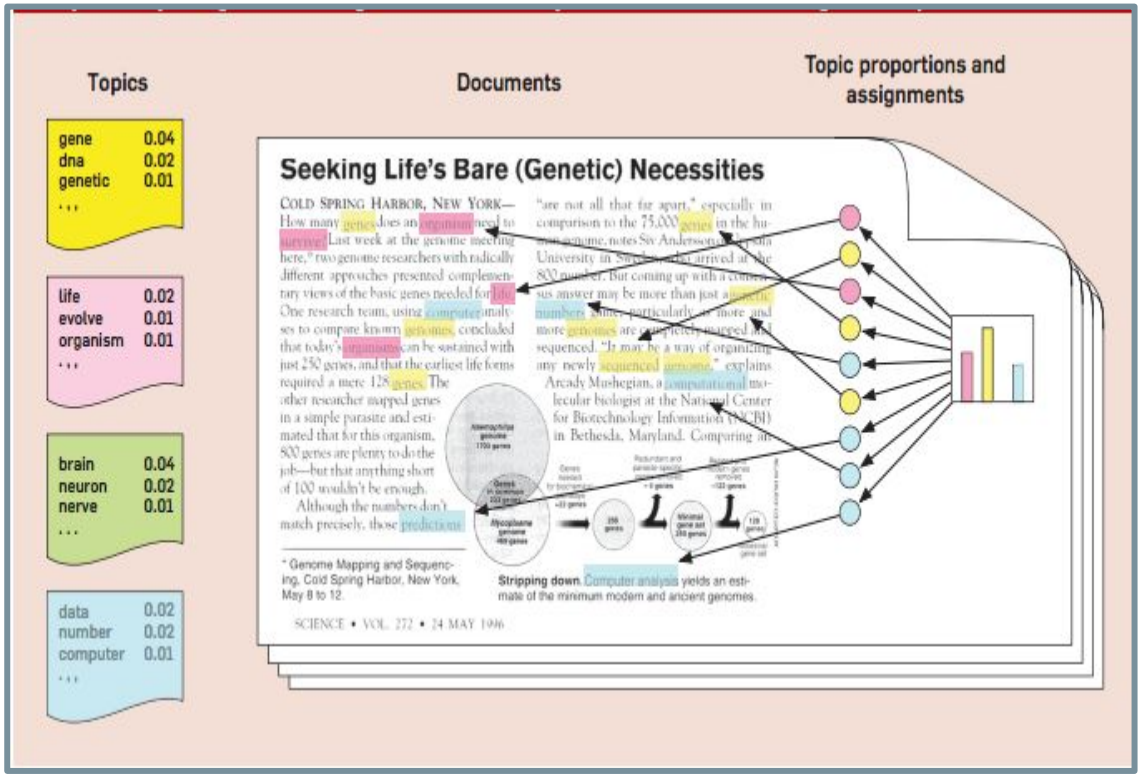
TOPIC MODELING AND IMPLEMENTATION:

Topic Modeling:

Topic models are a suite of algorithms that uncover the hidden thematic structure in document collections.

IN LAYMAN TERMS

A method of text mining to identify patterns in a corpus. Topic modeling helps us develop new ways to search, browse and summarize large archives of texts.



FIRST PHASE: Hadoop configuration (outline)

1. **INSTALL JAVA**
   1.1. sudo apt-get install sun-java-8-jdk
2. **CONFIGURE SSH**
   2.1. ssh-keygen -t rsa -P “ “
   2.2. ssh localhost
   2.3. ssh slave
3. **HADOOP INSTALL**
   3.1. Download Hadoop
   3.2. cd /usr/local
   3.3. sudo tar hadoop
   3.4. sudo chown -R hduser:hadoop hadoop
4. **ADD THE FOLLOWING PROPERTIES IN conf/core-site.xml**
   4.1. hadoop.tmp.dir
   4.2. fs.default.name
5. **ADD THE FOLLOWING PROPERTIES IN conf/mapred-site.xml**
   5.1. mapred.job.tracker
6. **ADD THE FOLLOWING PROPERTY IN hdfs-site.xml**
   6.1. dfs.replication
7. **CONFIGURE /etc/hosts**

COMMANDS TO RUN HADOOP:

