The 8th International Conference on e-Business (iNCEB2009)
October 28th-30th, 2009

ELIMINATING MULTIPLE NOISE PATTERNS IN WEB PAGES FOR WEB DATA EXTRACTION

Thanda Htwe
University of Computer Studies, Yangon
tdhtwe80@gmail.com

Abstract

With the exponentially growing amount of information available on the Internet, an effective technique for users to discern the useful information from the unnecessary information is urgently required. Eliminating multiple noise patterns in web pages for web data extraction becomes critical for improving performance of information retrieval and information extraction. So, we investigate to remove various noisy data patterns in Web pages instead of extracting relevant content from Web pages to get main content information. In this paper, we propose an approach that detect multiple noise patterns and remove these noise patterns from Web pages of any Web sites. Our approach is based on the basic idea of Case-Based Reasoning (CBR) to find noise pattern in current Web page by matching similar noise pattern kept in Case-Based. We also apply a back propagation neural network algorithm to classify the stored various noise patterns by matching similar noise data in current Web page. We have implemented our method on several commercial Web sites and News Web sites to evaluate the performance and improvement of our approach. Experiments show that results leads to a more accurate and effectiveness of the approach.

Index Terms- Noise detection, noise elimination, neural network, information extraction

1. INTRODUCTION

Nowadays, a large number of web pages contain useful information is often accompanied by a large amount of noise such as banner advertisements, navigation bars, copyright notices, etc. These noise data can seriously harm for Web miners by extracting whole document rather than the informative content and also retrieve non-relevant results. It is also important to distinguish valuable information from noisy data within a single Web page. As we all know Web pages are constructed not only main contents information like product information in shopping domain, job information in a job domain but also advertisements bar, static content like navigation panels, copyright sections, etc.

When we process Web documents, the main content is surrounded by noise in the retrieved data. Therefore, without removing such data, the efficiency of feature extraction and finally text classification is certainly degraded. Web noise can be classified into global noises and local noise by Yi and Liu [1] [4]. Global noises include mirror sites; legal/illegal duplicated Web pages, old versioned Web page with advertising segments, unnecessary images, or navigation links, etc.

This paper propose a system for eliminating various noise patterns from web pages for purpose of improving the accuracy and efficiency of web content mining. Many studies on information extraction (or information retrieval) also try to discover informative content from a set of Web documents [11]. Extraction of "useful and relevant" content from web pages has many applications, including cell phone and PDA browsing, speech rendering for the visually impaired, and text summarization. Most approaches to removing clutter or making content more readable involve changing font size or removing HTML and data components such as images, which takes away from a webpage's inherent look and feel. However, it is relatively little work has been done on eliminating noisy data from Web pages in the past. Hence, we mainly focus on efficiently and automatically detecting and removing noisy data from Web pages to extract only relevant information.

Web pages are often cluttered with distracting features around the body of an article that distract the user from the actual content they're interested in. These "features" may include pop-up advertisement, flashy banner advertisements, search and filtering panel, unnecessary images, or links scattered around the screen. However, these noisy data formed in various patterns in different Web sites. When we extract only relevant information, such items are irrelevant and should be removed.

Therefore, we propose the mechanisms in this paper to eliminate multiple noise patterns in Web pages to reduce irrelevant and redundancy data. We apply Case-Based Reasoning technique to detect multiple noise patterns in current Web page and also present back propagation neural network algorithm for matching current noise with storing noise patterns for noise classification. And then we remove this noise pattern in current page for content extraction. The utilization of a neural network in the detection of instance of noise pattern would be the flexibility that the network would provide. A neural network would be capable of analyzing the data from the network, even if the data is incomplete or distorted. A neural network might be trained to recognize known

suspicious events with a high degree of accuracy.

