

2014/5/12

---

---

---

# Hot Topic Data Analysis and Identification System

*CSIT 6910 - Independent Project (Spring 2014 semester)*

---

---

---

XU Yunlong | [yxiao@connect.ust.hk](mailto:yxiao@connect.ust.hk)  
Supervisor: Prof. Rossiter

# Hot Topic Data Analysis and Identification System

*CSIT 6910 - Independent Project (Spring 2014 semester)*

## Table of Contents

1	Introduction.....	3
2	Main functionalities .....	3
2.1	Crawling news.....	3
2.2	HTML analysis.....	3
2.3	Keyword abstraction.....	3
2.4	Word to vector .....	3
2.5	Clustering.....	3
2.6	Ranking.....	3
2.7	Front end .....	4
3	Design issue .....	4
4	Implementation.....	4
4.1	Development environment.....	4
4.2	Third party libraries and frameworks .....	4
4.2.1	Heritrix .....	4
4.2.2	MOA.....	4
4.2.3	cx-extractor.....	4
4.2.4	Ansj .....	4
4.2.5	word2vec.....	4
4.2.6	commons-math .....	5
4.2.7	Spring.....	5
4.2.8	Maven .....	5
4.2.9	jQuery .....	5
5	Activity diagram .....	5
6	Class diagram .....	6

7	Mechanism .....	6
7.1	Crawler.....	6
7.2	HTML analyzer .....	7
7.3	Semantics analyzer .....	8
7.3.1	Chinese word segmentation and keywords abstraction .....	8
7.3.2	Word2Vector .....	11
7.4	Clustering.....	11
7.4.1	Requirements for data stream mining.....	12
7.4.2	Strategy.....	12
7.4.3	Stream clustering algorithms .....	13
7.4.4	Demonstration .....	14
7.5	Ranking.....	14
7.6	Front end .....	15
8	Acknowledgement .....	17
9	Summary.....	17
10	Appendix.....	17
11	Minutes .....	18
11.1	Minutes of the 1 <sup>st</sup> Project Meeting .....	18
11.2	Minutes of the 2 <sup>nd</sup> Project Meeting .....	19
11.3	Minutes of the 3 <sup>rd</sup> Project Meeting .....	20
11.4	Minutes of the 4 <sup>th</sup> Project Meeting.....	21

## 1 Introduction

---

With the coming of information age, the Internet generates a huge number of informations every day. Some people are suffering from how to obtain useful informations from the Internet.

After consulting my friends and relatives, I think most of people are interested in hot topics. Therefore, I want to try to find the hot topics for people by information technology, which is the purpose of my independent project, hot topic data analysis and identification system. The system looks like a vertical search engine, it crawls the latest news from news website from time to time and pushes the crawled webpage to the analysis system to trigger identification process. Eventually, the result, hot news, will be displayed in a website. People can browse the website for the latest breaking news.

In terms of technology, the system is consist of web crawling, html analysis, semantic analysis, data mining and website. Besides, the whole system is designed as an open architecture, which allows me to configure and enhance any component at any time without impacting the others. Implementing such a large system in three months is not an easy task, therefore, I design and develop the system with the help of some third-party libraries and frameworks, which is similar with the work flow in enterprise.

## 2 Main functionalities

---

To make the system easy to maintain, I divide the system into many independent components, which are crawler, html analyzer, semantic analyzer, data miner and front end website. The components are organized by abstract interface, instead of specific implementation, which means that they are decoupled. The functionality of the components are as follows.

### 2.1 Crawling news

The system is driven by a customized web spider based on Heritrix, a famous open-source web crawler designed for web archiving. The crawler runs scheduled tasks for getting the latest news from news website, and feed the news to identification system.

### 2.2 HTML analysis

The raw data from crawler is the HTML of the news page, the HTML analysis system is responsible for abstracting the main body (without html tags) from the complex HTML file.

### 2.3 Keyword abstraction

Keywords of the news are good for later statistics. The keyword abstraction system is responsible for abstracting the keywords from the main body of the news based on the title and term frequency.

### 2.4 Word to vector

To quantify the news, I project the keywords into a 200-dimension vector space based on semantics. Each vector in the space stands for a news, which is prepared for data mining process.

### 2.5 Clustering

To find the hot topics, I cluster the vectors of the news I gain from the crawler. A big cluster means that there are many news about one topic, which may be the hot topic.

### 2.6 Ranking

To display the latest hot topics to users, I rank the clustered news by a two-dimension normal distribution model. One dimension is the size of the cluster the news is belong to, the other dimension is the timespan between the post date of the news and ranking time. The weight of the two dimensions

can be configured by normal distribution parameters, which can make the ranking result more reasonable.

## 2.7 Front end

A website is responsible for displaying the hot topics to the users.

## 3 Design issue

---

The architecture of the system is designed as a web-service. The crawler feeds the raw news to the analysis system through a post request. The user access the website through a get request.

What is more, to flexibly organize so many functionalities, I associate the different processors I mentioned above by decorator pattern, which exposes a simple interface to invoke the series of process. As the result, I can attach new improvement to the processor without modifying the existing code, which makes the system extendable and easy to maintain.

## 4 Implementation

---

The system is implemented based on Java, thus, it can be deployed on many operating systems.

### 4.1 Development environment

- Foundation: Java SE Development Kit 7 [1]
- IDE: Eclipse [2]
- Build tool: Maven 3 [3]
- Code management: Github [4]

### 4.2 Third party libraries and frameworks

Never reinvent wheels is the core idea of modern software development. Many talent programmers and associations share their code for free, which decreases our workload a lot. Thus, my project is based on some brilliant open libraries and frameworks. The brief introductions of them are as follows.

