Un-moderated real-time news trends extraction from World Wide Web using Apache Mahout

A Project Report Presented to

Professor Rakesh Ranjan

San Jose State University

Spring 2011

By
Kalaivanan Durairaj (006993792)
Kalaivanan.Durairaj@gmail.com
CMPE 272

Abstract

Un-moderated real-time news trends extraction from World Wide Web using Apache Mahout

Over the past 2 decades we built the World Wide Web with tremendous amount of information that no other system in the world can match. An estimate says that approximately 14.66 billion pages and still growing day by day. In addition to the web pages we have ever growing newer sources of data such as feeds, blogs, tweets etc. If we could find a reliable way to understand process and extract intelligence out of it, we can do wonders by improving the life of human being in multiple ways. This project is to explore one such area to research the possibility of building un-moderated news and trends webpage by applying Symantec web concepts with Apache Mahout Machine learning framework.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS

I would like to take this opportunity to sincerely thank Professor Rakesh Ranjan for his invaluable guidance throughout this semester and for introducing us to lot of exciting cutting-edge trends in enterprise software engineering such as big data, distributed database, appliance model etc.

Kalaivanan Durairaj

1. Introduction

The data available in World Wide Web is massive and exploding day by day. Google is estimated to index over 15 billion web pages and that's just the tip of iceberg. In addition to web pages, the Internet has many other conventional and news sources that constantly inject with data. Social media sites and personal blogs expedited this expansion. If we can find smart ways to extract knowledge and intelligence out of it we can create many fascinating applications.

The problem with this idea is that it is not easy and simple. The following properties of this web data make this notion a very tough challenge:

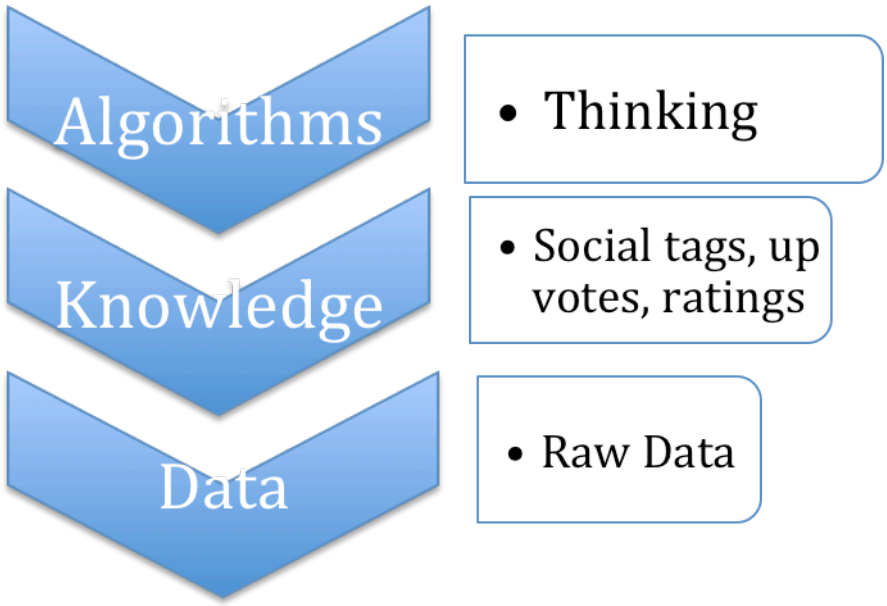
- Unstructured and not originally intended for machine processing
- Massively large
- Available many different formats and languages
- Distributed across the geography
- Mixed with lot of noise / bad data
- Processing requires huge amount of computing power
- No dictionary or classification

Thanks to the developments in distributed processing frameworks such as Map Reduce and scalable open source implementations of various analysis, clustering and classification algorithms, it possible now to extract the knowledge and intelligence out of this.

The intent of this paper is not just limited to news trends extraction. News trends extraction is on metaphor of the possibilities we are trying to explore.

2. Components of an Intelligent Software

All of the modern intelligent analytical software system used by many of the successful Internet companies and startups such as Google, digg.com etc has three basic components in common.

