공학석사 학위논문

A Real-time News Monitoring System for Trend Analysis

추세 분석을 위한 실시간 뉴스 모니터링 시스템

2016 년 8 월

서울대학교 대학원

전기.정보공학부

왕재환

A Real-time News Monitoring System for Trend Analysis

지도 교수 서종모

이 논문을 공학석사 학위논문으로 제출함
2016 년 8 월

서울대학교 대학원
전기.정보공학부
왕재환

왕재환의 공학석사 학위논문을 인준함
2016 년 8 월

위 원 장 윤성로 (인)

부위원장 서종모 (인)

위 원 한규섭 (인)

Abstract

A Real-time News Monitoring System for Trend Analysis

Wang Zihuan
Electrical and Computer Engineering
The Graduate School
Seoul National University

In recent years, internet news has become one of the most important channel for obtaining information. More and more people read the news through internet-connected computer, tablet, and smartphones. At the same time, the number of online media is also increasing explosively. News are constantly being reproduced, and as a result, the number of news is growing very fast. Consequently, getting the key point or issue from large amount of news in the internet is of great interest.
This thesis introduces a real-time news monitoring system which can crawl internet news every day and how to extract keywords from the crawled news data to obtain the hot issues and its relationships.

Keywords: News Monitoring, Nature Language Processing, Trend Analysis, Internet News
Student Number: 2014-25150

Contents

List of Tables

List of Figures

1. Introduction

In recent years, internet news has become one of the most important channel for obtaining information. More and more people read the news through internet-connected computer, tablet, and smartphones. At the same time, the number of online media is also increasing explosively. News are constantly being reproduced, and as a result, the number of news is growing very fast. Consequently, getting the key point or issue from large amount of news in the internet is of great interest.

This thesis covers the related work, theory, system implementation including how to crawl online news from internet, and how the system handles the online news data to obtain important information.

1.1 Motivation

The main purpose of this thesis is to investigate if and how the internet news can be used to monitor events of high interest and then derive and track important information. This news monitoring system can not only be used for news monitoring, but it also can be a data monitoring system if the input data format is the same as news data and that is why the principle behind the system will be explained in detail.

1.2 Limitation

Once the amount of data grows, it will be more of a struggle to handle the entire system rather than focusing on finding what is behind the data. We are using the free MySQL database to store all the data.

2. Related Work

2.1 Natural language processing

Natural Language Processing (NLP) is an area of research about how to let the computer understand and manipulate natural language to do useful things. It began in the 1950s. We aim to acquire knowledge on how human beings understand and use human language so that appropriate tools and techniques can be implemented to let computer understand and manipulate natural languages to perform desired tasks. The foundations of NLP lie in a number of disciplines: computer and information sciences, linguistics, mathematics, electrical and computer engineering, artificial intelligence (AI) and robotics. Applications of NLP include a number of fields of search, such as machine translation, natural language text processing and summarization, user interfaces and cross-language information retrieval (CLIR), speech recognition, artificial intelligence, and expert systems. [18]

Language processing has history nearly as old as that of computers and comprises a large amount of work. However, many early attempts remained in the stage of laboratory demonstrations or failure. Significant applications are not quite ready for the industry yet, and they are still relatively scarce compared to the universally deployed technologies such as operating systems, databases, and networks. Nevertheless, the number of commercial applications or significant laboratory prototypes embedding language processing techniques is increasing. Examples include:

- Text indexing and information retrieval from the Internet.
- Spelling and grammar checkers.
- Interactive voice response applications.
- Machine translation
- Speech dictation of letters or reports.
- Conversational agents
- Voice control of domestic devices.

2.2 Part-of-speech tagging

Part of Speech (POS) tagging is the problem of assigning each word in a sentence the part of speech that it assumes in that sentence.

In corpus linguistics, part-of-speech tagging (POS tagging or POST), also called grammatical tagging or word-category disambiguation, which is the process of marking up a word in a text corresponding to a particular part of speech based on both its definition and its context—i.e., its relationship with adjacent and related words in a phrase, sentence, or paragraph.

A simplified form of this is commonly taught to school-age children, in the identification of words as nouns, verbs, adjectives, adverbs, etc.

Different languages have different tagging methods. Korean is an agglutinative language with a very productive inflectional system. Inflections include postpositions, suffixes and prefixes on nouns, tense morphemes, honorifics and other endings on verbs and adjectives. Table 2.1 shows the Korean tags.