/usr/local/hadoop/bin/hadoop namenode-format
/usr/local/hadoop/bin/hadoop/bin/start-all.sh
/usr/local/hadoop/bin/hadoop dfs -copyFromLocal <source> <destination>
/usr/local/hadoop/bin/hadoop/bin/stop-all.sh

Word Frequency Program : Processing

```
farheen@slave2:/usr/local/hadoop$ bin/hadoop namenode -format
Warning: $HADOOP_HOME is deprecated.

15/11/23 23:01:55 INFO namenode.NameNode: STARTUP_MSG:
/************************************************************
STARTUP_MSG: Starting NameNode
STARTUP_MSG:   host = slave2/127.0.1.1
STARTUP_MSG:   args = [-format]
STARTUP_MSG:   version = 1.2.1
STARTUP_MSG:   build = https://svn.apache.org/repos/asf/hadoop/
013
STARTUP_MSG:   java = 1.8.0_66
************************************************************/
```

Namenode Formatted

```
farheen@slave2: /usr/local/hadoop
farheen@slave2:~$ cd /usr/local/hadoop/
farheen@slave2:/usr/local/hadoop$ sudo rm -Rf /app/hadoop/tmp/*
[sudo] password for farheen:
farheen@slave2:/usr/local/hadoop$
farheen@slave2:/usr/local/hadoop$
farheen@slave2:/usr/local/hadoop$ bin/start-all.sh
```

Starting Hadoop

```
farheen@slave2: /usr/local/hadoop
farheen@slave2:/usr/local/hadoop$ jps
6736 TaskTracker
6144 NameNode
6579 JobTracker
6297 DataNode
6458 SecondaryNameNode
6891 Jps
farheen@slave2:/usr/local/hadoop$
```

All Process Running

Copy Files From Local To HDFS

```
farheen@slave2:/usr/local/hadoop$ bin/hadoop dfs -copyFromLocal /home/farheen/gut /user/farheen/inputHDFS
Warning: $HADOOP_HOME is deprecated.
```

```
farheen@slave2:/usr/local/hadoop$ bin/hadoop fs -ls /user/farheen/inputHDFS
Warning: $HADOOP_HOME is deprecated.

Found 3 items
-rw-r--r--   1 farheen supergroup    1428841 2015-11-24 07:26 /user/farheen/inputHDFS/5000-8.txt
-rw-r--r--   1 farheen supergroup     674570 2015-11-24 07:26 /user/farheen/inputHDFS/pg20417.txt
-rw-r--r--   1 farheen supergroup    1573151 2015-11-24 07:26 /user/farheen/inputHDFS/pg4300.txt
```

```
bin/hadoop jar hadoop-examples-1.2.1.jar wordcount /user/farheen/inputHDFS /user/farheen/outputHDFS
```

