****

**A thesis paper submitted to the Department of Computer Science and Engineering, Shah Jalal University of Science & Technology, in partial fulfillment of the requirement for the degree of Bachelor of Science (Engineering).**

**Supervisor**   **Thesis Committee**

Md. Saiful Islam Prof. Dr. Shahidur Rahman

Lecturer, Dept. of CSE Head, Dept. of CSE

Shahjalal University of Science & Shahjalal University of Science &

Technology, Sylhet-3114 Technology, Sylhet-3114.

**20th April 2014**

**Thesis on Information Retrieval:**

**Location Based Crime Ranking Retrieving News from Different Newspaper**

****

Department of Computer Science and Engineering

Shahjalal University of Science and Technology

Sylhet, Bangladesh

**CSE 404 Report**

**Submitted By \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Rajib Chandra Das (2009331008)**

**Md. Ruhul Amin (2009331011)**

**Md. Jumman Hossain (2009331054)**

**Acknowledgement:**

We are grateful to our supervisor “Md. Saiful Islam” for giving us cordial support and direction to our working progress. We also thank to the author and websites from which we get so much knowledge and support for our thesis.

**Abstract:**

**Table of Contents**

1. Introduction
2. **Introduction**

Constantly, there have lots of crime happening all around. Most crime is not reported to the police so there is lot of room for error. Law enforcement agencies can affect the amount of crime reported through aggressive interactions with citizens. Crime statistics are confusing and frequently misunderstood. There are criminologists who spend their professional lives investigating the complexity of crime data. National Institute of Justice release crime survey data for the country based on reported and unreported crime and does not offer crime statistics for states, metro areas or cities.

Most crime rankings are based on crimes per 1,000 residents which immediately creates an unfair playing field if you get thousands of tourists or workers per day. Those thousands of “outsiders” will inevitably commit crimes or inadvertently create opportunities for crime that would not exist in cities or states not getting a lot of tourists or daily workers.

So the bottom line is that crimes and crimes reported can and will differ for reasons having little or nothing to do with the quality of policing or crime control strategies.

Considering all of this cases we have come up with a solution which can give an approximation to people about the safety of a specific location with crime ranking if different areas. Our solution collects crime information from different online newspapers and point them on a local map with ranking.

1. **Web Crawler**

A web crawler is a program that, given one or more seed URLs, downloads the web pages associated with these URLs, extracts any hyperlinks contained in them, and recursively continues to download the web pages identified by these hyperlinks. Web crawlers are an important component of web search engines, where they are used to collect the corpus of web pages indexed by the search engine. Moreover, they are used in many other applications that process large numbers of web pages, such as web data mining, comparison shopping engines, and so on. Despite their conceptual simplicity, implementing high-performance web crawlers poses major engineering challenges due to the scale of the web. In order to crawl a substantial fraction of the “surface web” in a reasonable amount of time, web crawlers must download thousands of pages per second, and are typically distributed over tens or hundreds of computers. Their two main data structures – the “frontier” set of yet-to-be-crawled URLs and the set of discovered URLs – typically do not fit into main memory, so efficient disk-based representations need to be used. Finally, the need to be “polite” to content providers and not to overload any particular web server, and a desire to prioritize the crawl towards high-quality pages and to maintain corpus freshness impose additional engineering challenges.

* 1. **How does a web crawler work**

When a search engine's web crawler visits a web page, it "reads" the visible text, the hyperlinks, and the content of the various tags used in the site, such as keyword rich Meta tags. Using the information gathered from the crawler, a search engine will then determine what the site is about and index the information. The website is then included in the search engine's database and its page ranking process.

Web crawlers may operate one time only, say for a particular one-time project. If its purpose is for something long-term, as is the case with search engines, web crawlers may be programed to comb through the Internet periodically to determine whether there has been any significant changes. If a site is experiencing heavy traffic or technical difficulties, the spider may be programmed to note that and revisit the site again, hopefully after the technical issues have subsided.

Initialize frontier with

seed URLs

Check for termination

[

]

done

]

not done

[

Pick URL

from frontier

[

no URL

]

]

[

URL

Fetch page

Parse page

Add URLs

to frontier

end

start

Crawling Loop

Fig. 1. Flow of a basic sequential crawler

* 1. **Parsing**

**2.3 Stoplisting and Stemming**

When parsing a Web page to extract content information or in order to score new URLs suggested by the page, it is often helpful to remove commonly used words or stopwords such as “it” and “can”. This process of removing stopwords from text is called stoplisting. The stemming process normalizes words by conflating a number of morphologically similar words to a single root form or stem. For example, “connect,” “connected,” and “connection” are all reduced to “connect.” Implementations of the commonly used Porter stemming algorithm are easily available in many programming languages. One of the authors has experienced cases in the biomedical domain where stemming reduced the precision of the crawling results.