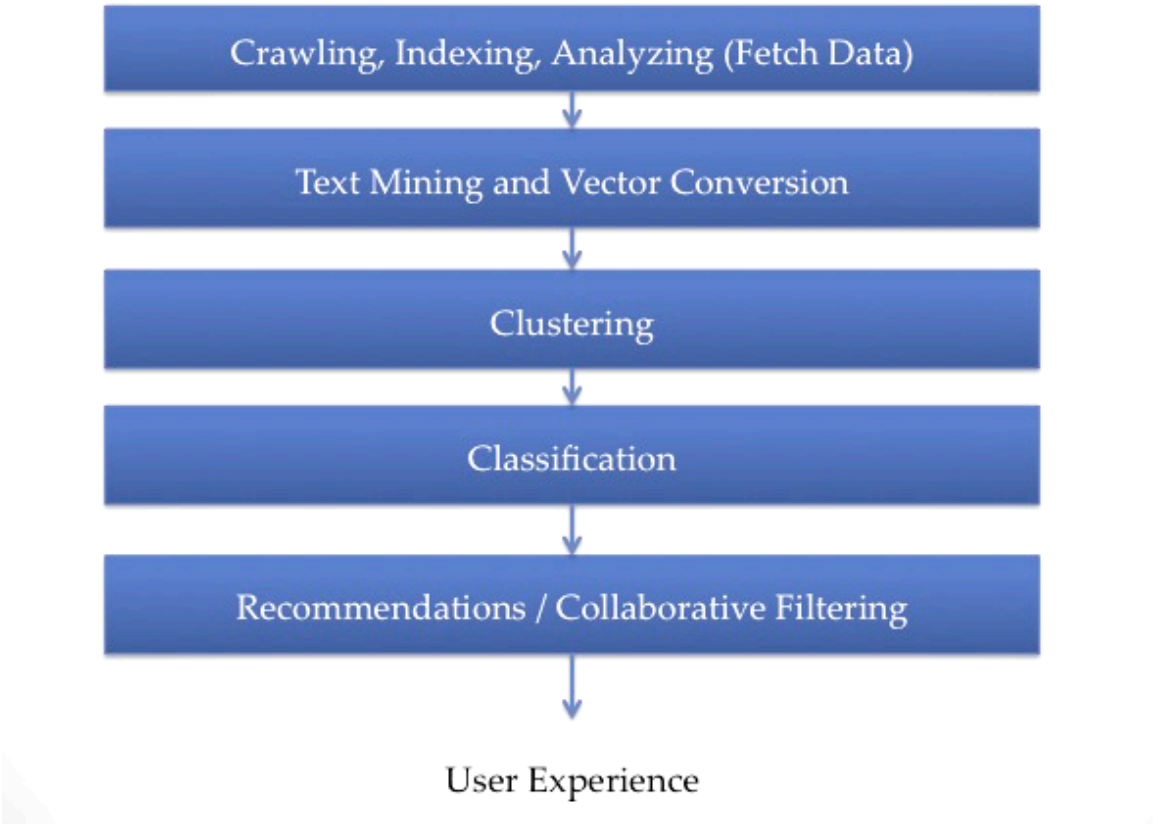


The data part is the unstructured, geographically dispersed pieces of information with links and associations. Knowledge is the semantic information, which helps in interpretation of the content. They are either already existing or user generated information such as knowledge bases, folksonomy, social tags, collaborative tagging, social indexing, social classification etc.

The most interesting part of this triangle is the algorithms that help software think, learn and make intelligent decision on the available data and content.

In next section we will first see how a real-time news extraction system can be built by further breaking down these components. Then we will see the algorithms that make it possible. And then in last section we will see how we can build a scalable implementation of this system by employing the open source software stack including Apache Mahout, Apache Hadoop, Apache Nutch and Apache Lucene.

3. Architecture of an automated news trends extraction system



First step of our process is to extract the data from web. Crawling is the process of navigating thru the web pages in the public Internet domain, read and extract the desired content from it. Crawler can be configured to follow a list of domains at certain level of depth and certain number of pages.