Table 2.1 Korean Tags

Tag	Description	Tag	Description
NNG	일반 명사	JX	보조사
NNP	고유 명사	EP	선어말어미
NNB	의존 명사	EF	종결 어미
NR	수사	EC	연결 어미
NP	대명사	ETN	명사형 전성 어미
VV	동사	ETM	관형형 전성 어미
VA	형용사	XPN	체언 접두사
VX	보조 용언	XSN	명사파생 접미사
VCP	긍정 지정사	XSV	동사 파생 접미사
VCN	부정 지정사	XSA	형용사 파생 접미사
MM	관형사	XR	어근
MAG	일반 부사	SF	마침표, 물음표, 느낌표
MAJ	접속 부사	SE	줄임표
IC	감탄사	SS	따옴표,괄호표,줄표
JKS	주격 조사	SP	쉼표,가운뎃점,콜론,빗금
JKC	보격 조사	SO	붙임표(물결,숨김,빠짐)
JKG	관형격 조사	SW	기타기호 (논리수학기호,화폐기호)
JKO	목적격 조사	SH	한자
JKB	부사격 조사	SL	외국어
JKV	호격 조사	SN	숫자
JKQ	인용격 조사	NF	명사추정범주
JC	접속 조사	NV	용언추정범주
		NA	분석불능범주

2.3 Dendrogram

A dendrogram is a branching diagram that represents the relationship of similarity among a group of entities.

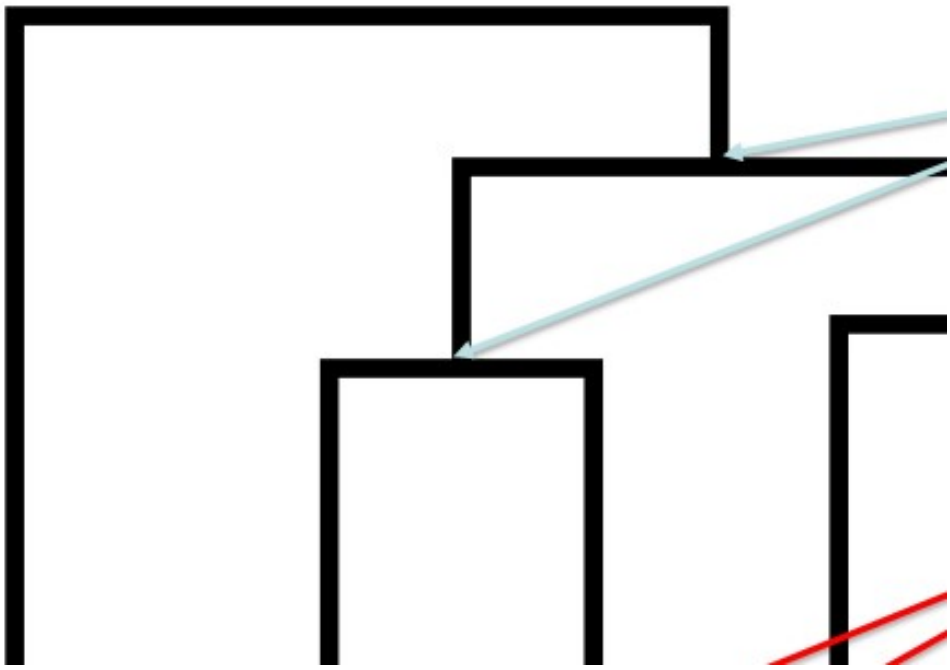

Figure 2.1 A dendrogram example

A basic dendrogram is shown in Fig. 2.1. Each branch is called a clade. The end of each clade is called a leaf. A clade may have just one leaf or they can have more than one leaves. Two-leaved clades are called bifolious, three-leaved are called trifolious, and so on. There is no limit to the number of leaves in one clade. The arrangement of the clades represent which leaves are most similar to each other. The height of the branch points indicates how similar or different they are from each other: the greater the height, the greater the difference.

A dendrogram can be used to represent the relationships between any kinds of entities as long as their similarity to each other can be measured. In Lexomic analysis, the distribution of different words among whole texts or segments of texts are compared.

There are two ways to interpret a dendrogram: in terms of large-scale groups or in terms of similarities among individual chunks.

3. Methodology and implementation

3.1 System framework

Getting the key point or issue from a large amount of news in the internet is of great interest. We investigated a system that can “read” internet news articles and extract key points that can be useful in trend analysis. Fig. 3.1 shows an overview of a system which can “read” data then determine the key point.