running jar of wordcount on data

mapReduce Complete

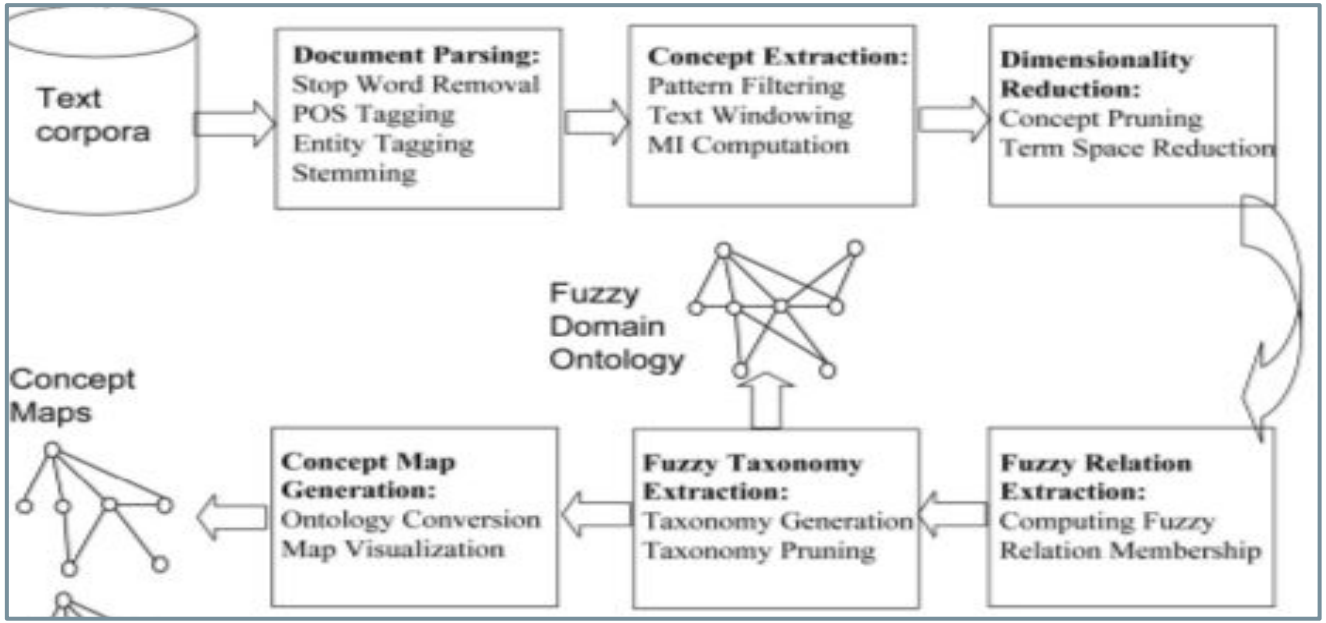
list copied files

```
input.FileInputFormat: Total input paths to process : 3
util.NativeCodeLoader: Loaded the native-hadoop library
snappy.LoadSnappy: Snappy native library not loaded
mapred.JobClient: Running job: job_201511240726_0001
mapred.JobClient:  map 0% reduce 0%
mapred.JobClient:  map 33% reduce 0%
mapred.JobClient:  map 66% reduce 0%
mapred.JobClient:  map 100% reduce 0%
mapred.JobClient:  map 100% reduce 100%
mapred.JobClient: Job complete: job_201511240726_0001
```

```
"explain"       1
"eye-hole"      1
"fan"   1
"fan,"  1
"fasciculus"    1
"feathering"    1
"feigning       1
"field,"        1
"find"  1
"finder";       1
"fire-mist"     1
"fire-mists"    1
"fire-mists,"   1
"first  1
"floating       1
"floor" 1
"flying 5
"flying"        1
"fondamento",   1
"food," 2
"fourth 1
"friar-birds"   1
"gas    1
"gliding"       1
"gone   1
"grouse 2
"half-monkeys"  1
"handiness."    1
"has    1
"he     2
"home," 1
"home." 1
"hooky" 1
"hormones,"     1
"if     1
"il     1
```

Wordcount output

SECOND PHASE:
Implementation of Fuzzy Domain Ontology Extraction Algorithm



WHY WE NEED HADOOP FOR THIS PHASE ?

ALGORITHM:

1. Standard Preprocessing
   a. Stop Word Removal
   b. Cleaning using Reg Ex
   c. Porter Stemming
   d. Tagging
   e. Noun Extraction
2. Text Windowing
3. Context vector formation
4. Mutual Information probability
5. Concept Filtering
6. Formation of Attribute-Concept Matrix
7. Fuzzy Relation Extraction
8. Concept Pruning
9. Fuzzy Taxonomy Extraction

2) For each document $d \in D$ Do
   a) Construct text windows $w \in d$
   b) Remove stop words sw from w
   c) Perform POS tagging for each term $t_i \in w$
   d) Apply Porter stemming to each term t_i
   e) Filter specific linguistic patterns such as NN, AN, NVN, etc.
   f) Accumulate the frequency for $t_i \in w$ and the joint frequency for any pair $t_i, t_j \in w$
   g) IF $lower \leq Freq(t_i) \leq upper$, THEN $A = A \cup t_i$
3) For each term $t_i \in A$ Do /* Concept Extraction */
   a) compute its context vector c_i using BMI, MI, JA, CP, KL, ECH, or NGD
   b) $C = C \cup c_i$
4) For each $c_i \in C$ Do /* Concept Filtering - α-cut */
   a) IF $\exists t_i \in c_i : \mu_{c_i}(t_i) < \zeta$, THEN $C = C - c_i$
5) For each $c_i \in C$, $t_i \in A$ Do /* Update R_{AC} relations */
   a) IF $\mu_{c_i}(t_i) \geq \zeta$, THEN $R_{AC} = R_{AC} \cup (t_i, c_i)$
6) $\forall c_i \in C$: Compute $Rel(c_i, D_j)$
7) IF $Rel(c_i, D_j) < \varpi$, THEN $C = C - c_i$ /* Concept Pruning */
8) Perform Dimensionality Reduction SVD
9) For each pair of concepts $(c_i, c_j) \in C$ Do
   a) Compute the taxonomy relation (c_i, c_j) using $Spec(c_i, c_j)$
   b) IF $\mu_{R_{CC}}(c_i, c_j) > \lambda$, THEN $R_{CC} = R_{CC} \cup (c_i, c_j)$