* 1. **Stemming**
  2. **Keyword Extraction**
  3. **Cosine Similarity**

The cosine similarity between two vectors (or two documents on the Vector Space) is a measure that calculates the cosine of the angle between them. This metric is a measurement of orientation and not magnitude, it can be seen as a comparison between documents on a normalized space because we’re not taking into the consideration only the magnitude of each word count (tf-idf) of each document, but the angle between the documents. What we have to do to build the cosine similarity equation is to solve the equation of the dot product for the:

And that is it, this is the cosine similarity formula. Cosine Similarity will generate a metric that says how related are two documents by looking at the angle instead of magnitude.

**2.6 TF-IDF**

TF–IDF, short for term frequency–inverse document frequency, is a numerical statistic that is intended to reflect how important a word is to a document in a collection or corpus. It is often used as a weighting factor in information retrieval and text mining. The TF-IDF value increases proportionally to the number of times a word appears in the document, but is offset by the frequency of the word in the corpus, which helps to control for the fact that some words are generally more common than others.

Variations of the TF–IDF weighting scheme are often used by search engines as a central tool in scoring and ranking a document's relevance given a user query. TF–IDF can be successfully used for stop-words filtering in various subject fields including textsummarization and classification.

* 1. **Jaccard Similarity**

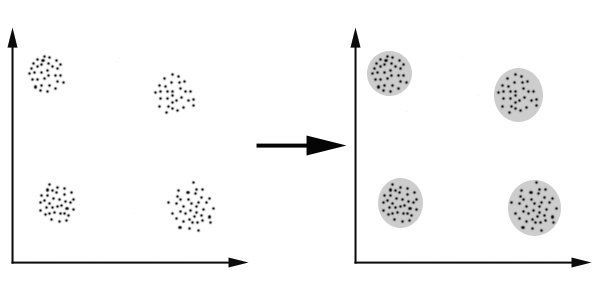
Jaccard similarity is statistic used for comparing the similarity and diversity of sample sets. The Jaccard coefficient measures similarity between finite sample sets, and is defined as the size of the intersection divided by the size of the union of the sample sets:

 J(A,B) = {{|A \cap B|}\over{|A \cup B|}}.

The Jaccard distance, which measures dissimilarity between sample sets, is complementary to the Jaccard coefficient and is obtained by subtracting the Jaccard coefficient from 1, or, equivalently, by dividing the difference of the sizes of the union and the intersection of two sets by the size of the union:

* 1. **Clustering**

Clustering can be considered the most important unsupervised learning problem; so, as every other problem of this kind, it deals with finding a structure in a collection of unlabeled data. A loose definition of clustering could be “the process of organizing objects into groups whose members are similar in some way”. A cluster is therefore a collection of objects which are “similar” between them and are “dissimilar” to the objects belonging to other clusters. We can show this with a simple graphical example:



In this case we easily identify the 4 clusters into which the data can be divided; the similarity criterion is distance: two or more objects belong to the same cluster if they are “close” according to a given distance (in this case geometrical distance). This is called distance-based clustering. Another kind of clustering is conceptual clustering: two or more objects belong to the same cluster if this one defines a concept common to all that objects. In other words, objects are grouped according to their fit to descriptive concepts, not according to simple similarity measures.

* 1. **K-means Clustering**

K-means is one of the simplest unsupervised learning algorithms that solve the well-known clustering problem. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume k clusters) fixed a priori. The main idea is to define k centroids, one for each cluster. These centroids should be placed in a cunning way because of different location causes different result. So, the better choice is to place them as much as possible far away from each other. The next step is to take each point belonging to a given data set and associate it to the nearest centroid. When no point is pending, the first step is completed and an early groupage is done. At this point we need to re-calculate k new centroids as barycenters of the clusters resulting from the previous step. After we have these k new centroids, a new binding has to be done between the same data set points and the nearest new centroid. A loop has been generated. As a result of this loop we may notice that the k centroids change their location step by step until no more changes are done. In other words centroids do not move any more.

Finally, this algorithm aims at minimizing an objective function, in this case a squared error function. The objective function

http://home.deib.polimi.it/matteucc/Clustering/tutorial_html/images/image009.gif

Wherehttp://home.deib.polimi.it/matteucc/Clustering/tutorial_html/images/image011.gif is a chosen distance measure between a data pointhttp://home.deib.polimi.it/matteucc/Clustering/tutorial_html/images/image013.gif and the cluster centerhttp://home.deib.polimi.it/matteucc/Clustering/tutorial_html/images/image015.gif, is an indicator of the distance of the n data points from their respective cluster centers.