#### 4.2.1 Heritrix

Heritrix [5] is a web crawler designed for web archiving. It was written by the Internet Archive. It is free software license and written in Java. The main interface is accessible using a web browser, and there is a command-line tool that can optionally be used to initiate crawls.

#### 4.2.2 MOA

MOA (Massive Online Analysis) [6] is a free open-source software specific for mining data streams with concept drift. Here is a video about stream clustering.

#### 4.2.3 cx-extractor

The cx-extractor [7] is a library for abstracting the main body from html based on block distribution. It is not relative with html tags, so it provides commonality for all kinds of Chinese website, no matter how ugly is the html.

#### 4.2.4 Ansj

Unlike English, Chinese words are not divided by space. Thus, to analyze Chinese sentence, the first step is to split the sentence into words based on part of speech and semantics. Ansj [8] is a library for Chinese Word Segmentation and abstracting the key words.

#### 4.2.5 word2vec

4 •

---

Word2vec [9] provides an efficient implementation of the continuous bag-of-words and skip-gram architectures for computing vector representations of words. These representations can be subsequently used in many natural language processing applications and for further research.

#### 4.2.6 commons-math

Commons Math [10] is a library of lightweight, self-contained mathematics and statistics components addressing the most common problems not available in the Java programming language or Commons Lang.

#### 4.2.7 Spring

Spring [11] helps development teams everywhere build simple, portable, fast and flexible JVM-based systems and applications.

#### 4.2.8 Maven

Maven [3] is a build automation tool used primarily for Java projects. Maven addresses two aspects of building software: First, it describes how software is built, and second, it describes its dependencies.

#### 4.2.9 jQuery

jQuery [12] is a fast, small, and feature-rich JavaScript library. It makes things like HTML document traversal and manipulation, event handling, animation, and Ajax much simpler with an easy-to-use API that works across a multitude of browsers. With a combination of versatility and extensibility, jQuery has changed the way that millions of people write JavaScript.

## 5 Activity diagram

The following activity diagram (Fig.1) shows the workflow of the whole system. There are mainly four components for processing a news.

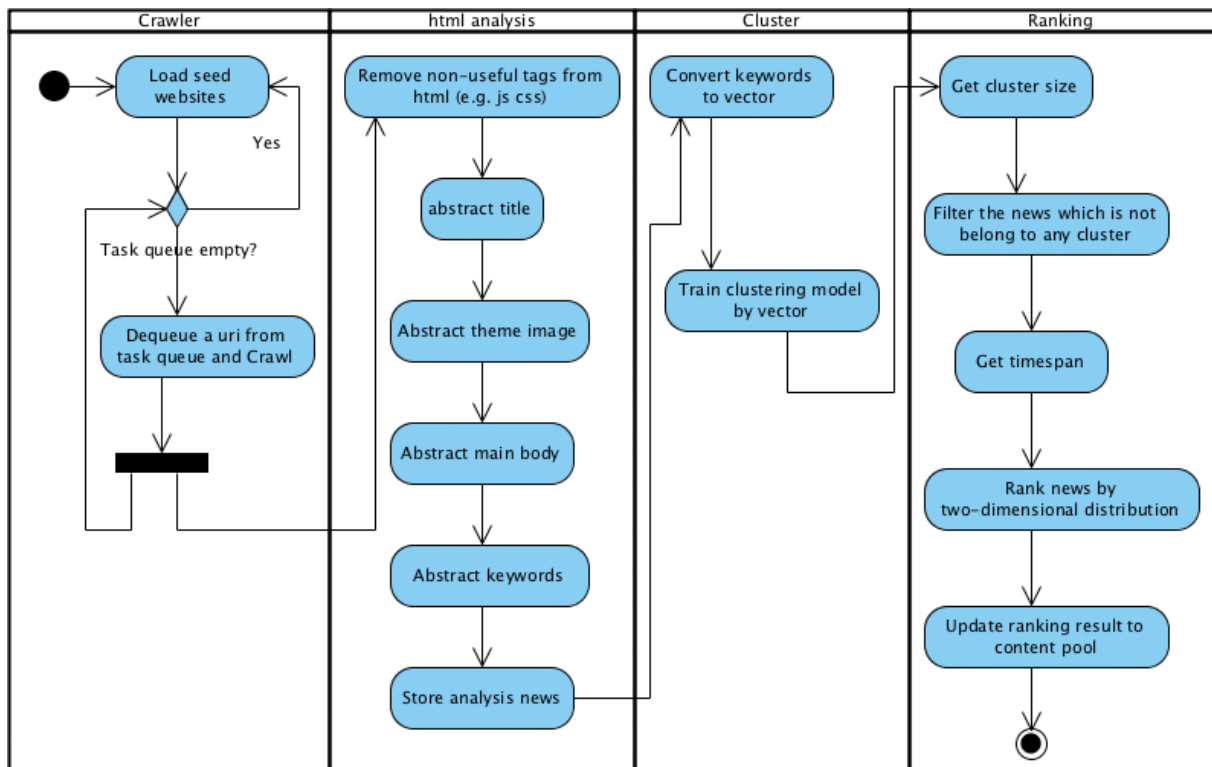


Fig.1 activity diagram of workflow

## 6 Class diagram

A class diagram (Fig.2) of the system is as follow. As the diagram is too large to insert into the report, you could check the large diagram via the link [13].