STEP 0: Standard Preprocessing:

```
1 going
2 gone
3 cleaning
4 lived
5 live
```

```
1 go
2 gone
3 clean
4 live
5 live
```

```
$ grep -v '^$' doc.txt > output.txt
$ cat all_words.txt | sort | uniq > unique_word.txt
$ wc -l all_words.txt unique_word.txt
```

```
89002 all_words.txt
 9560 unique_word.txt
```

```
1 the_DT
2 project_NN
3 gutenberg_NN
4 ebook_NN
5 of_IN
6 the_DT
7 notebook_NN
8 of_IN
9 leonardo_NN
```

```
while((sample = br.readLine())!=null)
{
//tag the string
//tagged = tagger.tagString(sample);

    index = sample.lastIndexOf("_NN");
    if(index !=-1)
    {
        substr = sample.substring(0,index);
        out.write(substr);
        out.newLine();
    }

}
```

```
1 project
2 gutenberg
3 ebook
4 notebook
5 leonardo
```

STEP 1: TEXT WINDOWING

```
*Using Text Window Method to find frequency of
     * words that occur together in a window*/

   for(i=0;i<len;i++)
       for(j=0;j<len;j++)
           arr[i][j]=0;

   for(i=0;i<len2;i++)
   {
       m = search(mylist2.get(i));
       if(i>5)
       {
           for(j=1;j<=5;j++)
           {
                 newstr = new String(mylist2.get(i-j));
                 n=search(newstr);
                 arr[m][n]+=1;
           }
       }
       if(i<(len-5))
       {
           for(j=1;j<=5;j++)
           {
                 newstr = new String(mylist2.get(i+j));
               n=search(newstr);

               arr[m][n]+=1;
           }
       }
   }
```

STEP 2: CONTEXT VECTOR

CONTEXT VECTORS

Concept	Context V
Computer	< (Table,5.00) (Cathode,5.00) (Bag,5.00) (Watch,5.00) (Air,5.00) (Mobile,1.00) (Mouse,1.00) (Key
Table	< (Computer,14.00) (Cathode,5.00) (Bag,5.00) (Watch,5.00) (Air,5.00) (Conditioner,1.00) (Mous
Cathode	< (Computer,14.00) (Table,14.00) (Bag,4.00) (Watch,4.00) (Air,4.00) (Conditioner,1.00) (Bottle,1
Bag	< (Computer,14.00) (Table,14.00) (Cathode,14.00) (Watch,4.00) (Air,4.00) (Conditioner,1.00) (B
Watch	< (Computer,14.00) (Table,14.00) (Cathode,14.00) (Bag,14.00) (Air,4.00) (Conditioner,1.00) (Bo
Air	< (Computer,14.00) (Table,14.00) (Cathode,14.00) (Bag,14.00) (Watch,14.00) (Conditioner,1.00
Conditioner	< (Table,5.00) (Cathode,5.00) (Bag,5.00) (Watch,5.00) (Air,5.00) (Bottle,2.00) (B
Bottle	< (Cathode,5.00) (Bag,5.00) (Watch,5.00) (Air,5.00) (Conditioner,6.00) (Box,2.00) (
Box	< (Bag,5.00) (Watch,5.00) (Air,5.00) (Conditioner,6.00) (Bottle,6.00) (Mobile,2.00) (
Mobile	< (Watch,5.00) (Air,5.00) (Conditioner,6.00) (Bottle,6.00) (Box,6.00) (Mouse,1.00) (K
Mouse	< (Table,1.00) (Cathode,1.00) (Bag,1.00) (Watch,1.00) (Air,5.00) (Cond
Keyboard	< (Cathode,1.00) (Bag,1.00) (Watch,1.00) (Air,1.00) (Conditione
Shoe	< (Bag,1.00) (Watch,1.00) (Air,1.00
Laptop	< (Watch,1.00) (Air,1.00) (Mouse,1.0
CPU	< (Air,1.00) (Mouse,1.00) (Keyboard,
Fan	< (Mouse,1.00) (Keyboard,1.00) (Sho
Bulb	< (Keyboard,1.00) (Shoe,1.00) (Lapt
Tube	< (Shoe,1.00) (Laptop,1.00) (CPU
Light	< (Computer,3.00) (Table,7.00) (Cathode,7.00) (Bag,7.00) (Watch,7.00) (Air,8.00) (Conditioner,1
Chair	< (Computer,3.00) (Table,3.00) (Cathode,7.00) (Bag,7.00) (Watch,7.00) (Air,7.00) (Conditioner,1
Card	< (Computer,3.00) (Table,3.00) (Cathode,3.00) (Bag,7.00) (Watch,7.00) (Air,7.00) (Bottle,1.00) (
Printer	< (Computer,3.00) (Table,3.00) (Cathode,3.00) (Bag,3.00) (Watch,7.00) (Air,7.00) (Box,1.00) (Mo
Pendrive	< (Computer,3.00) (Table,3.00) (Cathode,3.00) (Bag,3.00) (Watch,3.00) (Air,7.00) (Mobile,1.
Curtain	< (Bottle,1.00) (Box,1.00) (Mobile,1.0
Switch	< (Box,1.00) (Mobile,1.00) (Mouse,1.0
Socket	< (Bottle,1.00) (Box,1.00) (Mobile,1.0
Pipes	< (Box,1.00) (Mobile,1.00) (Mouse,1.0
Inverter	< (Mobile,1.00) (Mouse,1.00) (Keyboar
Door	< (Mouse,1.00) (Keyboard,1.00) (Socke
Wire	< (Keyboard,1.00) (Socket,1.00) (Pipe

