Implementation of focused crawler by using machine learning approach

Yamin Shwe Ye, Khin Mar Soe
University of Computer Studies, Yangon
yaminshweye@gmail.com,kmsucsy@gmail.com

Abstract

The rapid growth of web generated on the internet by millions of users poses many challenges for general purpose search engines (for example, scaling).Typically a general purpose search engine consists of three main parts, Crawler, Indexer and Query processing system. The crawlers of a general purpose search engine crawl every page. So problems arise when we need to retrieve only corresponding portion of the web, especially for a topic or a group of topic. Such requirement can be fulfilled by a domain specific crawler or focused crawler. Focused crawler crawls only those pages that are interested by the system. A focused crawler traverses the web selecting out relevant pages to a predefined topic and neglecting those out of concern. The focused crawler determines which portion of the web is relevant and which is not. That can be done by several machine learning approach used in text categorization. This thesis proposes a focused crawler by using neural network. It can be used to build general purpose domain specific search engine.
Keywords: Focused Crawling Approach, Search Engine, Machine Learning

1. Introduction

The World Wide Web is a rapidly growing and changing information sources. The growth of internet makes the task of finding relevant information difficult. Search engine attempts to exhaustively crawls the every pages the encountered during crawling. This pose a scalability problem for a general purpose search engine because there are millions of web contents created by users over the world.

Web crawlers are programs that exploit the graph structure of the web to move from page to page, they are also called wanderers, robots, spiders, fish and worms [3]. Focused Crawlers are crawlers that try to find high-quality information on a specific subject as soon as possible and try to avoid irrelevant pages in order to the results would be as accurate as possible [6]. Focused crawler mainly differs from general purpose crawlers in three ways, first they can

lower the network bandwidth required by the crawler because they don't need to crawl every page they visit, second, search results can be accurate because the crawler does not crawl irrelevant pages, third, domain specific search can be done with focused crawler for example for competitive market research [12].

Many machine learning techniques are incorporated for building topical crawler, such as SVM, neural network etc [2] and Latent Semantic Indexing [7]. Many approaches for text categorization can be used for building a topical guided classifier.

The remainder of the paper is organized as follows. Section 2 describes the related work of the paper. Section 3 presents Focused Crawler and Machine Learning Technique, and explains neural network, and back-propagation algorithm. System implementation is described in section 4 and experimental result is presented in section 5.

2. Related Work

There have been many reports on literature on focused crawling.

Focused crawler can be build using link analysis approached where link structure of the pages are analyzed to determine which link should be followed, another technique is used to determine content similarity between pages. [8] presents methods using both content and link structure analysis. In link structure based method famous algorithm like PageRank [9] and HITS [10] are used.

[12] used domain ontology to crawl for specific domain. Latent semantic indexing (LSI) is a concept-based automatic indexing method that models the semantics of the domain in order to suggest additional relevant keywords and to reveal the hidden concepts of a given corpus while eliminating high order noise. [7] Use latent semantic indexing method to classify the relevant page for crawling. [3] presents the details for topical crawler and evaluation method.

In [4] authors presents three approaches; Naïve Bayes, Neural Network and Support Vector Machine for building classifier for focused crawler. The crawlers used in their system are modeled as parallel best first crawler.

[1] presents their ALVIS semantic search engine; they used ontology for their classifier in crawling. [11] use a Support Vector Machine as classifier for the focused crawler. [5] use evolutionary approaches for their focused crawling agents.

3. Focused Crawler and Machine Learning Technique

One of the main components of a search engine is a web crawler which downloads the web page from the internet. **A web crawler** (also known as a **web spider**, **web robot**) is a program or automated script which browses the World Wide Web in a methodical, automated manner. Web crawlers are mainly used to create a copy of all the visited pages for later processing by a search engine that will index the downloaded pages to provide fast searches. Also, crawlers can be used to gather specific types of information from Web pages, such as harvesting e-mail addresses (usually for spam).