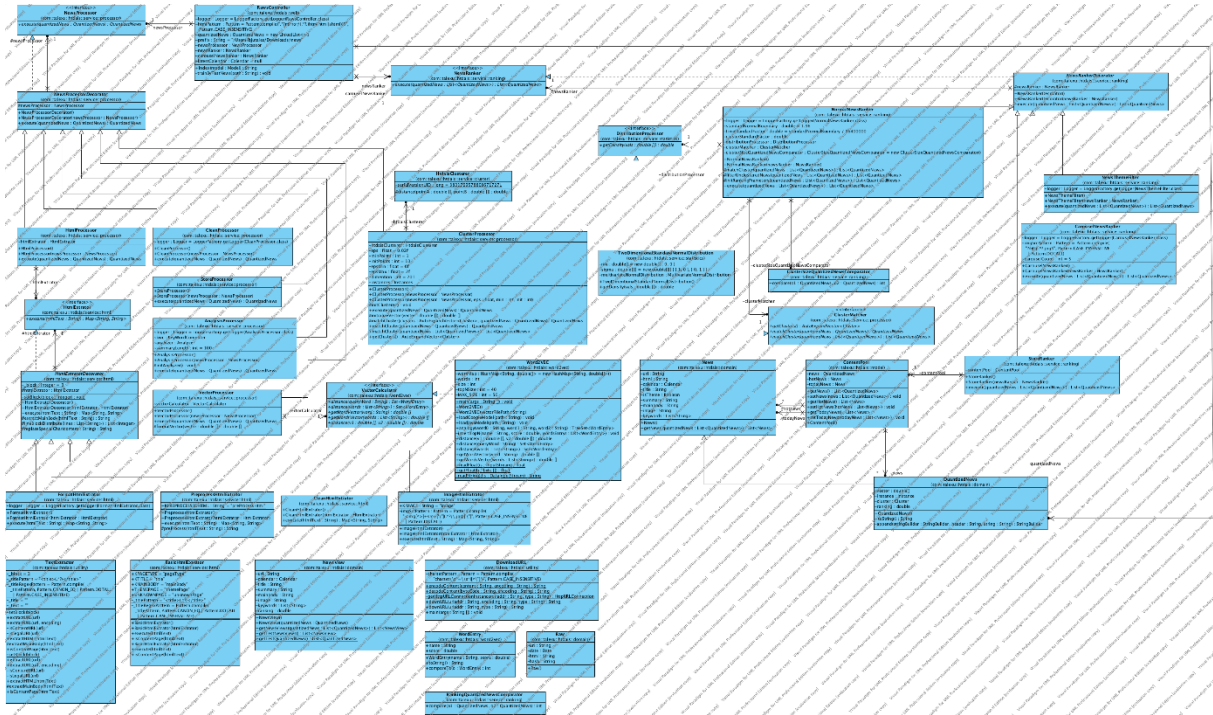


Fig.2 class diagram

## 7 Mechanism

To demonstrate the mechanism of each component, the principles will be explained in details.

### 7.1 Crawler

The crawler is responsible for downloading the content of webpage from a set of seed URIs periodically.

The architecture of the crawler is as shown in Fig.3 and it is designed in Chain of Responsibility Pattern. We are mainly interested in the processing chain components. These components are called in a predefined sequence to process each URI Below is a brief description of each sub component in the chain.

- Prefetch chain: responsible for investigating if the URI could be crawled at this point. that includes checking if all preconditions are met (DNS-lookup, fetching robots, text, authentication)
- Fetch chain: processors in this chain are responsible for getting the data from the remote server.
- Extractor chain: Process the HTML page. Typical functionality is to fetch new links from the web page and feed that back to the frontier.
- Write chain: Writing data to the archive.
- Postprocess chain: Do 'clean up' and return control to the frontier.