MUTUAL INFORMATION:

$$MI(t_i, t_j) = \log_2 \frac{Pr(t_i, t_j)}{Pr(t_i)Pr(t_j)},$$

Attributes	Mutula Index Without Pruning
Computer	< (Curtain,0.10) (Switch,0.10) (Pen,0.10) (Jacket,0.10) (Television,0.10) (Server,0.10) (Extension,0.10) (Plug,0.10) (Tiles,0.10) (Wifi,0.10) (Window,0.10) (Stablizer,0.10) >
Table	< (Curtain,0.10) (Switch,0.10) (Jacket,0.10) (Television,0.10) (Server,0.10) (Extension,0.10) (Tiles,0.10) (Wifi,0.10) (Window,0.10) (Stablizer,0.10) >
Cathode	< (Computer,0.08) (Table,0.08) (Curtain,0.10) (Switch,0.10) (Television,0.10) (Server,0.10) (Extension,0.10) (Wifi,0.10) (Window,0.10) (Stablizer,0.10) >
Bag	< (Computer,0.08) (Table,0.08) (Cathode,0.09) (Curtain,0.10) (Switch,0.10) (Server,0.10) (Extension,0.10) (Window,0.10) (Stablizer,0.10) >
Watch	< (Computer,0.08) (Table,0.08) (Cathode,0.09) (Bag,0.09) (Switch,0.10) (Extension,0.10) (Stablizer,0.10) >
Air	< (Computer,0.08) (Table,0.08) (Cathode,0.09) (Bag,0.09) (Watch,0.09) >
Conditioner	< >
Bottle	< (Conditioner,0.22) >
Box	< (Conditioner,0.22) (Bottle,0.22) >
Mobile	< (Conditioner,0.22) (Bottle,0.22) (Box,0.22) >
Mouse	< (Air,0.09) (Conditioner,0.21) (Bottle,0.21) (Box,0.21) (Mobile,0.21) >
Keyboard	< (Conditioner,0.21) (Bottle,0.21) (Box,0.21) (Mobile,0.21) (Mouse,0.29) >
Shoe	< (Bag,0.10) (Watch,0.10) (Air,0.10) (Mouse,0.29) (Keyboard,0.29) >
Laptop	< (Watch,0.10) (Air,0.10) (Mouse,0.29) (Keyboard,0.29) (Shoe,1.76) >
CPU	< (Air,0.10) (Mouse,0.29) (Keyboard,0.29) (Shoe,1.76) (Laptop,1.76) >
Fan	< (Mouse,0.29) (Keyboard,0.29) (Shoe,1.76) (Laptop,1.76) (CPU,1.76) >
Bulb	< (Keyboard,0.29) (Shoe,1.76) (Laptop,1.76) (CPU,1.76) (Fan,1.76) >
Tube	< (Shoe,1.76) (Laptop,1.76) (CPU,1.76) (Fan,1.76) (Bulb,1.76) >
Light	< (Laptop,0.14) (CPU,0.14) (Fan,0.14) (Bulb,0.14) (Tube,0.14) >
Chair	< (CPU,0.14) (Fan,0.14) (Bulb,0.14) (Tube,0.14) (Light,0.10) >
Card	< (Fan,0.14) (Bulb,0.14) (Tube,0.14) (Light,0.10) (Chair,0.10) >
Printer	< (Bulb,0.14) (Tube,0.14) (Light,0.10) (Chair,0.10) (Card,0.10) >
Pendrive	< (Tube,0.14) (Light,0.10) (Chair,0.10) (Card,0.10) (Printer,0.10) >
Curtain	< (Bottle,0.25) (Box,0.25) (Mobile,0.25) (Mouse,0.29) (Keyboard,0.29) >
Switch	< (Box,0.25) (Mobile,0.25) (Mouse,0.29) (Keyboard,0.29) (Curtain,1.76) >
Socket	< (Bottle,0.25) (Box,0.25) (Mobile,0.25) (Mouse,0.29) (Keyboard,0.29) >
Pipes	< (Box,0.25) (Mobile,0.25) (Mouse,0.29) (Keyboard,0.29) (Socket,1.76) >
Inverter	< (Mobile,0.25) (Mouse,0.29) (Keyboard,0.29) (Socket,1.76) (Pipes,1.76) >
Door	< (Mouse,0.29) (Keyboard,0.29) (Socket,1.76) (Pipes,1.76) (Inverter,1.76) >
Wire	< (Keyboard,0.29) (Socket,1.76) (Pipes,1.76) (Inverter,1.76) (Door,1.76) >
Calender	< (Socket,1.76) (Pipes,1.76) (Inverter,1.76) (Door,1.76) (Wire,1.76) >
Dustbin	< (Pipes,1.76) (Inverter,1.76) (Door,1.76) (Wire,1.76) (Calender,1.76) >
Pen	< (Inverter,1.76) (Door,1.76) (Wire,1.76) (Calender,1.76) (Dustbin,1.76) >