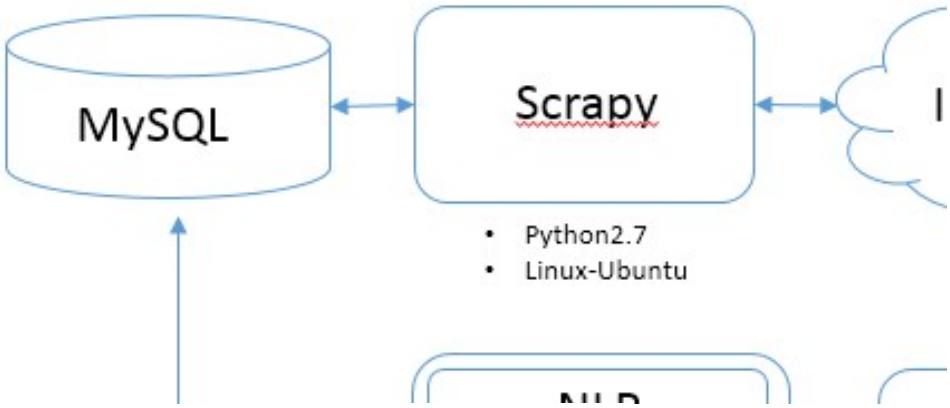

Figure 3.1 System framework

The entire system is composed of two parts: the data crawling part and the analysis part. For data crawling part, a platform named Scrapy is used to crawl data through the internet and save them to database so that the analysis part can read the database and perform analysis such as natural language processing and text analysis to get the desired result.

The entire system is based on Linux environment. Server performance is listed in Table 3.2. The OS we used is Ubuntu. More details will be introduced in Chapter 3.4~3.5.

3.2 Hardware and software

Table 3.1 Hardware performance

CPU	Intel(R) Xeon(R) CPU E3-1220 v3 @ 3.10GHz 4core
MEMORY	16GB
HARD	1TB

Table 3.2 Required software list

Software	**Version**
OS-Ubuntu	14.04.1
Python	2.7
Scrapy	1.0.3
Flask	0.10
MySQL	14.14

3.3 Internet news crawling

Scrapy is a framework used to do crawling task. Even though it is a very powerful crawling framework, some customizing work for each website was required because not all websites open their information to everyone by default. Some of them require login and others hide some information so that the spiders will not have easy access. This chapter will introduce the Scrapy framework and some issues we faced in web crawling.

3.3.1 Introduction of Scrapy

Scrapy is a python based framework which is used to crawl internet news data. Fig. 3.2 shows an overview of the Scrapy architecture with its components and an outline of the data flow that takes place inside the system, which was shown by the green arrows. At first, a start URL is given to Scarpy and tell it to get a specific format data from a specific location. It will download all page sources, find the specific location and save the data to items.

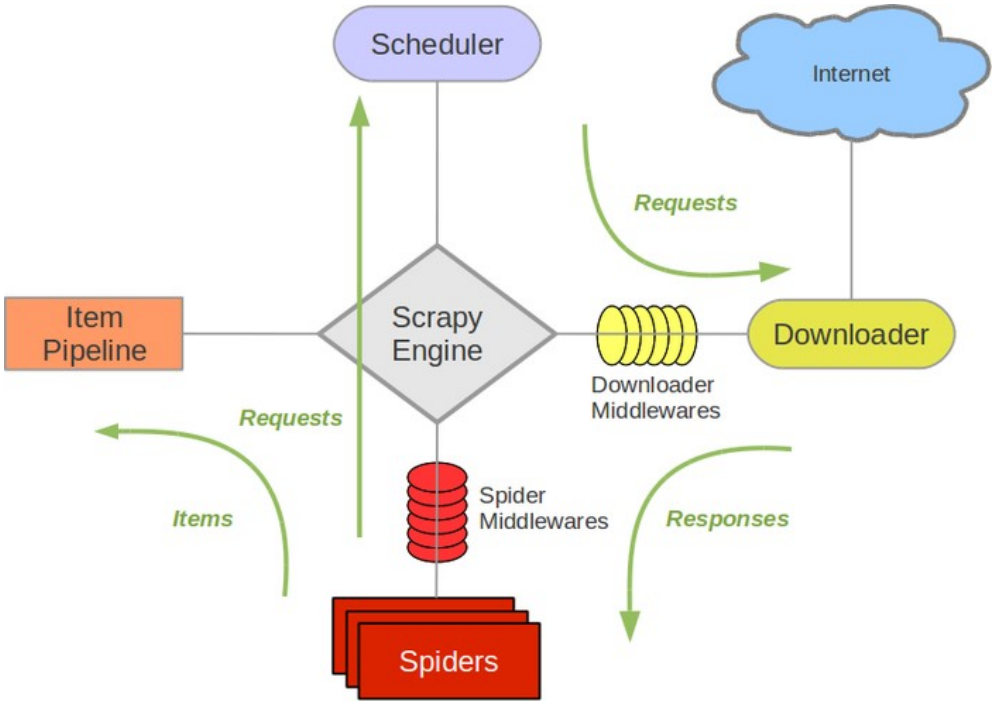

Figure 3.2 Scrapy architecture

Below is the data flow of Scrapy.[16]

