* Data Source

Checkpoint is one of the premier Tax and Accounting products from Thomson Reuters. The top 100 U.S. CPA firms rely on Checkpoint as their information resource for tax, audit and accounting, international trade, benefits and finance. (<https://tax.thomsonreuters.com/products/brands/checkpoint/advantages/>)

I have joined the web development team last summer (2015). The team has been logging user search logs since 2010. The logs have been in a NAS drive and is not really being used for anything other than tracing errors. Our searches in Checkpoint is passed on an enterprise search platform so it makes sense logging user searches in order to trace errors if the search from the enterprise search platform fails. The user search that I am going to analyze for this project has a date range from 2010-09-01 to 2016-03-06. The zipped logs are about 10 GB.

* data description and schema

The log data that I have uploaded in the cluster is a tab delimited text file that is variable in length.

The schema of the log file is as follows

fileCount+”\_”+sessionId+ “\t” +SearchKeyword+”\t”+practice+”\t”+userType+”\t”+searchType+”\t”+tab location+”\t”+version+”\t”+search results

10\_01BCEA5099D956DCE55F349110EEBF72 1.1031(k) 1 2 TC 60 v1 366

Sample log file entries:

10\_01BCEA5099D956DCE55F349110EEBF72 1.1031(k) 1 2 TC 60 v1 366

10\_01BCEA5099D956DCE55F349110EEBF72 1.1031(k)-1(g)(6)\* 1 2 TC 60 v1 65

10\_01BCEA5099D956DCE55F349110EEBF72 1.1031(k)-1(g)(6)\* & "blanket lien" 1 2 TC 60 v1 1

10\_01BCEA5099D956DCE55F349110EEBF72 1.1031(k)-1(g)(6)\* & lien or "security interest" 1 2 TC 60 v1 14

10\_01BCEA5099D956DCE55F349110EEBF72 1.1031(k)-1(g)(6)\* & pledge\* 1 2 TC 60 v1 59

10\_01BCEA5099D956DCE55F349110EEBF72 1.1031(k)-1(g)(6)\* & pledged 1 2 TC 60 v1 3

10\_082EA801EB377D2C7D7462031124E911 unreimbursed employee expense deduction 99 2 cobalt 1050 v1 1800

10\_0FC604003EFB7BFC6C028E56BB6A7580 950 1 9 cobalt 60 v1 8

10\_0FC604003EFB7BFC6C028E56BB6A7580 gift tax 1 9 cobalt 60 v1 95

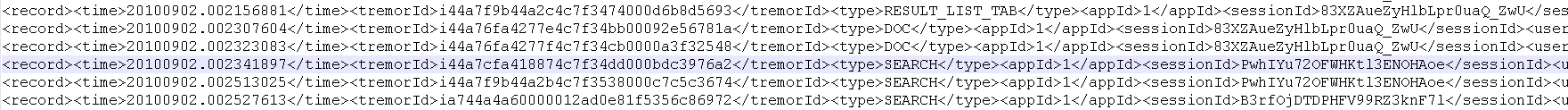
10\_111F2556FACC14BF924072D913B64B75 6013 1 2 cobalt 60 v1 2001

10\_1400EFDD444AE0A7E40549E9E87C0CC7 mortgage insurance premiums 1 2 cobalt 60 v1 2000

* data pre-processing required (parsing, filtering)

The original log is actually in xml format being logged with other user events like documents viewed. Tab location that were clicked for each user session.

Here is a sample of the original log files.



Here is a sample user search log (substituted X for some fields – identifiable data)

from this.

<record><time>20100902.002341897</time><tremorId>XXXXXXXXXX</tremorId><type>SEARCH</type><appId>1</appId><sessionId>PwhIYu72OFWHKtl3ENOHAoe</sessionId><userId>XXXXXX</userId><userType>9</userType><searchResultId>XXXXXXXXXX</searchResultId><search><searchType>TC</searchType><keywords>e.g.-4A(b)(2)(ii)(C)</keywords><resultCount>0</resultCount><practiceArea>1</practiceArea><tabLoc>60</tabLoc><collections><collectionId>CODE</collectionId><collectionId>CODEHIST</collectionId><collectionId>FTREGS</collectionId><collectionId>ADVREGS</collectionId><collectionId>TD</collectionId><collectionId>NOTICE</collectionId><collectionId>REPREGS</collectionId><collectionId>REPPREGS</collectionId><collectionId>COMREP</collectionId></collections></search></record>

The uploaded log file in the cluster was parsed from these logs.

* any bad data issues

TopKeywords MapReduce

The search keywords come in 2 formats: Natural language format and Terms and Connector format. I have to account for both when counting and sorting the keywords. There are also keywords that have multiple double quotes and multiple trailing and leading spaces inside or outside the multiple double quotes.

Analysis Map Reduce

Integer division is giving me an error ArithmeticException: / by zero. Changed this to a double precision.

Double precision has to me to minimum of 4 decimal places for the analysis otherwise it will not be able to sort the analysis result properly.