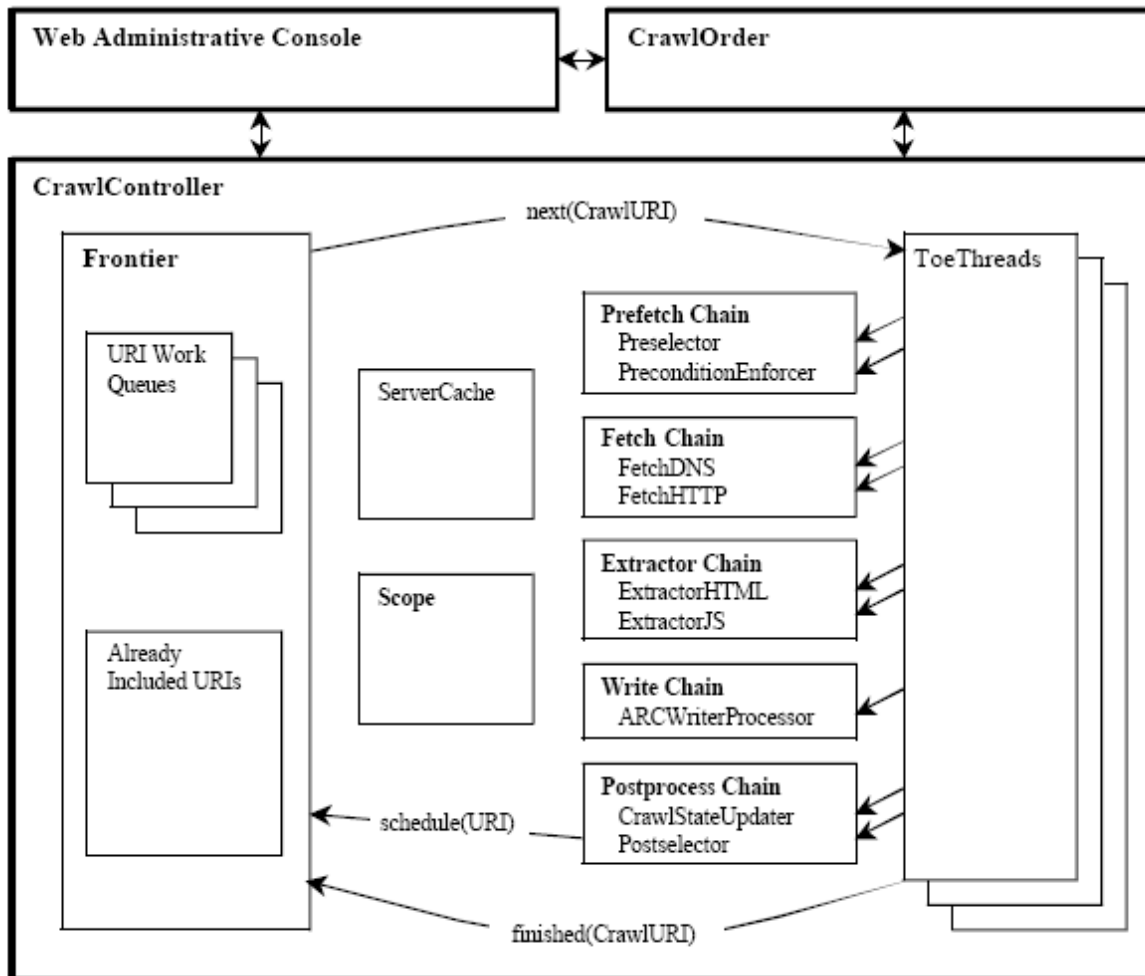


Fig.3 architecture of crawler

## 7.2 HTML analyzer

As the raw data from crawler is HTML, we need to extract the main body from the complex HTML file. A lot of work, such as filtering the JavaScript, CSS and non-useful content, need to be done for the future process. However, the task is not easy to do because only less than 5% of a HTML is main body. The others are advertisement, link to other pages, rubbish informations and so on.

A clever guy invents a robust algorithm to locate the main body based on block distribution. The idea of the algorithm is that after remove the obvious non-useful tag (e.g. JavaScript, CSS, and link), the main body is centralized in a big block. If we successfully find the biggest block, we find the main body.

A block distribution after preprocessing (remove obvious non-useful tags) is show in Fig.4. We may notice a pretty big block between line 107 and 160, which is the main body we want to find.



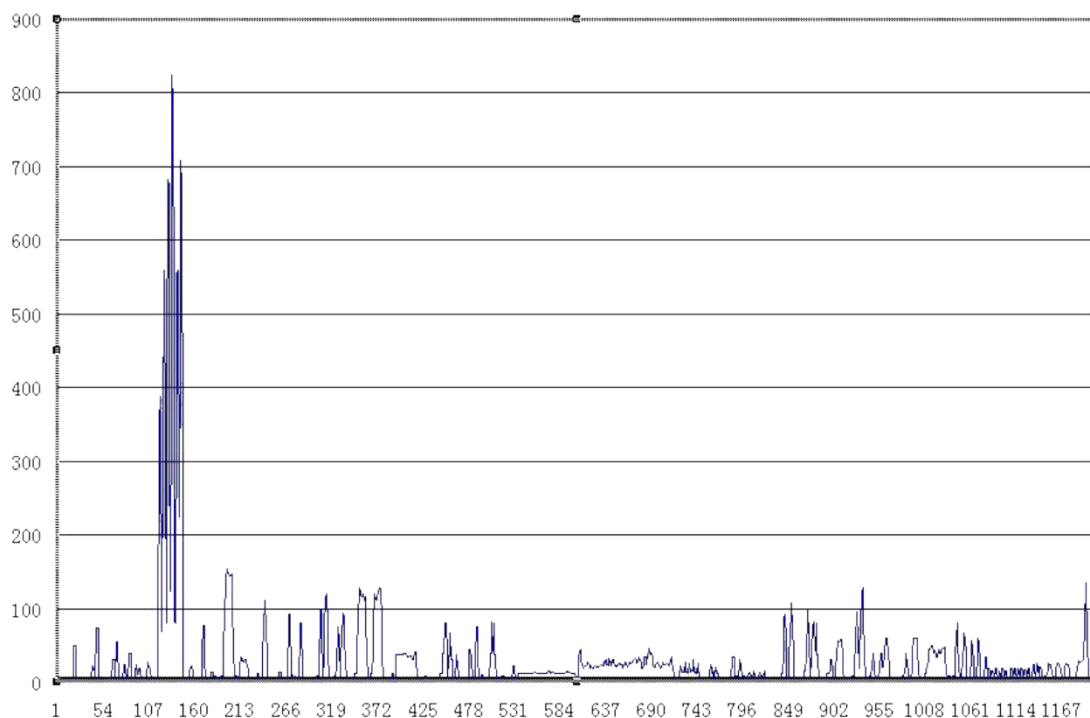


Fig.4 block distribution

## 7.3 Semantics analyzer

Semantics analysis is the most difficult part of the system because Chinese Word Segmentation and the Recognition of Semantics are both extremely academic and technique. Fortunately, two libraries can help to do the tasks, which are Ansj and word2vec I mentioned above.

### 7.3.1 Chinese word segmentation and keywords abstraction

Ansj is very easy to use in my program, I pass a string of sentences into the library, a set of split words will be returned. Besides, based on a corpus, the keywords of a sentence will be returned as well.

For instance, I want to split the following sentences (List.1) and abstract the keywords.