In general, it starts with a list of URLs to visit, called the **seeds**. As the crawler visits these URLs, it identifies all the hyperlinks in the page and adds them to the list of URLs to visit, called the **crawl frontier**. URLs from the frontier are recursively visited according to a set of policies.

A focused crawler or topical crawler is a Web crawler aiming to search and retrieve web page from the World Wide Web that is related to specific topic. Rather than collecting and indexing all accessible Web documents, a focused crawler analyzes its crawling boundary to be mostly relevant for the crawling so that irrelevant regions of the Web can be skipped. This leads to significant saving in computing resources such as network bandwidth and processor time.

A simple crawler consists of URL queue, Link extraction module and downloader, URL queue is used for storing URL that should be visited by the crawler, the link extractor module is used for extraction of hyperlink from the web documents, the task of link extractor can include URL cleaning, downloader download the pages from the URL queue and stored the downloaded page in the file system. A focused crawler works as the same way as simple crawler whereas focused crawler used classifier in determining which URL should be stored in their URL queue. One of the most important parts of the focused crawler is classifier; classifier determines which URL should be store in the URL by using some measurement upon context graph, link structure or link content. Many approaches can be used for building a classifier.

Neural network are on the sophisticated machine learning technique used for text categorization, which can also be applied to Focused crawling. Neural Network is an interconnected assembly of simple processing elements, unit or nodes, whose functionality is loosely based on the animal neuron. On the best known method for learning neural network is back-propagation algorithm. Neural network must be trained before they can be used as classifier for focused crawler.

3.1. Neural Network

An artificial neural network (ANN) is often, just called a "Neural Network" is an interconnected group of artificial neurons that use a mathematical model or computation model for information processing based on connectionist approach to computation. Neural Network consists of (possibly larges) number or simple neuron like processing units, organized in a layer. Neural Network has one or more hidden layer of sigmoid neurons followed by output layer linear neurons. Every unit in a layer is connected with all the units in the previous layer. These connections are not equal; each connection may have a different strength or weight. The weights on these connections encode the knowledge of the network. Knowledge is represented in a Neural Network by the pattern of connection among the processing elements and by the adjustable weights of these connections. Neural Network is motivated by their capability for their learning. The operation of a processing element depends on the number of inputs to it that are currently activated and on their weights. The weights or strengths of the links between the neuron are where the functionality of the network resides.

3.2. Back-propagation

Back-propagation has two phases: namely, forward pass phase and backward pass phase. The forward phase computes "functional signal", feed forward propagation of input pattern signals through network. The backward pass phase computes "error signal" propagate the error backward through network starting at output units where the error is the difference between actual and desired output values. Back propagation learns by iteratively processing a set of training samples, comparing the network's prediction for each sample with the actual known class label. These modifications are made in the backward direction, that is, from the output layer, through each hidden layer down to the first hidden layer. Back-propagation algorithm is given below.

Input: The training samples, ***samples;*** the learning rate, ***l;***
Output: A neural network trained to classify the samples.
Method:
(1) Initialize all weights and biases in network;
(2) While terminating condition is not satisfied {
(3) for each training sample ***x in samples {***
(4) // Propagate the *inputs* forward :
(5) for each hidden or output layer unit ***j {***

(6) ***Ij= Σi Wij Oj + θj ;*** // compute the net input of unit j with respect to the layer, i
(7) ***Oj =1/(1+e^{-IJ}) ; } // compute the*** output of each unit ***j***
(8) // Backpropagate the errors:
(9) for each unit ***j in the output layer***
(10) ***Errj = Oj (1 – Oj)(Tj – Oj) ; //*** compute the error
(11) for each unit ***j*** in the hidden layers, from the last to the hidden layer
(12) ***Errj = Oj (1 – Oj)Σj Errj Wjk ; //*** compute the error withe respect to the next higher layer, ***k***
(13) for each weight ***Wij*** in network {
(14) ***ΔWij = (l) Errj Oi ; //*** weight increment
(15) ***Wij = Wij + ΔWij ;*** } // weighr update
(16) for each bias ***θj*** in network
(17) ***Δθj = (l) Errj ; //*** bias increment
(18) ***θj = θj + Δθj ; } //*** bias update
(19) }}