$$\mu_{R_{CC}}(c_x, c_y) \approx Spec(c_x, c_y) = \begin{cases} 0 & \text{if } SSIM(c_y, c_g) > SSIM(c_x, c_g) \\ \frac{SSIM(c_x,c_g) - SSIM(c_y,c_g)}{SSIM(c_x,c_g)} & \text{otherwise.} \end{cases}$$

$$SSIM(c_x, c_y) = l(c_x, c_y) \cdot c(c_x, c_y) \cdot s(c_x, c_y),$$

$$l(c_x, c_y) = \frac{2M_{c_x}M_{c_y} + Q_1}{M_{c_x}^2 + M_{c_y}^2 + Q_1},$$

$$c(c_x, c_y) = \frac{2\sigma_{c_x}\sigma_{c_y} + Q_2}{\sigma_{c_x}^2 + \sigma_{c_y}^2 + Q_2},$$

$$s(c_x, c_y) = \frac{\sigma_{c_x,c_y} + Q_3}{\sigma_{c_x}\sigma_{c_y} + Q_3},$$

```java
public static void concept_pruning()
{
    int c=0;
    float p;

    for(int j=0;j<len;j++)
    {
        for(int i=0;i<len;i++)
            if(prob[j][i]>0)
                c++;
        p = (float)((float)c/(float)len);

        if(p>0.14){
            concept_count[j]=1;
            System.out.println(j+" "+p);
        }
        c=0;
    }

}
public static void Rac_calculation()
{
    for(j=0;j<len;j++)
    {
        if(concept_count[j]>0)
        {
            for(i=0;i<len;i++)
                Rac[i][j]=prob[i][j];

        }
    }
}
```