a. Engine opens, locates Spider, schedule the first url as a Request.
b. Scheduler sends the url to the Engine, which sends it to Downloader
c. The Downloader sends completed page as a Response through the middleware to the engine.
d. Engine sends Response to the Spider through middleware.
e. Spiders sends Items and new Requests to the Engine.
f. The Spider processes the Response and returns scraped items and new Requests (to follow) to the Engine.
g. Engine sends Items to the Item Pipeline and Requests to the Scheduler
h. Repeat from b.

The interface between Scrapy and database is pipeline. The spiders save the crawled news data into items and pipeline will send all items into the database.

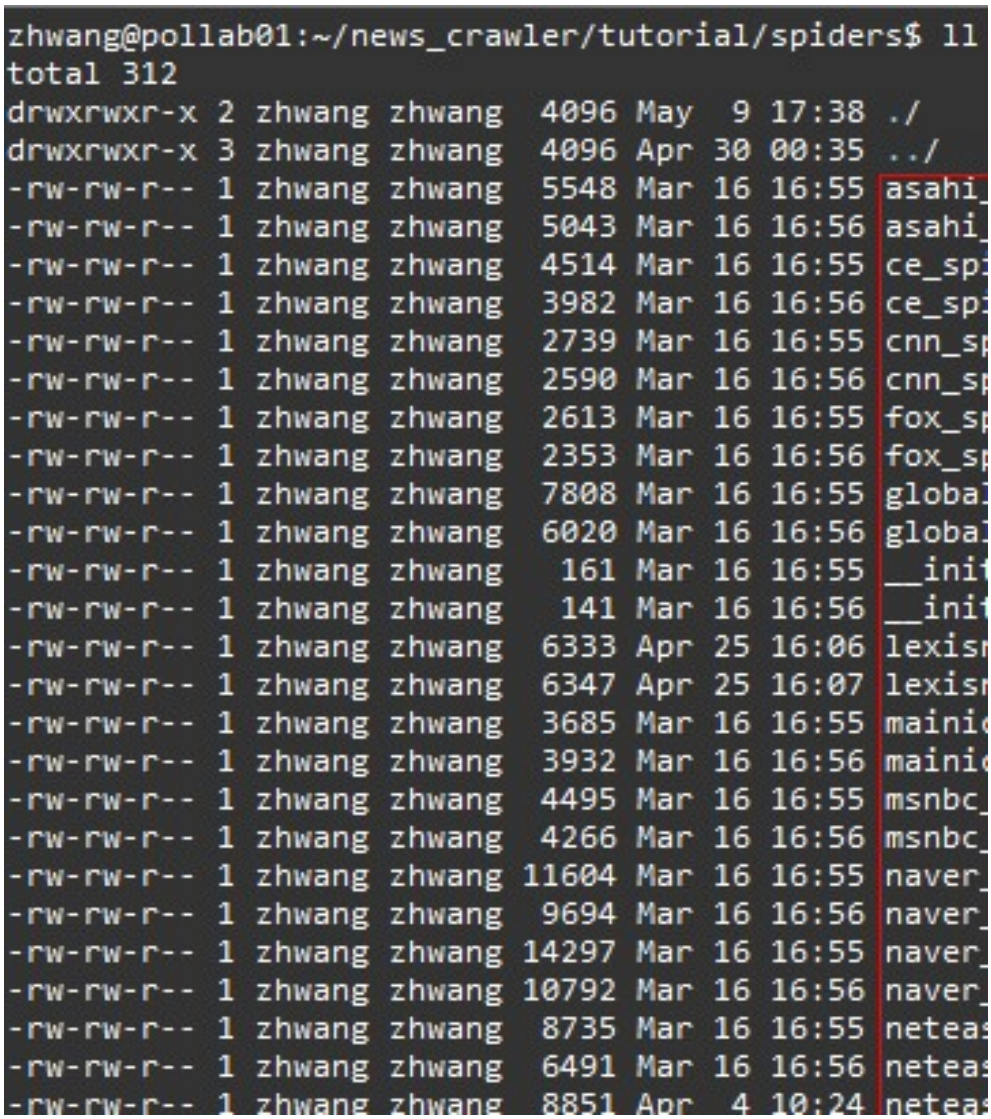

Figure 3.3 Spider List

Figure 3.3 shows the spiders developed in this project. Files with name that ends with ‘.py’ are python file and those that ends with ‘.pyc’ are compiled python files. It can be observed that each spider is only for one website because each website has different structure.