---

5月11日电 据俄新网消息，乌克兰外交部11日发表声明称，顿涅茨克和卢甘斯克州当日举行的全民公投结果对乌克兰领土完整和国家体制不会有法律影响。

乌克兰外交部断言，地区大部分居民没有参与投票。

顿涅茨克和卢甘斯克州11日就地区地位举行全民公投，投票是在基辅当局于两地展开特别行动大背景下举行的，行动期间动用了装甲车和火炮，造成平民死亡。莫斯科认为基辅针对居民动用军队的决定是极端危险的态势发展，称这一行动为“讨伐行动”。

---

---

目前卢甘斯克全民公投投票率截至当地时间 12 时已达到 65%。顿涅茨克的投票率截至 11 时 20 分超过 30%。公投于当地时间 8 时开始，持续至当地时间 20 时。

---

*List. 1 Original Chinese Text*

---

*May 11, according to Itar-Tass news, the Ukrainian Foreign Ministry said in a statement on the 11th, the referendum results in Donetsk and Luhansk Oblast held on the same day there will be no legal effect on Ukraine's territorial integrity and national institutions.*

*Ukrainian Foreign Ministry asserted that most of the residents did not participate in the voting.*

*Donetsk and Luhansk State on the 11th held a referendum on the region's status, voting was held in Kiev during the action the authorities in the two special operations launched against the backdrop of armored vehicles and artillery spent, resulting in civilian deaths. Moscow believes Kiev's decision to use military force against residents is extremely dangerous development of the situation, saying that this action is "punitive action."*

*Currently Luhansk referendum turnout as of 12:00 local time has reached 65%. Turnout in Donetsk ended 11:20 over 30%. Started at local time 8:00 referendum, continued until 20:00 local time.*

---

*List.2 Translation of List.1*

### 7.3.1.1 Split words

I pass the sentences above as a single string to Ansj, the split words (List.3) are returned as following pattern, **word / part of speech**. Stop words (e.g. a, an, the) will be filtered at this stage.

---

5 月/m    11 日/m    电/n    据/p    俄/j    新网/nz  
 消息/n , /w    乌克兰/ns    外交部/nt    11 日/m    发表声  
 明/v    称/v    , /w    顿涅茨克/nrf    和/c    卢甘斯克州/ns 当日/t  
 举行/v 的/uj    全民/n 公投/nz    结果/n 对/p    乌克兰/ns

领土完整/nz 和/c 国家/n 体制/n 不会/v 有/v 法律/n 影响/vn。 /w

乌克兰/ns 外交部/nt 断言/v， /w 地区/n 大部分/n 居民/n 没有/v 参与/v 投票/v。 /w

顿涅茨克/nrf 和/c 卢甘斯克州/ns11 日/m 就/d 地区/n 地位/n 举行/v 全民/n 公投/nz， /w 投票/v 是/v 在/p 基辅/n 当局/n 于/p 两地/n 展开/v 特别/d 行动/vn 大/a 背景/n 下/f 举行/v 的/uj， /w 行动/vn 期间/f 动用/v 了/ul 装甲车/n 和/c 火炮/n， /w 造成/v 平民/n 死亡/v。 /w 莫斯科/ns 认为/v 基辅/n 针对/p 居民/n 动用/v 军队/n 的/uj 决定/v 是/v 极端/n 危险/an 的/uj 态势/n 发展/vn， /w 称/v 这/r 一/m 行动/vn 为/p “/w 讨伐/v 行动/vn ” /w。 /w

目前/t 卢甘斯克/nrf 全民/n 公投/nz 投票率/n 截至/v 当地/s 时间/n 12/m 时/ng 已/d 达到/v 65%/m。 /w 顿涅茨克/nrf 的/uj 投票率/n 截至/v 11/m 时/ng 20 分/m 超过/v 30%/m。 /w 公投/nz 于/p 当地/s 时间/n 8/m 时/ng 开始/v， /w 持续/vd 至/p 当地/s 时间/n 20/m 时/ng。 /w

---

*List.3 Split word and its corresponding part of speech*

### 7.3.1.2 Keywords

The keywords (List.4) of the sentences returned by Ansj are as follows.

---

乌克兰外交部, 顿涅茨克, 全民公投, 乌克兰领土完整, 乌克兰外交部, 行动, 当地时间

---

*List.4 Keywords of the sentences in List.1*

---

*Ministry of Foreign Affairs of Ukraine, Donetsk, referendum, the territorial integrity of Ukraine, the Ukrainian Foreign Ministry, action, Local*

---

*List.5 Translation of List.4*

The report is talking about independence in Ukraine, we can see that the keywords appropriately reflect the core content of the report. Usually, we can believe that the keywords abstraction of Ansj is correct.

### 7.3.2 Word2Vector

Another challenging task is to quantify the news I gain. The library of word2vec provided by google is a library for projecting a word into a high dimensional vector space based on training corpus.

The word2vec tool takes a text corpus as input and produces the word vectors as output. It first constructs a vocabulary from the training text data and then learns vector representation of words. The resulting word vector file can be used as features in many natural language processing and machine learning applications.

A simple way to investigate the learned representations is to find the closest words for a user-specified word. The distance tool serves that purpose. For example, if you enter 'france', distance will display the most similar words and their distances to 'france', which should look like (List.6):

Word	Cosine distance
spain	0.678515
belgium	0.665923
netherlands	0.652428
italy	0.633130
switzerland	0.622323
luxembourg	0.610033
portugal	0.577154
russia	0.571507
germany	0.563291
catalonia	0.534176

*List.6 Cosine similarity of word2vec*

It was recently shown that the word vectors capture many linguistic regularities, for example vector operations  $\text{vector}(\text{'Paris'}) - \text{vector}(\text{'France'}) + \text{vector}(\text{'Italy'})$  results in a vector that is very close to  $\text{vector}(\text{'Rome'})$ , and  $\text{vector}(\text{'king'}) - \text{vector}(\text{'man'}) + \text{vector}(\text{'woman'})$  is close to  $\text{vector}(\text{'queen'})$ .

To observe strong regularities in the word vector space, it is needed to train the models on large data set, with sufficient vector dimensionality. Using the word2vec tool, it is possible to train models on huge data sets (up to hundreds of billions of words).