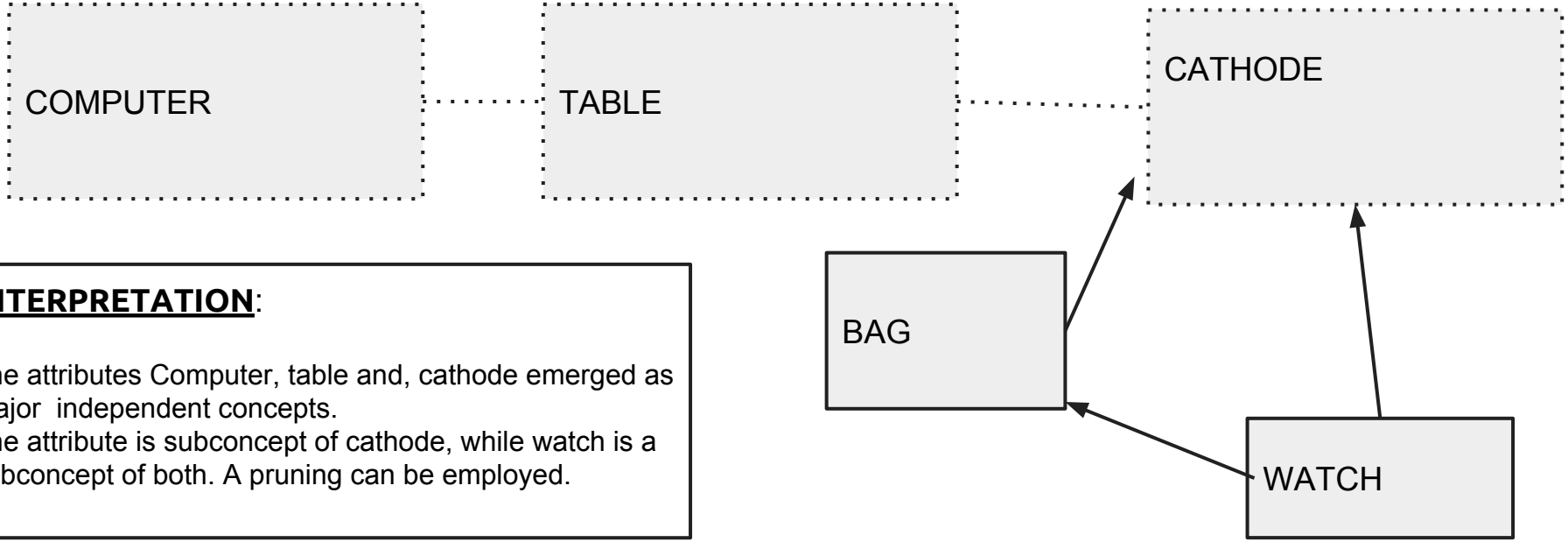
```
The prunned concepts::
Computer
Table
Cathode
Bag
Watch

Attribute-Concept
Computer 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0
Table 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
Cathode 0.08052287995815277 0.08052287995815277 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.
Bag 0.08052287995815277 0.08052287995815277 0.08525951951742172 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.
Watch 0.08052287995815277 0.08052287995815277 0.08525951951742172 0.08525951951742172 0.0 0.0

Common Concept calculation
Computer NaN NaN NaN NaN NaN 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0
Table NaN NaN NaN NaN NaN 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
Cathode NaN NaN 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.
Bag NaN NaN 5.740759052675054E-5 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0
Watch NaN NaN 1.146974312105059E-4 5.6447591154446693E-5 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0
```

RESULT:

Concept	Concept
Computer	< >
Table	< >
Cathode	< >
Bag	< (Cathode,0.00) >
Watch	< (Cathode,0.00) (Bag,0.00) >



INTERPRETATION:

The attributes Computer, table and, cathode emerged as major independent concepts.
The attribute is subconcept of cathode, while watch is a subconcept of both. A pruning can be employed.

WORK ACCOMPLISHED:

- Configuration of hadoop on single system.
- Configuration of hadoop on multi cluster system (3 systems currently).
- Development and implementation of Fuzzy ontology extraction algorithm.
- Graphical display of concept vectors.

WORK TO BE DONE:

- Complete and convert the algorithm in Map-Reduce form.
- Application of algorithm on big data set.
- Construction of Concept graph.

REFERENCES:

1. Toward a Fuzzy Domain Ontology Extraction Method for Adaptive e-Learning
2. Probabilistic Topic Models for Learning Terminological Ontologies
3. The Role of Domain Knowledge in a Large Scale Data Mining Project
4. A Short Fuzzy Logic Tutorial

ACKNOWLEDGEMENTS:

1. Dr Tanvir Ahmad
2. Rafeeq Ahmed
3. Michel Noll
4. Sujee Maniyam
5. Petri Kainulainen

THANK YOU