3.3.2 JSON page crawling

Considering the efficiency of crawling we usually try to find if the websites stores the news data in JSON(JavaScript Object Notation); if so, the news data in JSON can be easily handled and extract the wanted data. Fig3.4 shows how to get the JSON page for news data.

Figure 3.4 Using browser inspector mode to get JSON page

We have a target site ‘news.163.com/latest’ which shows the latest news and it refreshes every 60 seconds. It can be conjectured that it brings the latest news data from somewhere. To check if the website using a JSON page is communicating with a certain source page, the browser inspect mode is often used to see if any js (java script) runs on it. So if the inspector is opened and manually refreshed, a list running js can be acquired. Fortunately, a js named ‘news_json.js?’ is found which stores the required news and it directs to ‘http://news.163.com/special/0001220Onews_json.js?’.

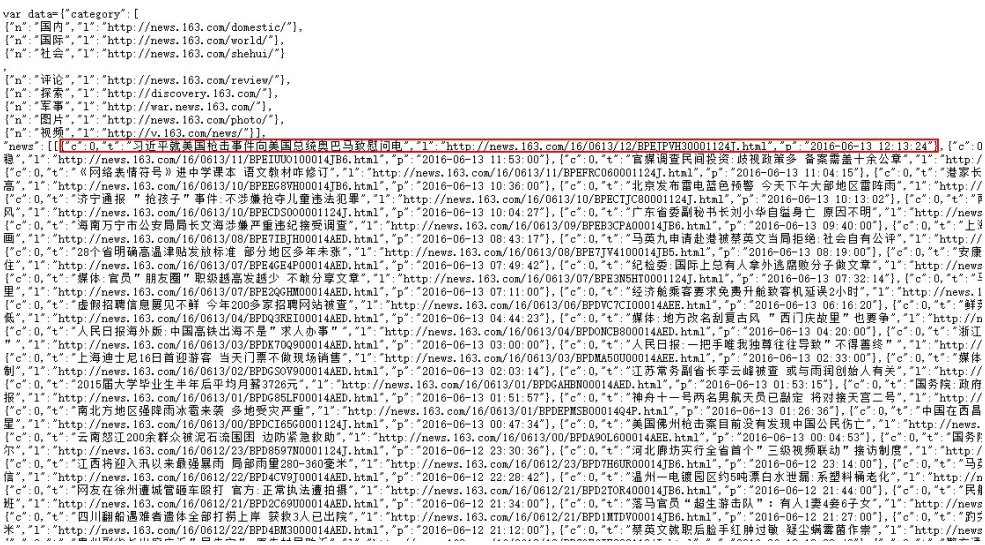

Figure 3.5 News JSON page

Figure 3.5 shows the contents of this JSON page. It is a data set of news information which contains category, title, date and url. Our spider will acquire all the information once and go to each url to acquire the news contents.

3.3.3 Tailoring of the target field

If a site does not have a JSON page or cannot be found, the wanted information can be acquired from specific page source. Fig. 3.6 shows a Naver news main page.

Figure 3.6 Naver news page

To get the article title, the first step is to locate where it is. The inspection tool will can help find the location. Figure 3.7 shows the located article title. If it is not clear enough, the xpath address can be obtained manually, which is '//*[@id="articleTitle"]' in this case, by right clicking the mouse. In the same way, each part of a news article should be located.