4. System Implementation

4.1. Training neural network

Before neural network can be used to recall, it must be trained. A set of relevant and irrelevant example pages are given to the training process for a specific domain eg, computer science web page and web pages those are not concerned with computer science. Those examples pages are cleaned. Cleaning examples involves removing HTML tags, CSS and JavaScript elements so that each word or feature in the example page can be extracted. Each word in the example pages is tokenized and stored in the database, their frequency and other needed information to calculate Mutual Information is also recorded. Feature for each of the category is selected using Mutual Information. Those selected feature will be fed into the neural network. The network is constructed with three layered network. Input nodes correspond to features selected. The output nodes correspond to relevant class or not. Back propagation algorithm is used to train the network. Mutual Information is used for feature selection. Figure.1. shows training used by focused crawler.

Neural network used by this focused crawler is a three layer feed-forward network. User can specify how many hidden node they want to construct and other parameters for neural network such as learning rate. Figure.2. shows the structure of the neural network used by the system.

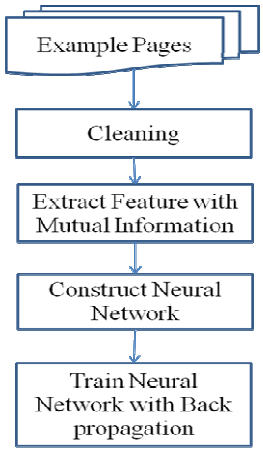

Figure.1. Training

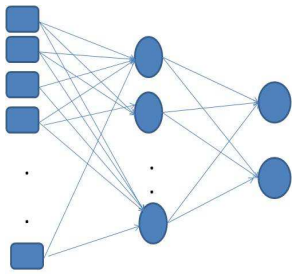

Figure.2. Structure of the neural network used by the system

4.2. Focused Crawler

The flow diagram of the system is presented in figure.3.

User must specify and input seed URL into the system. Seed URL is used to initialize the URL queue. URL queue holds the URLS that should be visited by the focused crawler. Focused Crawler fetches a URL from the URL queue, and then it download the web page from that URL and that work is done by the downloader. Downloader is used to download the web page from internet; it used Java libraries to download the web page. After the web page is downloaded into the system, the page is preprocessed. Preprocessing is used to clean the web page to extract the hyperlinks from the web page. Each downloaded pages is cleaned, all of the hyperlink and their context is extracted. And the extracted context is used as feature input into the neural network. The hyperlink context is feed into the neural network to determine if the link is concerned with the relevant class or not, if the link is relevant, it is put into the URL queue and the crawler download the pages and repeated the entire cycle. All of the downloaded pages are stored in a user specified folder. Downloaded pages from the focused

crawler can be further processed such as search engine indexer.

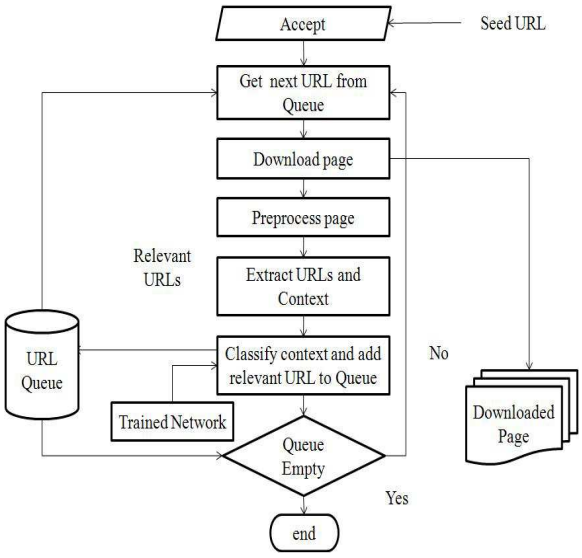

Figure.3. Flow diagram of the System

Screenshot for training neural network is presented in the Figure.4. The number of hidden node, learning rate, mean square error, and the name of the neural network must be given as input to the system. The system used 50 hidden nodes, learning rate is 0.1 and mean square error is 0.001.And then, Network Name is given by any name. Example name is network1.