The linearity of the vector operations seems to weakly hold also for the addition of several vectors, so it is possible to add several word or phrase vectors to form representation of short sentences.

The quality of the word vectors increases significantly with amount of the training data. I gain a set of Chinese corpus of all the internet for two months period from sogou lab [14].

## 7.4 Clustering

After quantify the news, the next step is to cluster them. As we may gain a large number of news (depend on configuration of deployment), we need to reduce the computation time and memory needed to process large but static data sets. If the data cannot fit into memory, it may be necessary to sample a smaller training set. Alternatively, algorithms may resort to temporary external storage, or only process subsets of data at a time. Commonly the goal is to create a learning process that is linear in the number of examples. The essential learning procedure is treated like a scaled up version of classic machine learning, where learning is considered a single, possibly expensive, operation—a set of training examples are processed to output a final static model.

The data mining approach may allow larger data sets to be handled, but it still does not address the problem of a continuous supply of data. Typically, a model that was previously induced cannot be updated when new information arrives. Instead, the entire training process must be repeated with the new examples included. There are situations where this limitation is undesirable and is likely to be inefficient.

The data stream paradigm has recently emerged in response to the continuous data problem. Algorithms written for data streams can naturally cope with data sizes many times greater than memory, and can extend to challenging real-time applications not previously tackled by machine learning or data mining. The core assumption of data stream processing is that training examples can be briefly inspected a single time only, that is, they arrive in a high speed stream, then must be discarded to make room for subsequent examples. The algorithm processing the stream has no control over the order of the examples seen, and must update its model incrementally as each example is inspected. An additional desirable property, the so-called anytime property, requires that the model is ready to be applied at any point between training examples.

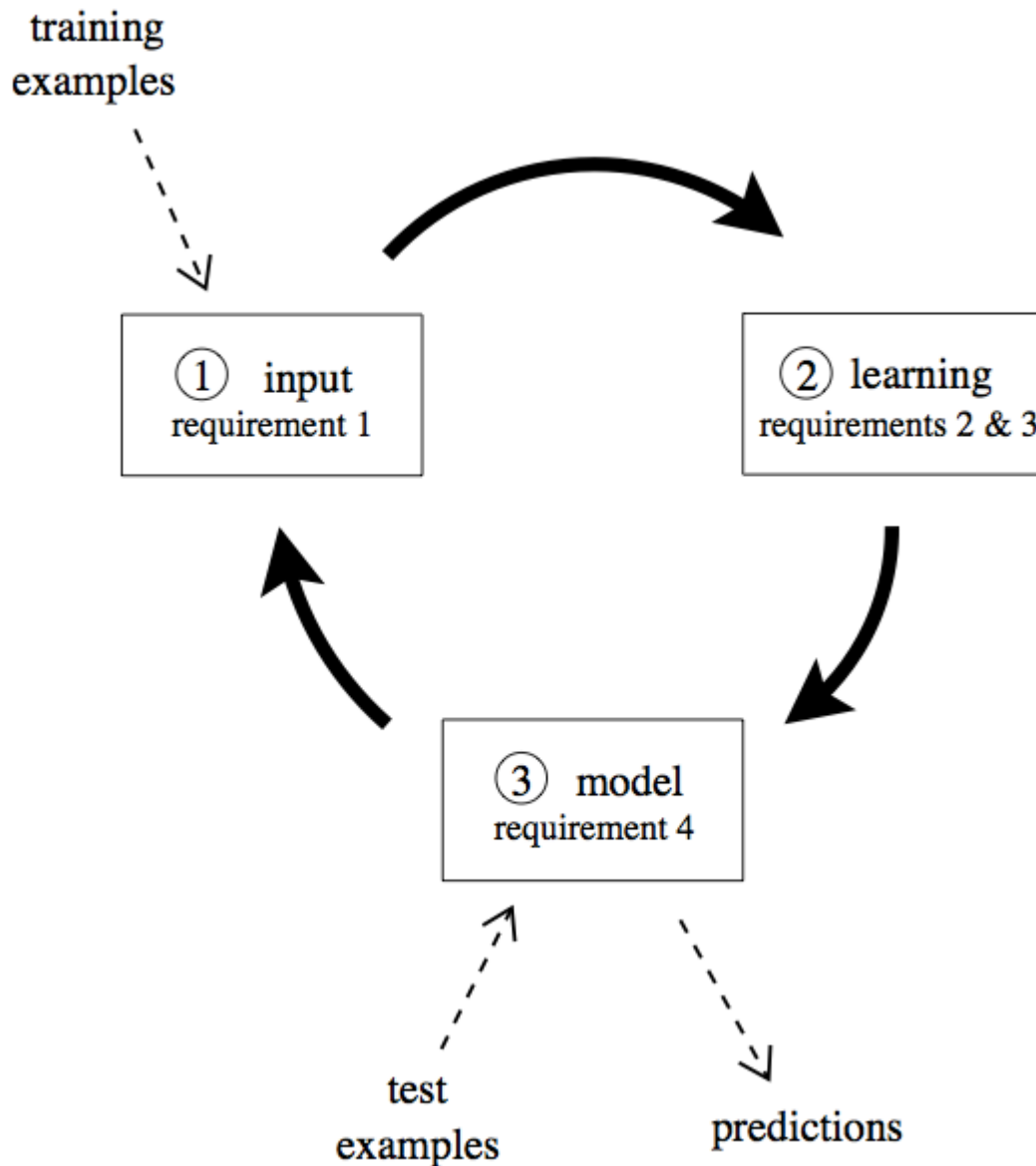
#### 7.4.1 Requirements for data stream mining

A data stream environment has different requirements from the traditional setting. The most significant are the following:

- Requirement 1: Process an example at a time, and inspect it only once (at most).
- Requirement 2: Use a limited amount of memory.
- Requirement 3: Work in a limited amount of time.
- Requirement 4: Be ready to predict at any time.