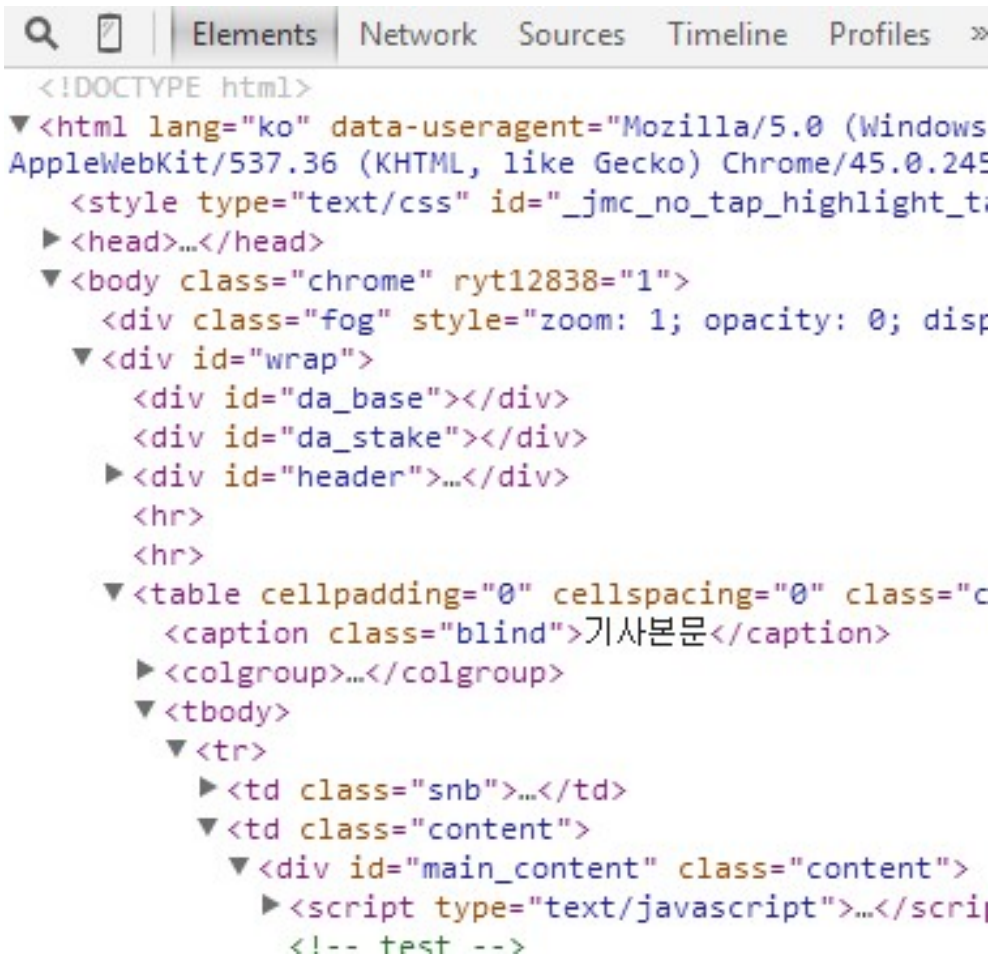

Figure 3.7 Title location

3.3.4 Login issues

Some online news website such as Nikkei only provide service to subscriber so login as a premium user is required to get news data. First, we need to find out what information will be submitted when we login. In order to intercept them an arbitrary ID and password is given to login and the form data in Figure3.8 was captured (Using network function of browser inspector).