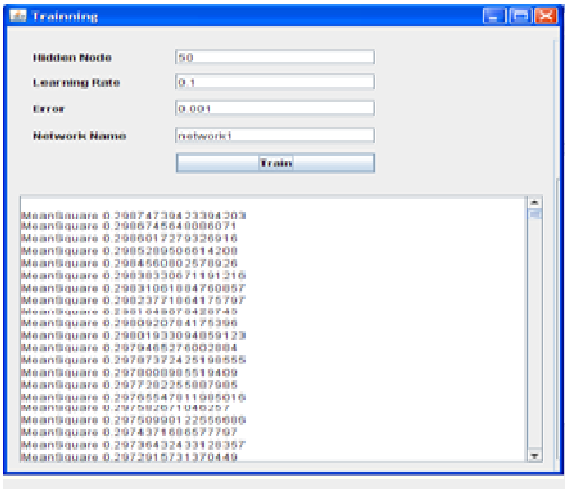

Figure.4. Training neural network

Focused crawler downloading web page is shown in Figure.5. Seed URLs must be given as input to the system. Seed URLs are URLs of computer science sites so that the crawler can exploits other links from seed URLs. The user must also specify the location of the folder to store the downloaded page. The focused crawler will crawl the URLs, download the pages and stored in the user specified folder. All of the pages will be in computer science domain. The text boxes in Figure.5.will display the hyperlink that is downloaded by the system with associate label, positive or negative.

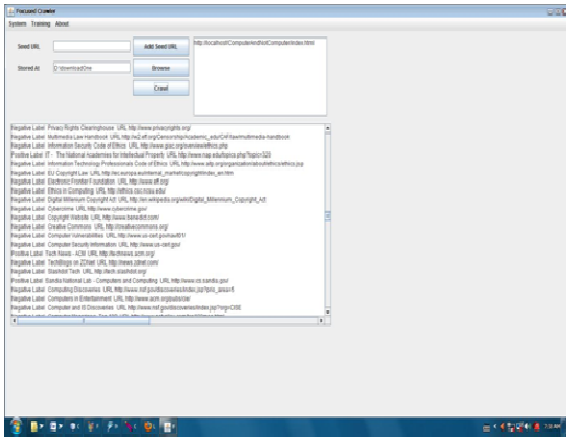

Figure.5. Focused Crawler

5. Experimental Result

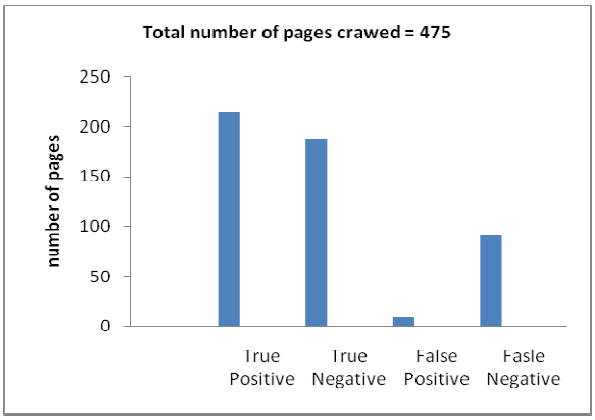

Figure.6. Experimental result

1. **True Negative:** case was negative and predicted negative
2. **True Positive:** case was positive and predicted positive
3. **False Negative:** case was positive but predicted negative
4. **False Positive:** case was negative but predicted positive