In the following section of the paper, we first describe related studies in Section 2. Then, in Section 3, we illustrate the representation of noisy data in a page and present our proposed system NoiseEliminator to detect, classify and eliminate multiple noise patterns for extracting main content information. Section 4 describes the results of our approaches and finally, the paper concludes with the conclusion and future work in Section 5.

2. RELATED WORK

Many Researchers have developed several approaches for retrieving and extracting main content from Web pages. Most of them have focused on detecting main content blocks in Web pages. Although cleaning noisy data is an important task, relatively little work has been done in this field.

InfoDiscoverer [6] was proposed an approach to discover informative contents from a set of tabular documents of a Web site by dynamically select the entropy threshold. The system first partitioned a page into several content blocks according to HTML tag <TABLE> in a Web page. The system is not applicable general Web pages which is consisted using tag <DIV>. [7] proposed a redundant information elimination approach in the Web documents from the same URL path. Three filtering methods, tag based filtering, redundant words filtering and redundant phrase filtering, are used in their system. A redundant word/phrase filtering method is used for single or multiple tokenizations.

An approach that allows for fully-automated extraction of content based on distilling linguistic and structural features from text blocks in HTML News Pages is proposed in [8]. In this approach, content extraction is applicable to any type of news pages using thresholds learned by the Particle Swarm Optimizer. However, human effort is required to label documents for classification. Song et. al [9] investigated mechanisms for segmenting Web pages by assigning importance values to blocks.

The next proposals [13] for content extraction take into account visual cues of Web page rendering, e.g., the distance of conceptual blocks from the screen's center, the alignment of page segments, and so forth. The systems make use of rendering engines in Web browsers, which translate directly between HTML elements and their positioning within the browsing window.

The approach mentioned in [10] builds site style tree in simplistic manner, which is generalized DOM tree presentation of related pages. Noisy elements in the tree are detected based on entropy calculation over set of features. They identified the common presentation style and content and then compressed them into site style tree. To construct the site style tree, the system needs to learn the whole web site to detect the common presentation style and content. The similar problem is addressed in [11] which based on entropy calculations over set of features, but without using site style tree.

Some Web sites are structured with dynamic Web pages and their content and presentation style are not common. It is difficult to detect different noise patterns for those Web sites by using above technique. These techniques are less successful in identifying noise patterns which vary from expected patterns. This paper, therefore, proposes an effective technique to eliminate multiple noise patterns in Web page for information extraction no need to learn the whole Web sites. Our solution employs multiple extensible techniques that incorporate the advantages of the previous work on content extraction.

3. THE PROPOSED SYSTEM

In this section, we firstly illustrate the two mechanisms that determine which region of current Web page contains noise or mixture (data and noise region). Then, we devise another mechanism on matching to determine how we process the three classes (noise, data and mixture) in case based. Lastly, we remove the various noise patterns in current Web page and show extracted main content data. Figure (1) describes the detail architecture of the proposed approach.

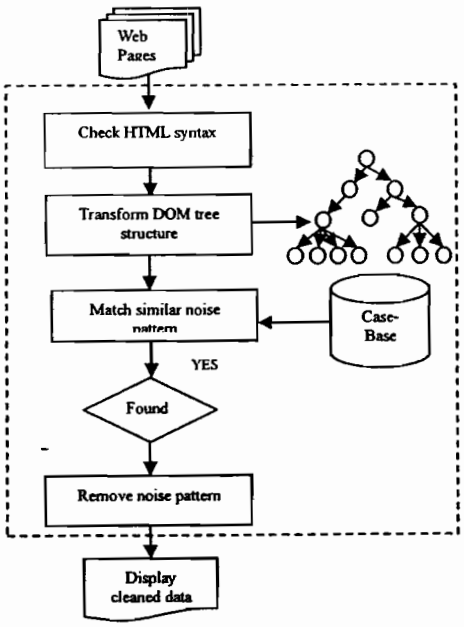

Figure 1. An overall flow of proposed approach

3.1 Types of Noise Pattern

Noise data of Web documents can be categorized into two groups such as global noise and local noise [7]. Global noises are redundant Web pages over the Internet such as mirror sites and legal or illegal duplicated Web pages. Local noises only related intra-page redundancy and exist in the Web page. This paper only focuses on the local noise elimination method. There are at least four different known categories of noise pattern within Web pages of any Web sites including banners with links including search panels, advertisements, navigational panel (directory list) and copy right and privacy notice in each Web site.