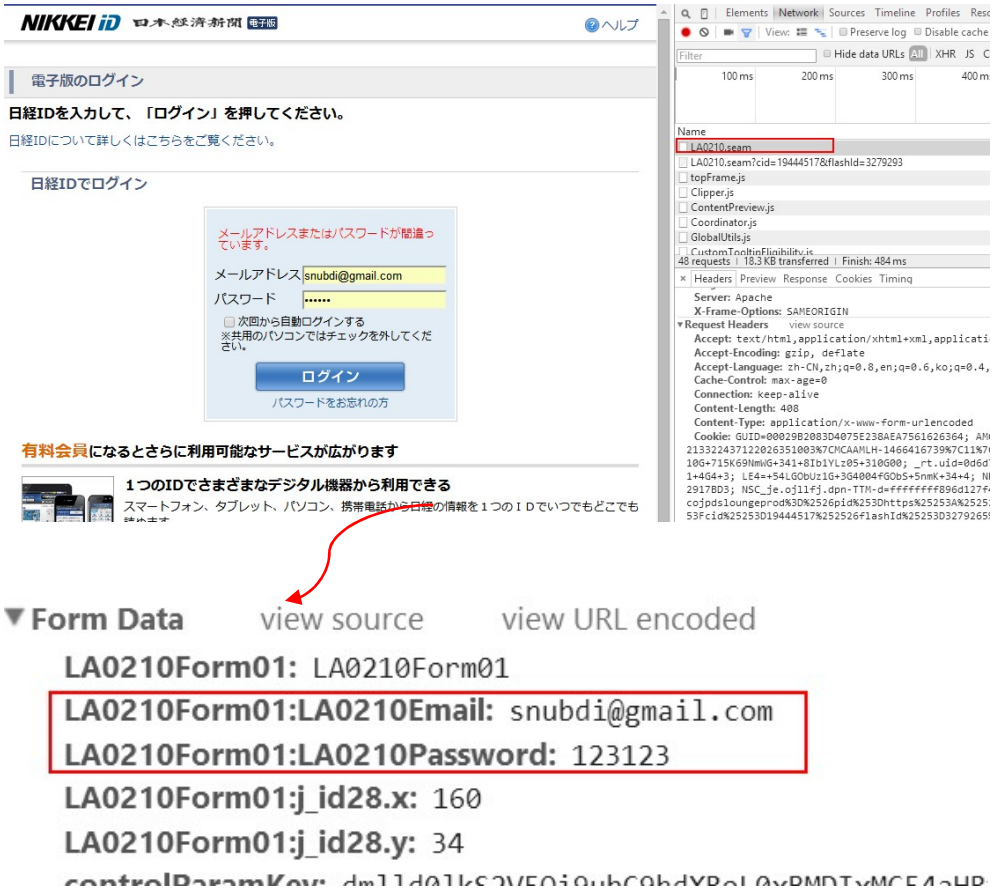

Figure 3.8 Login form data of Nikkei

From the captured form data, it can be seen that the Email and Password is the ID and password, theoretically if we create a form with a correct ID and password and other information exactly like the captured form data, we will successfully login. However, the controlParamKey value appears to be an encrypted value and usually is an access token which changes every time. So we tried to find the value in the page source and it turns out it hides in the login page.

Figure 3.9 Hidden access token

Figure 3.9 shows its location, we first extract it and write it into the form data when we try using Scrapy to login. That is how we solved login issue.

3.3.5 Disarming customized anti-bot

Selenium is a browser automation tool which is used for automating web applications for testing purposes. To use selenium, we simply import selenium library into our python file. In this project, selenium is used to open a browser to help Scrapy to do news crawling. Here are the conditions required to use selenium.

1. Real-time changed target information

Some information such as numbers of comments change in real-time; when a page is opened, the function which can get the numbers of comments runs and return the value. If the scrapy downloader just download the page source, it will be empty. The idea is after selenium opens a browser, it waits for the function to return a value before the Scrapy download the page source.

2. Imitate the click operation

Some sites need a click operation to see all information such as a button ‘see more’. Only when it is clicked, the full news can be loaded. Then scrapy will download the page source.

3. Popup

When we tried to crawl lexisnexis, as a keyword is given to search for related news, a popup will jump out alerting there are more than 3000 result. Only when ‘Retrieve Results’ button is clicked, the news list page can be accessed. The issue is news list page link always changes so selenium is needed to click the button to get the link and scrapy will do the rest.

3.4 News data

There are a lot of ways to crawl internet news. In this project we only use the Scrapy framework to do the crawling and we use MySQL to store news data. The table structure and description is shown in Table 3.3.

Table 3.3 Universal database table structure

Column	Description
Adi	Article id
Date	News date
Title	News title
Contents	News contents
Agency	Media, newspaper company
Url	News link
Category	Policy, Sports, Entertainment etc.
Tagged_text	Contents with Pos-tag
Keywords	Extracted keywords from contents

We usually use the aid (article id) as primary key to avoid duplicate value because for some website we try to crawl it several time a day and our crawling logic does not check the duplicate data.

The accumulated data overview is shown in Table 3.4.

Table 3.4 Crawled news data

Media	**Amount**	**Country**
Chosun Joongang Hankyoreh Donga Kyunghyang	580,000	Korea
Netease China economy People Globaltimes etc.	500,000	China
Niki Asahi Yomiuri Sankei Yahoo	260,000	Japan
CNN FOX NEWS NBC NEWS LexisNexis DB	60,000	USA

By April 30th the statistical result of news data is shown before and the data is increasing every day.

3.5 Keywords extraction and evaluation

The main question is how to use the data to find something useful. To do this, according to [11] and [17], we split the text we crawled and score each word with a graph-based keyword extraction method by the quotient of degree and frequency, where degree means how many times the specific word connected with other words and frequency means how many times the specific word appeared. With this formula, every word in texts are scored, and the key word and the words that are related to are determined.

If a related word is used as a key word and repeat the process, its related word will be found.

4. Result and discussion

So far, the keyword extraction method was introduced. Here, the system performance is tested by giving some real input data. First, a start date and an end date must be set. In this experiment, the start date is set to 2014.07.01, and the end date is set to 2014.07.07. News data from 2014.07.01 to 2014.07.07 will serve as the input data.

If we choose “시_주석” as topic keyword and want to see its related words, we set the number of related words be 20. Then the result of dendrogram and the evaluation of each word will be like in Figure 4.1.

Figure 4.1 Dendrogram of the keyword” 시_주석” and evaluation of related words

If we set the input news data to the entire year of 2014 with the news category is politic and society, it is reasonable that “박근혜_대통령” always take a place. But we will get “침물_사고” and “실종자_가족” appeared during 04.01~06.30 that is because the sinking of MV Sewol which happened in

2014.04.16 and our system successfully extracted the relevant keywords.