* your MR algorithm
* Know the Top 10 search keywords for the last 5-6 years
* What do Tax professional using Checkpoint search the most during Tax Season (Mar - Apr)? Top 10 Search keywords during Tax Season (Mar - Apr)
* Comparative Analysis of Searches in the month of Mar-Apr against searches for the whole year.
  + For the comparative analysis (<https://en.wikipedia.org/wiki/Chi-squared_test>)

I decided that I would apply a Chi-Squared to quantify how much users prefer the Mar-Apr keywords over all the keywords being searched in Checkpoint

To Apply the Chi-Squared technique I will use this formula:

Preference = kT + kTotal

Where:

kT = (Count of Search Keyword Tax Season – Total Count for Keyword)^2 / Total Count for Keyword

This is the Chi-square value of the Keyword that we found during tax season compared to the Total Count for the Keyword

kTotal = (Total Count for Keyword – Total Count of All Keywords)^2 / Total Count of All Keywords

This is the Chi-square value of the Total Count of Keyword compared the total number of all keywords

Preference value of closest to 1.0 means that Search is preferred during Tax Season

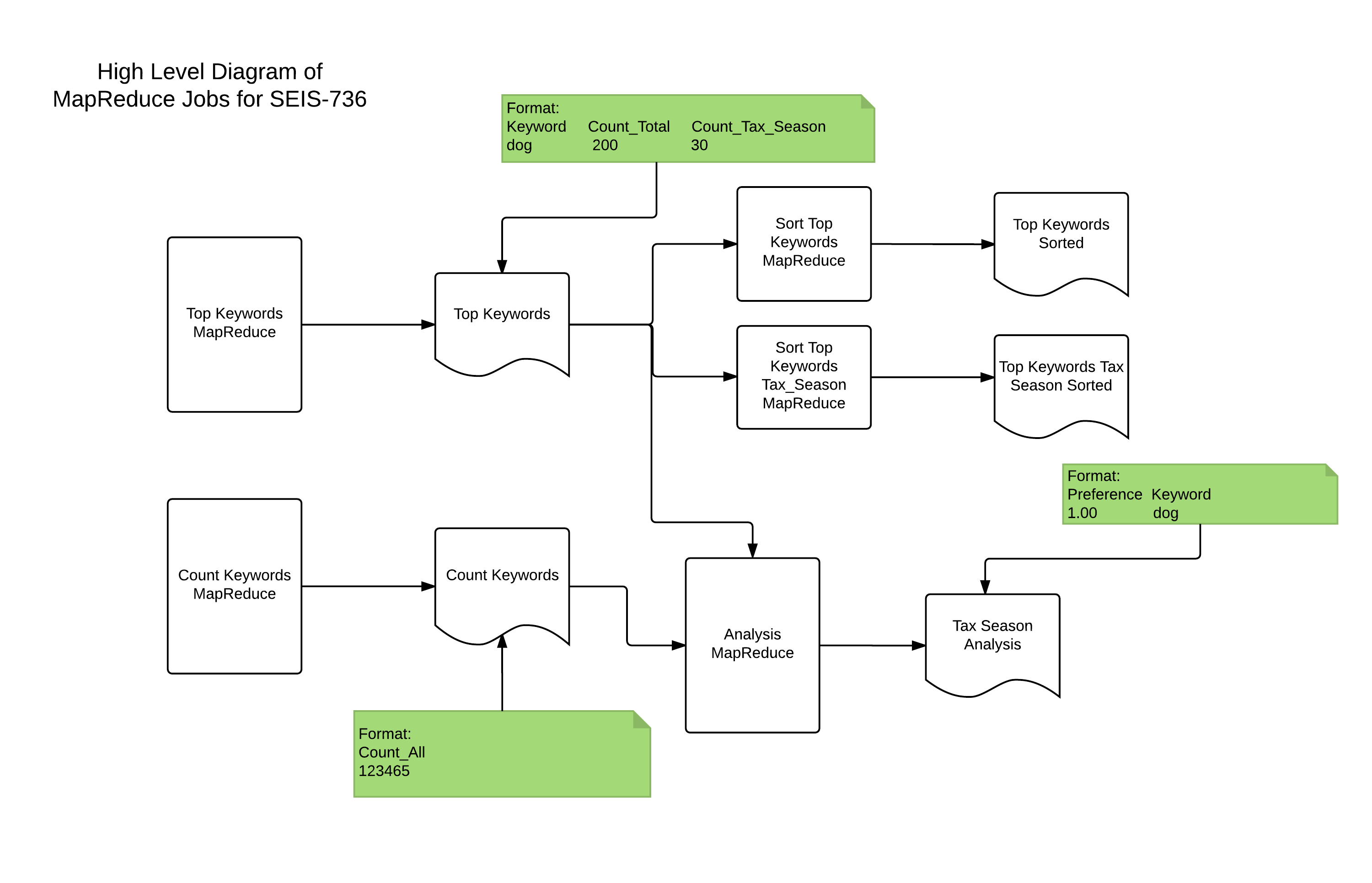


Figure Map Reduce Job Dependencies

* description of any other ecosystem or additional tools

After the MapReduce jobs I have imported the data in R to visualize (as you will see in the graph in the conclusions section). This is good for presenting the results rather than just having a tab delimited file.

Used standalone java to parse and clean the log files from the original format for me to be able to upload to the cluster.

Shell script using the hdfs to load the log files.

Used DistributedCache to read the Total Count of Keywords when doing the Comparative Analysis.

* output description

As you can see in Figure 1 Map Reduce Job Dependencies. The whole process has 5 output.

1.Top Keywords – This contains the total count for each keyword for all time and counts during tax season (Mar - Apr)

2. Count Keywords – This contains the total count of all the keywords parsed. This will be used as a Distributed Cache File in the Analysis Map Reduce Job.

3.Top Keywords Sorted – This is a sorted file in descending count sort order for Keywords for all time.

4. Top Keywords Tax Season Sorted – This is a sorted file in descending count sort order for Keywords during Tax Season.

5. Tax Season Analysis – This is the analysis output where the Chi-squared test technique is applied for keywords between counts during Tax Season vs counts for All Time. The file is sorted in descending chi value order.

* how did you verify that your output is correct?

Created a small test file with expected values where I did an initial run of all the MapReduce Job. This is under testLogFile under my user name in hdfs.

Wrote MRUnit Test for all map reduce jobs with expected input and output values.

* performance/scale characteristics
* what would you have done differently if you did this again?
* Conclusions