The algorithm is composed of the following steps:

1. Place K points into the space represented by the objects that are being clustered. These points represent initial group centroids.
2. Assign each object to the group that has the closest centroid.
3. When all objects have been assigned, recalculate the positions of the K centroids.
4. Repeat Steps 2 and 3 until the centroids no longer move. This produces a separation of the objects into groups from which the metric to be minimized can be calculated.

Although it can be proved that the procedure will always terminate, the k-means algorithm does not necessarily find the most optimal configuration, corresponding to the global objective function minimum. The algorithm is also significantly sensitive to the initial randomly selected cluster centers. The k-means algorithm can be run multiple times to reduce this effect.

* 1. **Naïve Bayes Classifier**

A Naive Bayes classifier is a simple probabilistic classifier based on applying Bayes theorem (from Bayesian statistics) with strong (naive) independence assumptions. A more descriptive term for the underlying probability model would be "independent feature model".

In simple terms, a naive Bayes classifier assumes that the presence (or absence) of a particular feature of a class is unrelated to the presence (or absence) of any other feature. For example, a fruit may be considered to be an apple if it is red, round, and about 4" in diameter. Even if these features depend on each other or upon the existence of the other features, a naive Bayes classifier considers all of these properties to independently contribute to the probability that this fruit is an apple. Depending on the precise nature of the probability model, naive Bayes classifiers can be trained very efficiently in a supervised learning setting. In many practical applications, parameter estimation for naive Bayes models uses the method of maximum likelihood; in other words, one can work with the naive Bayes model without believing in Bayesian probability or using any Bayesian methods.

An advantage of the naive Bayes classifier is that it only requires a small amount of training data to estimate the parameters (means and variances of the variables) necessary for classification. Because independent variables are assumed, only the variances of the variables for each class need to be determined and not the entire covariance matrix.

1. **Implementation** 
   1. **News crawling**

For crawling the news we have used Pipilika search engines crawler. News contains the title of the news, domain name, published date and location in some cases. The crawled news looks like this:

|  |
| --- |
| <Index>  <filePath>F:\Data WareHouse\small repository\Crawler\_Data\www.amadershomoy2.com\12-7-2012-12-48-16</filePath>  <byteInfo>1 3700861728 3700861791 3700861791 3700870240</byteInfo>  <indexed>true</indexed>  <TITLE> রাজধানীতে ছিনাতাইকারীর ছুরিকাঘাতে নিহত ১, আহত ১</TITLE>  <CONTENT> রাজধানীতে ছিনাতাইকারীর ছুরিকাঘাতে নিহত ১, আহত ১ ||  অসধফবৎঝযড়সড়ু.ঈড়স (আমাদের সময়.কম) জুলাই ১২, ২০১২, বৃহস্পতিবার : আষাঢ় ২৮, ১৪১৯ । আপডেট বাংলাদেশ সময় রাত ১২:০০ অৎপযরাব আজব্জ€্বকর পাতাসমঞ্চহ রাজধানীতে ছিনাতাইকারীর ছুরিকাঘাতে নিহত ১, আহত ১ নিজস্ব প্রতিবেদক আমাদের সময়.কম ইসমাইল হোসেন ইমু ও জোনায়েদ মানসুর : রাজধানীর মহাখালীতে ছিনাতাইকারীর ছুরির আঘাতে মাছ ব্যবসায়ী নিহত। তার নাম হাফিজ উদ্দিন (৪০) । এ ঘটনায় ইউসুফ (১৫) নামে আরেক জন আহত হয়েছেন। আজ ভোর সাড়ে পাচঁটার দিকে এ ঘটনা ঘটে। ইউসুফকে ঢাকা মেডিকেল কলেজ হাসপাতালে ভর্তি করা হয়েছে। ইউসুফ হাসপাতালে সাংবাদিকদের জানায়, তারা দু""জনই গাজীপুর এলাকার মাছ ব্যবসায়ী। ভোরে কাওরান বাজারের উদ্দেশ্যে গাজীপুর থেকে রওনা হলে মহাখালী ফ্লাইওভারের নিচে পৌছঁালে একটি সাদা মাইক্রোবাস তাদের গতিরোধ করে। মাইক্রোবাস থেকে এক সাদা পোশাকধারী ছুরি বের করে তাদের জিম্মি করে। টাকা ও মোবাইল চাইলে তারা দিতে রাজি না হওয়ায় দুর্বৃত্তরা তাদের দু""জনকে এলোপাতাড়ি ছুরিকাঘাত করে পালিয়ে যায়। এ ব্যাপারে শিল্পাঞ্চল থানায় পুলিশ আজ বেলা সাড়ে ১১টা পর্যন্ত এ ঘটনা জানেনা বলে আমাদের সময়.কমকে জানায়। বিস্তারিত আসছে----------- স্থানীয় সময়: ১২.১১ ঘন্টা, ১২ জুলাই ২০১২ বদরুল বোরহান / </CONTENT>  <CATEGORY> অন্যান্য</CATEGORY>  <CITY> ঢাকা গাজীপুর</CITY>  <DOMAIN> www.amadershomoy2.com</DOMAIN>  <DATE> 201207120642</DATE>  <URL> http://www.amadershomoy2.com/content/2012/07/12/middle0103.htm/</URL>  <TYPE> news</TYPE>  <PATH> F:\Data WareHouse\small repository\Crawler\_Data\www.amadershomoy2.com\12-7-2012-12-48-16</PATH>  <BYTE\_INFO> 1 3700861728 3700861791 3700861791 3700870240</BYTE\_INFO>  </Index> |