#### 7.4.2 Strategy

MOA is developed to provide useful insight about classification performance. The data stream classification cycle is as shown in Fig.5.



*Fig.5 The data stream classification cycle*

### 7.4.3 Stream clustering algorithms

Currently MOA contains several stream clustering methods including:

- StreamKM++: It computes a small weighted sample of the data stream and it uses the k-means++ algorithm as a randomized seeding technique to choose the first values for the clusters. To compute the small sample, it employs coresets constructions using a coresets tree for speed up.
- CluStream: It maintains statistical information about the data using micro-clusters. These micro-clusters are temporal extensions of cluster feature vectors. The micro-clusters are stored at snapshots in time following a pyramidal pattern. This pattern allows to recall summary statistics from different time horizons.
- ClusTree: It is a parameter free algorithm automatically adapting to the speed of the stream and it is capable of detecting concept drift, novelty, and outliers in the stream. It uses a compact and self-adaptive index structure for maintaining stream summaries.

- Den-Stream: It uses dense micro-clusters (named core-micro-cluster) to summarize clusters. To maintain and distinguish the potential clusters and outliers, this method presents core-micro-cluster and outlier micro-cluster structures.
- CobWeb. One of the first incremental methods for clustering data. It uses a classification tree. Each node in a classification tree represents a class (concept) and is labeled by a probabilistic concept that summarizes the attribute-value distributions of objects classified under the node.

In my project, I choose Den-Stream for clustering, but we can change different algorithm from configuration file when we deploy the system.

### 7.4.4 Demonstration

To visualize the concept of clustering, the following Fig.6 shows a two-dimensional vector example. Notice that each point in the figure stands for a vector of a news. C0 to C4 are five clusters, because the vectors in the cluster are closed with each other. The cluster is hot topic, the news in the cluster is hot news. In my program, it is 200-dimentional.

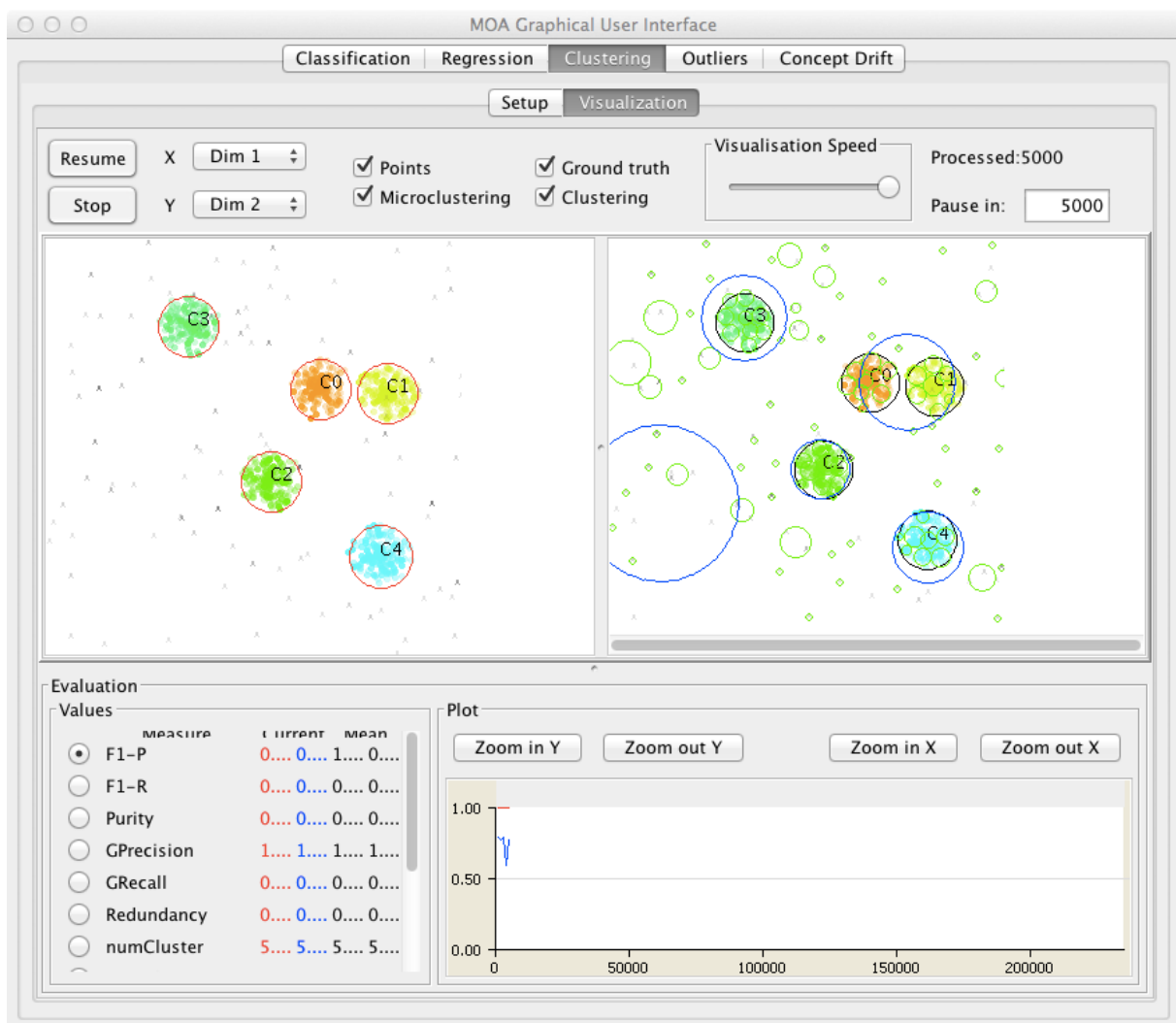


Fig.6 Clustering

## 7.5 Ranking

After clustering, the final task is to rank the hot topics reasonable. I sort them by two factors, news heat and timeliness.