This entire system is not only for crawling and analysis news data, but it can be used for variety of applications.

5. Future work

Our spiders are crawling internet data every day and we are trying to cover the news data before the system was made.

We crawled Korean news, Chinese news, English news and Japanese news but until now only Korean NLP is available, we are still working on the Chinese, English and Japanese NLP.

We are also planning to use this system to compare Pubmed with KoreaMed and Naver when we give a medical related keyword so that we will get domestic and international trends and how Naver is inappropriate for search of the medical knowledge due to severe amount of advertisement.

Reference

[1] Prakash M Nadkarni, Lucila Ohno-Machado, Wendy W Chapman, “Natural language processing: an introduction”, J Am Med Inform Assoc. 2011.

[2] Manning C, Schuetze H. Foundations of Statistical Natural Language Processing. Cambridge, MA: MIT Press, 1999.

[3] https://en.wikipedia.org/wiki/Part-of-speech_tagging

[4] N. Bansal and N. Koudas. Blogscope: A system for online analysis of high volume text streams. In WebDb, 2007.

[5] J. Leskovec, L. Backstrom, and J. M. Kleinberg. Meme-tracking and the dynamics of the news cycle. In KDD, 2009.

[6] Konchady, M. (2006). *Text Mining Application Programming.* Charles River Media.

[7] Signorini, A., Segre, A., and Polgreen, P. (2011). The use of twitter to track levels of disease activity and public concern in the u.s. during the influenza a h1n1 pandemic. PLoS ONE,6:e19467.

[8] Mathioudakis, M. and Koudas, N. (2010). TwitterMonitor: Trend detection over the Twitter stream. In Proceedings of the 2010 ACM SIGMOD International Conference on Management of data, pages 1155–1158, New York, NY, USA.

[9] Diao, Q., Jiang, J., Zhu, F., and Lim, E. (2012). Finding bursty topics from microblogs. In Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL HLT 2012), pages 536–544, Jeju Island, Korea.

[10] Allan, J. (2002). Introduction to topic detection and tracking. In Allan, J., editor, Topic Detection and Tracking: Event-based Information Organization, pages 1–16. Kluwer.

[11] Youngsam Kim, Munhyong Kim, Andrew Cattle, Julia Otmakhova, Suzi Park, and Hyopil Shin (2013), Applying Graph-based Keyword Extraction to Document Retrieval, IJCNLP 2013.

[12] Culotta, A. (2010). Towards detecting influenza epidemics by analyzing Twitter messages. In Proceedings of the First Workshop on Social Media Analytics,

pages 115–122, New York, NY, USA.

[13] Fang Chen, Kesong Han and Guilin Chen (2008), “An approach to sentence selection based text summarization”, Proceedings of IEEE TENCON02, 489- 493.

[14] Guihua Wen, Gan Chen, and Lijun Jiang (2006), “Performing Text Categorization on Manifold”, 2006 IEEE International Conference on Systems, Man, and Cybernetics, Taipei,Taiwan, IEEE, 3872-3877.

[15] Pak Chung Wong, Paul Whitney and Jim Thomas,“Visualizing Association Rules for Text Mining”, International Conference, Pacific Northwest National Laboratory, USA, 1-5.

[16] http://doc.scrapy.org/en/1.1/

[17] Wang, Jinghua, Liu, Jianyi, & Wang, Cong. (2007). Keyword extraction based on pagerank. Advances in Knowledge Discovery and Data Mining, 857-864.

[18] http://onlinelibrary.wiley.com/doi/10.1002/aris.1440370103/full

초 록

인터넷 뉴스는 현대사회에서 정보를 얻기 위한 중요한 매체로써 자리 잡고 있다. 인터넷 매체의 부흥과 더불어 컴퓨터뿐만 아니라 테블릿, 스마트폰 등의 보급률이 높아지면서 더욱 많은 사람들이 인터넷 기사를 접하게 되었다. 인터넷 뉴스는 계속적으로 상호 매체간의 재생산이 이루어지기 때문에 기사의 수 증가는 기하급수적으로 늘어난다. 때문에 과거의 정보 부족 시대와는 다르게 무수히 많은 정보를 모두 습득할 수 없어서 핵심적인 정보만 빠르게 얻어내는 능력이 매우 중요해졌다. 이 논문에서는 실시간으로 모든 인터넷 기사를 얻어내고 그로부터 핵심 키워드와 키워드 간의 관계를 알려주는 실시간 뉴스 모니터링 시스템을 제안하고자 한다.

주요어 : 뉴스모니터링, 자연언어처리, 추세분석, 인터넷 뉴스
학 번 : 2014-25150