The introduction of page-rank algorithm by Google and user click analysis has a major impact in this area. Instead of just doing a crude extraction of content, link-analysis helps extracting the content with high reputation eliminating all the noise. User click analysis adds user specific behavior knowledge to this process.

Indexing of the extracted content helps searching the massive content in blazing fast. Indexed content are then analyzed and processed to extract the useful data out of this. This is not limited to text, but for our purpose this means doing text mining [3].

Text mining field have many fascinating inventions in past decade. Companies like IBM research invested a lot in this field making leaps and bounds of progress in natural language processing techniques and various related text-mining algorithms. This helps in extracting the important content not just based on words, but also semantic meaning and references.

Once we have the processed clean text content, we need a way to apply the classification and clustering algorithms to it. But these algorithms are mathematical and cannot be applied directly to text content. The process that helps in transforming this text content to vector (n-dimensional points in space) is called vectorization. Vector Space Model (VSM) is a model that helps representing text data in vector form. We will see more details on this in next section.

Classification is a class of machine learning algorithms that help in classifying the input data under n number of topics based on multiple attributes. Real life objects such as text document and people profile are complex and require sophisticated algorithms to carryout classification automatically.

Clustering algorithms group the classified articles based on its content. This helps clustering articles based on topics.

These classification and clustering algorithms are smart enough to work on fuzzy data which helps in tasks like news trend extraction, since there is no defined boundary or domain for the input data.

We can also have a recommendations engine, which have some knowledge on user behavior and does a collaborative filtering on content to present articles that a user specifically interested in.

4. Algorithms for News trend extraction

As we saw in previous section, we have to employ a list of intelligent algorithms for our processing. The following sections present a overview of various algorithms available under each classification. The selection of the right algorithm is crucial since our problem domain has a fuzzy data. Fortunately many mathematicians and scientists blessed us with this knowledge. Many of these algorithms are in-fact devised many decades back and many of them borrowed from algebra.

4.1 Searching, Indexing and Analysis

Indexing helps in fast and accurate retrieval of information from large data set. Many search engines employ inverted index instead of forward indexing. This helps in searching documents based on keywords. State of art searching by companies like Google goes beyond indexing to link analysis, user click analysis and natural language processing.

Link Analysis Algorithm (PageRank)

PageRank algorithm, invented by Google, computes reputation of a page based on the number of links referenced to it. This concept helps in eliminating noise and fetch content that has high quality and usefulness.

User Click Analysis

User click analysis is a behavioral algorithm that learns a user interest by collecting data on which search result user clicked.

Search Index is a crude technique for indexing which may not know how to separate spam from genuine content. PageRank/Link analysis adds weight and orders search result in an intelligent way. User click analysis further improves the search result by taking user behavior or user interest into account.

4.2 Recommendations and Collaborative Filtering

The text data obtained by Crawling and Analyzing need to be converted to a format where we can apply the mathematical algorithms to produce recommendations, clustering and classification.

These text data cannot be directly fed to the mathematical algorithms, which operate on vectors. Fortunately, we already have a well-designed technique to represent the text data and text queries in mathematical format.

VSM (Vector Space Model) helps representing text document in vector format as shown below:

$$d_j = (w_1 j, w_2 j, \ldots, wtj)$$

d is the vector representation of the document with weights for each word in document set.

An algorithm called term frequency – inverse document frequency is widely used for calculating the weight of each word [5]. This algorithm helps out all the common high-occurring simple stop words from important words.

4.3 Recommendations and Collaborative Filtering

Recommendation and collaborative filtering is getting wide popularity in recent days. Many large corporates and successful startups such as Amazon, Netflix improve user experience by a surprising scale by employing these techniques.

User feedback and preferences are stored in database and there are analyzed to find matches in taste or profile of users. Using this information, recommendations are make with more accuracy thereby increasing the user experience and business revenue.

The important process in this step is finding the similarity or distance between different user / objects or any abstract entries.

Euclidian Distance and Jaccard Index are two very widely used algorithms to find the distance between two different vectors.