Another consideration is how to balance the two sorting factors. My solution is to project the two factors into a two dimensional normal distribution, each dimension is corresponding to one sorting factor.

The means of the normal distribution are the biggest size of cluster and current sorting time. If the cluster size is big, it means that the topic is hot, and the news in the cluster will have a higher weight. If the post time of a news is closed to current, it means that the news is latest.

The density of the news in the two dimensional normal distribution is the sorting weight. I can also configure the variances and the correlation of the two dimensional normal distribution in order to balance the two sorting factors.

## 7.6 Front end

A concise website is used for displaying the hot news. There is a carousel on the top of the web, and many summary cards are full of the following space. Some pictures of hot topics are shown in the carousel. The summary of a news is displayed on the card. When you click a card, a panel of main body of the news will be pop up, you can also switch to the previous or the next news by clicking the navigation button. Some screen shots of the website are as follows.



Fig.7 Carousel





Fig.8 Summary cards



Fig.9 Pop up panel after clicking the top-right summary card in Fig.8

## 8 Acknowledgement

---

The project is not only academic, but also enterprise. Thank to my supervisor, Prof. Rossiter, for giving me guidance and arranging the progress. Otherwise, the project may not be finished.

## 9 Summary

---

The independent project is a practice for the knowledge I gain in my postgraduate study, including networks, data mining, object-oriented software development of enterprise systems, multimedia and so on.

After finish the project, I deeply understand what I have learned in the courses and utilize the theoretical knowledge into practical project, which is a good experience for my engineering career.

In terms of the project, I implement all the parts of the whole system which is really a huge workload. The system can run as my expectation, crawling information from time to time, identifying hot topics in real time and have a beautiful front end website.

Besides, the architecture of the system is open. In fact, during my development period, I have iterated each component for many times, and it doesn't impact the whole system. The system is easy to maintain and could still be expanded larger and larger in the future.

## 10 Appendix

---

[1] Java SE Development Kit 7:

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

[2] Eclipse: <http://www.eclipse.org>

[3] Maven 3: <http://maven.apache.org>

[4] Code management: <https://github.com/talexu/HTDAIS>

[5] Heritrix: <https://webarchive.jira.com/wiki/display/Heritrix/Heritrix>

[6] MOA: <http://moa.cs.waikato.ac.nz>

[7] cx-extractor: <https://code.google.com/p/cx-extractor>

[8] Ansj: [https://github.com/ansjsun/ansj\\_seg](https://github.com/ansjsun/ansj_seg)

[9] word2vec: <https://code.google.com/p/word2vec>

[10] commons-math: <http://commons.apache.org/proper/commons-math>

[11] Spring: <http://spring.io>

[12] jQuery: <http://jquery.com>

[13] Class diagram: <http://1drv.ms/1qHISim>

[14] Sogou lab corpus: <http://www.sogou.com/labs/dl/ca.html>

## 11 Minutes

---

There are four meetings with Prof. Rossiter for my project.

### 11.1 Minutes of the 1<sup>st</sup> Project Meeting

- Date: Friday, 21 February 2014
- Time: 2:10 PM
- Place: Room 3512
- Attending: Prof. Rossiter, XU Yunlong
- Absent: None
- Recorder: XU Yunlong
- Approval of minutes: This is first formal meeting, so there were no minutes to approve.
- Report on progress: An initial technical research for my project about web crawler and html analyzer.
- Discussion Items and Things To Do:
  - The workflow of the whole system.
  - The ranking strategy.
  - User interaction.
- Meeting adjournment: 2:30 PM

## 11.2 Minutes of the 2<sup>nd</sup> Project Meeting

- Date: Friday, 21 March 2014
- Time: 2:10 PM
- Place: Room 3512
- Attending: Prof. Rossiter, XU Yunlong
- Absent: None
- Recorder: XU Yunlong
- Approval of minutes: The minutes of the last meeting were approved without amendment.
- Report on progress: Finish the demo of semantics analyzer. It can project a word into a 200 dimensional vector space, and calculate the distance of two vectors by cosine similarity.
- Discussion Items and Things To Do:
  - The accuracy of the semantics model.
- Meeting adjournment: 2:30 PM

## 11.3 Minutes of the 3<sup>rd</sup> Project Meeting

- Date: Wednesday, 23 April 2014
- Time: 1:30 PM
- Place: Room 3512
- Attending: Prof. Rossiter, XU Yunlong
- Absent: None
- Recorder: XU Yunlong
- Approval of minutes: I was asked to let the system runnable and show the analyzing result.
- Report on progress: Integrate the components together, including html analyzer, semantics analyzer, ranking processor and a web service return the result in JSON format.
- Discussion Items and Things To Do:
  - The front end website development.
- Meeting adjournment: 1:50 PM

## 11.4 Minutes of the 4<sup>th</sup> Project Meeting

- Date: Wednesday, 14 May 2014
- Time: 1:30 PM
- Place: Room 3512
- Attending: Prof. Rossiter, XU Yunlong
- Absent: None
- Recorder: XU Yunlong
- Approval of minutes: I was asked to finish the webpage, report and video.
- Report on progress: Finish everything, including project, report and video.
- Discussion Items and Things To Do:
  - Optimize the experience of website.
  - Improve report for details.
  - Improve video for explanation.
- Meeting adjournment: 2:00 PM