* 1. **Parsing the crawled news**

For parsing relevant data from news we have used JSOUP library.

* 1. **Extracting the keywords from different news**
  2. **Indexing the root words**
  3. **Extracting locations and dates**
  4. **Finding similarity of different news**

To find the similarity between different news, first we have calculated the Term Frequency (TF) of the inputted news, then the Inverse Document Frequency (IDF) and finally the cosine similarity. Measurement of cosine similarity ensures the similarity between different news**.** Similarity helps to distinguish the similar type crime in the particular area. We also use this cosine similarity to find the same news published in different newspapers. For finding the same news first we calculate the cosine similarity but cosine similarity is not enough for this. We also find the occurrence date and the location where this crime scene happened. If the date and location is same and similarity value is greater than a threshold value then we decide that this documents are same. Suppose we have several news having words like “ছিনতাই”, “খুন”, “ডাকাতি” and he we have calculated the TF value of those words in the below documents.

|  |
| --- |
| 1. মেহেরপুর সদর উপজেলার মেহেরপুর-রাইপুর সড়ক থেকে পুলিশের তালিকাভুক্ত সন্ত্রাসী খাদেমুল ইসলামকে (৩০) ছিনতাই প্রস্ততিকালে একটি পিস্তল ও একটি ম্যাগজিনসহ আটক করেছে পুলিশ। রোববার মধ্যরাতে তাকে আটক করা হয়। 2. আপডেট বাংলাদেশ সময় রাত ১২:০০ অৎপযরাব আজব্জ€্বকর পাতাসমঞ্চহ নাটোরে ডাকাতি, গৃহকর্তা আহত ডেস্ক রিপোর্ট : নাটোর সদর উপজেলার নলডাঙ্গা থানার মাধনগর গ্রামে দুধর্ষ ডাকাতি হয়েছে। 3. রাহ্মণবাড়িয়ায় ২ ব্যবসায়ী খুন, সাভার ও চৌদ্দগ্রামে যুবকের লাশ উদ্ধার : রাজধানীতে গৃহবধূকে গলা কেটে হত্যা ডেস্ক রিপোর্ট সাভারে মাথাবিহীন এক যুবকের লাশ উদ্ধার করেছে পুলিশ। ব্রাহ্মণবাড়িয়ায় ২ ব্যবসায়ীকে খুন করেছে সন্ত্রাসীরা । |

* 1. **Categorizing the news**
  2. **Finding and removing the same news.**

A single news can be published in different newspapers. For calculating the exact crime occurrence, we’ve to remove the repetitive news form the sample data. To calculate this we use cosine similarity for matching the document and then find the location and published date. If the crime occurred date and locations are same and similarity is greater than a threshold value then we can assume that this two documents are same. Suppose we have doc1 and doc2 two news from different news source. If the similarity S[i, j] between this doc is greater than 60% and the location of loc-i and loc-j is same and there publishing date pi and pj is same then this two document is similar. If there are n documents then the complexity will be O (n^2). Because we need to calculate all pair similarity.

* 1. **Finding different crime categories**

1. **Mapping the extracted data.**
   1. **Designing the map**
   2. **Plotting the data on map based on crime categories and location**
2. **Statistical analysis** 
   1. **Finding different crime rates in different locations.**
   2. **Ranking locations based on crimes.**
   3. **Finding patterns**
   4. **Predicting future crimes**