We can see that many Web pages contain these four noise categories together but most of noise patterns are structured by using sectioning tags such as <TABLE> and <DIV> and sectioning separating tags like <FRAMESET>, and interactive tags like <SELECT>, <FIELDSET> , <INPUT>. Moreover, anchor tag <A> and <IMG> tag are most commonly used to link another Web page or another Web site. However, these four noise categories can be structured by using various noise patterns (or tags).

3.2 Representation of Noise pattern in Case-Based

The basic idea of our approach is that noise data in Web pages of any Web site are always generated either by using various tags mentioned above. Once we summarize characteristic from sample pages in different Web sites, we can apply it to other similar pages. Thus, we try to borrow the idea of Case-Based Reasoning (CBR) to detect similar noise pattern in current Web page.

CBR is a machine learning methodology that adopts a lazy learning approach and contains no explicit model of the problem domain. Case-based reasoning (CBR) utilizes past experiences as a key data resource for future problem solving and is considered an innovative technique in the development of Artificial Intelligence. CBR uses periodic experiences stored in cases as a basis for decisions and has been implemented in a wide range of fields.

Each case is made up of a description of a past example or experience and its respective solution. The full set of past experiences encapsulated in individual cases is called the case base. The idea is to learn from experience. However, a crucial aspect of CBR lies in the term "similar". When a new problem is presented, the case base is searched, similar past examples are found and these are used to solve the presented problem.

Thus, we identify variety of noise patterns in many Web sites and these noise patterns in each site are represented by DOM (Document Object Model) tree and keep them into database as a case in our first task. DOM trees remain highly editable and can easily be reconstructed back into a complete webpage. The DOM tree is hierarchically arranged and can be analyzed in sections or as a whole, providing a wide range of flexibility for our proposed method. By parsing a webpage into a DOM tree, more control can be achieved while eliminating noise data. Moreover, increasing support for the Document Object Model makes our solution widely portable. We use the Xerces HTML DOM [12]. Some Web pages are not well formed documents, it is necessary to make well formed document before processing them. We first check the syntax of HTML Web page using online HTML tidy tools [14] before parsing Web page into DOM tree structure. Then, a variety of noise patterns are extracted as sub trees of DOM and store them in case-based.

Let $D = \{T_1, T_2, T_3, ..., T_n\}$ be a DOM tree with n sub-trees.

Let $T = \{n_1, n_2, n_3, ..., n_j\}$ be a sub tree of DOM where n_i is a pair (t, d) for i^{th} child node of T and t be tag node and d be data element. In this case, we only use tag nodes. Figure (2) illustrates how noise patterns are stored in case-based. If a new Web page enters, we first check the syntax of HTML Web page using online HTML tidy tools [14] before parsing Web page into DOM tree structure because some Web pages are not well formed documents, it is necessary to make well formed document before processing them. After parsing it, DOM tree structure divided it into several sub trees according to threshold level. Then, we can search similar existing noise pattern in Case-based. When we found the highest similarity value between two patterns, this pattern will classify it as noise and remove it.

The two most widely used techniques of similar case retrieval are Nearest-neighbor retrieval and inductive retrieval. In this case, we applied Artificial Neural Network model for pattern matching. The greatest strength of neural networks is their ability to learn by example.

3.2 Artificial Neural Network Model

Artificial Neural network (ANN) model can be known as a good problem-solving method for problems that can't be solved using conventional algorithms. Neural networks are very good at pattern-recognition and pattern-matching tasks. If the input is one it has never seen before, it produces an output similar to the one associated with the closest matching training input pattern.