Figure.6. shows the experimental result representing true negative, true positive, false negative and false positive for downloaded page. True Positive (TP) and False Negative (FN) are relevant. False Positive (FP) and True Negative (TN) are irrelevant. The total number of page crawled for

experimentation is 475, true negative is 188, true positive is 214, false negative is 92 and false positive is 9. The Precision for the crawler is 0.96, recall is 0.70 and accuracy is 85%. Precision of the focused crawler is high and it can reduce the amount of network bandwidth which will require for downloading the web pages.

Table.1. shows number of relevant and irrelevant pages.

Table.1. Relevant and irrelevant pages

Total number of pages	Relevant	Irrelevant
	214(TP)	9(FP)
475	92(FN)	188(TN)

Table.2. shows the precision, recall and accuracy for table.1.

Table.2. Precision and recall

Precision	Recall	Accuracy
0.96	0.70	0.85

6. Conclusion

This paper presented a focused crawler that can be used to build domain specific search engine or vertical search engine. Feed Forward Neural Network with back propagation is used to filter relevant or irrelevant URL to determine which URL should be downloaded. This system can be used for automatic digital library construction. User can input any kinds of example set for their desired domain to create a crawler for that domain. By using machine learning approach for building domain specific search engine, one needs not to write crawler for each domain.

7. References

[1] Anders Ardö, “Focused crawling in the ALVIS semantic search engine”, 2nd European Semantic Web Conference (ESWC 2005), Heraklion, Greece, 29. May - 1. June 2005.

[2] Filippo Menczer, Gautam Pant, Padmini Srinivasan, "Topical Web Crawler:Evaluating Adaptive Algorithms", ACM Transactionson Internet Technology,Vol.4,No.4,November2004,Pages378–419.

[3] G. Pant, P. Srinivasan, and F. Menczer, ” Crawling the web”, Web Dynamics,2004.

[4] Gautam Pant, “Learning to Crawl: Classifier Guided Topical Crawler”, Ph.D Thesis, University of Iowa, July 2004.

[5] Gautam Pant, Filippo Menczer, "MySpiders: Evolve Your Own Intelligent Web Crawlers", Proc of Autonomous Agents and Multi-Agent Systems,Nerthlands, 5,221–229,2002.

[6] George Almpanidis, Constantine Kotropoulos, and Ioannis Pitas, “Focused Crawling Algorithm Survey and new Aprroach with a manual analysis”, Ph.D Thesis,Lund University, 2008.

[7] George Almpanidis, Constantine Kotropoulos and Ioannis Pitas, "Focused Crawling Using Latent Semantic Indexing - An Application for Vertical Search", 9th European Conference on Research and Advanced Technology for Digital Libraries (ECDL 2005) September 18-23, 2005 Vienna, Austria.

[8] Mohsen Jamali, Hassan Sayyadi, Babak Bagheri Hariri and Hassan Abolhassani, “A Method for Focused Crawling Using Combination of Link Structure and Content Similarity” , ACM International Conference on Web Intelligence (WI 2006), 18-22 December 2006, Hong Kong, China 2006.

[9] S. Bri, L. Page, “The anatomy of large-scale hypertext Web search engine”, Proc of World-Wide Web Conference, Brisbane, Australia, 1998, 107-117.

[10] Jon M. Kleinberg, “Authoritative Sources in a Hyperlinked Environment”, Journal of the ACM, 1999, 46(5), 604-632.

[11] Tesi di Laurea, “An information guided spidering: a domain specific case study”, Ph.D Thesis, Processi decisionalie gestionedella conoscenza, 2008.

[12] Wei Huang, Liyi Zhang, Jidong Zhang, Mingzhu Zhu, "Semantic Focused Crawling For Retrieving E-Commerence Information", Journal of Software, Vol 4, No 5 (2009), 436-443, Jul 2009.