For example, Euclidian Distance, finds the distance between two vectors by finding applying the following formula.

$$d(p, q) = sqrt(sqr(q1-p1) + sqr(q2-p2) ... sqr(qn-pn))$$

p and q are two different vectors while d is the distance between them. This formula is used to form a similarity matrix, which helps in recommending user with news articles he will be interested in.

Another popular algorithm in this class that has application in fuzzy domain is Bayesian Theory. This algorithm is different from the earlier algorithms since instead of doing a metric or algebraic distance between two vector, this takes into account multiple probability factors.

4.4 Clustering

Clustering is the process of grouping content based on the fuzzy information such as words or word phrases in a set of documents or various attributes of a user profile. Clustering in a key component in machine language algorithms since its natural to cluster information and retrieve inferences out of it in any fundamental information process.

K-means[1] and ROCK[2] (Robust Algorithm for clustering categorical attributes) are two very popular algorithms for clustering.

This K-means algorithm partitions n number of objects into k number of clusters.

1. Randomly choose k number of initial centroids from n objects
2. Assign each object to its nearest centroid to form k clusters

3. Recalculate centroids of all k clusters
4. Repeat step 2 and 3 until there is no more need for movement

K-means is well suited for problems like our news trend extraction with large and fuzzy data [4]

4.5 Classification

Classification algorithms help classify the vectors / points (in our case articles) based on a training data into appropriate topics. Classification algorithms are central part of predictive analytics. The goal of predictive analysis is to build software systems that can predict human behavior in a given situation and make decisions similar to that. Classification is a key step in that.

One example of classification is email spam filter. The input data is not bound to any limits. The problem domain is huge that it's not very feasible to build a simple rule engine for it. The classification algorithm solves this getting trained based a set of inputs and then able to classify any further input based on that knowledge.

Classification algorithms are supervised learning algorithms while clustering is unsupervised learning algorithms.

Classification algorithms in general have these 3 steps in their algorithms

1. Training
2. Evaluating
3. Production

5. Apache Mahout

What is Mahout?

"Apache Mahout is a scalable machine learning library that supports large data sets" – mahout.apache.org

For building a system like our news portal, many of the algorithms we discussed in earlier chapter need to be implemented. This is not a small task.

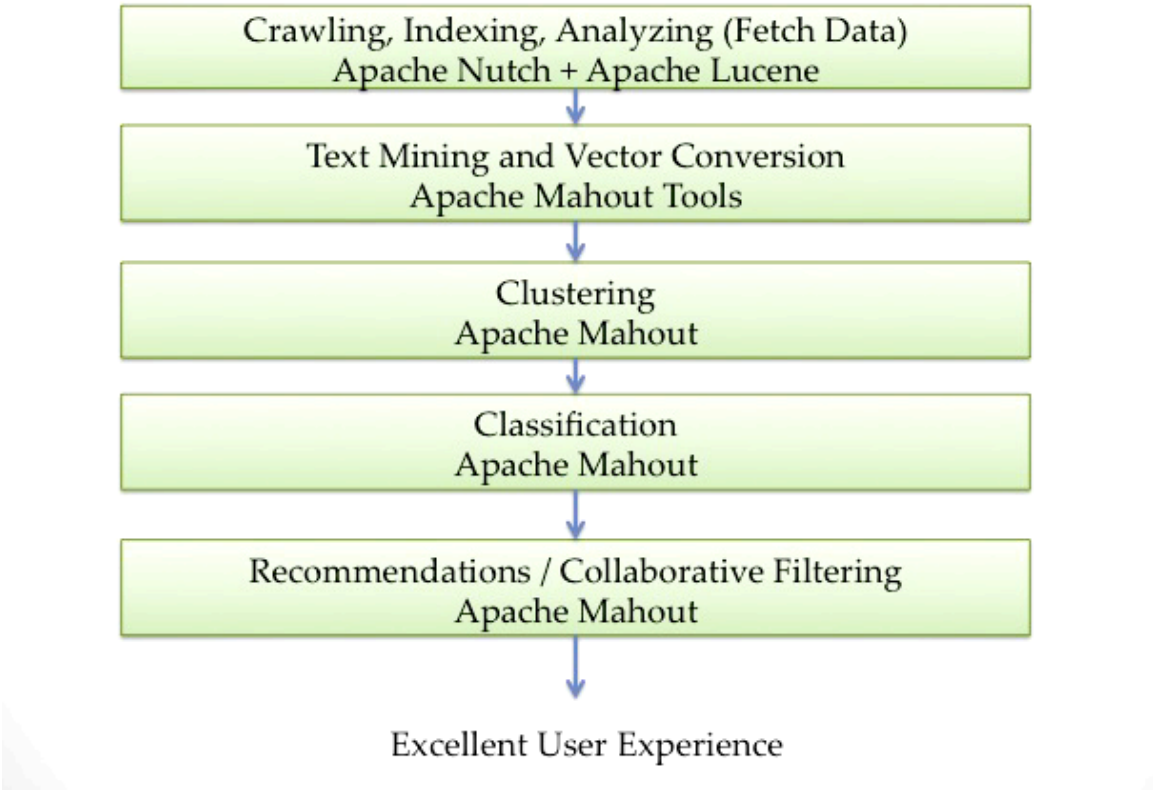
Not only these algorithms are complex in nature, but also they need be implemented in a scalable way so that we can apply them for large problem domains such as ours.

Fortunately, we don't need to start from the scratch. Apache Mahout is a collection of APIs that implement all of the above-discussed classes of algorithms in an efficient manner. Not just that, these algorithms are implemented as Apache Hadoop map-reduce jobs with massive scalability. That makes our design goal of crawling thousands of URLs, feeds, news media sites and process all the massive data possible.

Here is a list of algorithms that mahout implement as of now (and growing):

- Collaborative Filtering
- User and Item based recommenders
- K-Means, Fuzzy K-Means clustering
- Mean Shift clustering
- Dirichlet process clustering
- Latent Dirichlet Allocation
- Singular value decomposition
- Parallel Frequent Pattern mining
- Complementary Naive Bayes classifier
- Random forest decision tree based classifier
- High performance java collections (previously colt collections)

6. Scalable News trend Extraction System with Open-Source Software



With the open source software and frameworks that are available today from Apache software foundation, we revisit the architecture we presented in earlier section and show how open source makes it feasible. Almost each and every component of the software components we had in our original design can be replaced with open source implementations from Apache.

Apache Nutch is a high-performance scalable open source crawler that can crawl index, process and analyze data. Nutch uses Lucene for indexing. Lucene is an open source pure java indexing implementation. Apache Nutch also runs on Hadoop to make it scalable.

As discussed in previous section, Apache Mahout can be utilized for all the different algorithms we discussed including recommendations engine, clustering and classification.

7. Summary

In this paper we have taken a complex but interesting problem that has large fuzzy data set and huge computation requirement and tried to design a system that can solve this problem. The fascinating part of this is that we don't have to sweat on implementing most part of it. Open source software provides all of these for free of cost. When this combined with potential of cloud computing, we could easily design such system and bring it to production with relatively minimal effort and low startup costs. Only our imagination is our limit.

Reference

[1] K-means Clustering in the Cloud -- A Mahout Test
Esteves, Rui Maximo; Pais, Rui; Rong, Chunming Advanced Information Networking and Applications (WAINA), 2011 IEEE Workshops of International Conference on | 2011

[2] ROCK: a robust clustering algorithm for categorical attributes
Guha, S.; Rastogi, R.; Shim, K.
Data Engineering, 1999. Proceedings., 15th International Conference on 1999

[3] News clustering system based on text mining
Ji-Rui Li; Kai Yang
Advanced Management Science (ICAMS), 2010 IEEE International Conference on 2010

[4] Application of K-means Clustering Algorithms in News Comments
Hongwei Xie; Li Zhang; Jingyu Sun; Xueli Yu
E-Business and E-Government (ICEE), 2010 International Conference on 2010

[5] Improvement of Text Feature Selection Method Based on TFIDF
Shouning Qu; Sujuan Wang; Yan Zou
Future Information Technology and Management Engineering, 2008. FITME '08. International Seminar on 2008