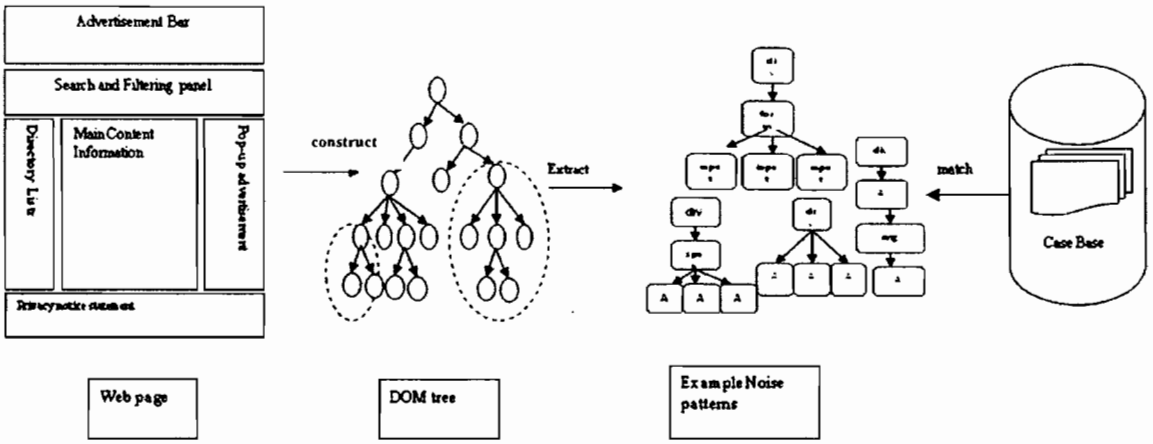

Figure 2. case representation in case-base

In neural network model architecture, each node at input layers receives input values, processes and passes to the next layer. This process is conducted by weight which is the connection strength between two nodes. The key feature of neural networks is that they learn the input/output relationship through training. There are two types of training used in neural networks: supervised and unsupervised training, of which supervised is the most common.

The model structure is trained with known samples of data. It is a learning scheme for updating a node's weights. In this phase, a pattern detected in the data set is presented to the user. It also corresponds to a particular abstract learning task. Nodes apply on iterative process to the number of inputs to adjust the weights of the network in order to optimally predict the sample data on which the training is performed.

In supervised learning, the training data contains examples of inputs together with the corresponding outputs, and the network learns to infer the relationship between the two. The ANN gains the experience initially by training the system to correctly identify pre-selected examples of the problem. The response of the neural network is reviewed and the configuration of the system is refined until the neural network's analysis of the training data reaches a satisfactory level. In addition to the initial training period, the neural network also gains experience over time as it conducts analyses on data related to the problem.

3.3 ANN model for noise classification

To train the model, randomly selected several Web pages used as a data set. The present study is aimed to solve a multi class problem in which not only noise patterns are distinguished from Web page, but also data and mixture patterns are identified. Neural networks process numeric data in a fairly limited range. So, currently enter Web page is parsed into sub-trees according to the threshold level and then converted these sub trees into a standardized numeric representation by using eq(1).

$$x_i = \frac{S_n}{T_n} \qquad \text{eq (1)}$$

where x_i be input nodes at input layer, S_n be the number of occurrence of same leaf nodes in sub-tree and T_n be the total number of leaf nodes in sub-tree.

Here, three classes are described which can be extended to cases with more noise types. An output layer with three neurons output states was used: [1 0 0] for Noise class, [0 1 0] for Data class and [0 0 1] for Mixture (data and noise) class. All the implemented neural networks had seventeen neurons and three output neurons (equal to the number of classes). The number of hidden layers and neurons in each were parameters used for the optimization of the architecture of the neural network.

The widely used learning method, back propagation algorithm is used to train for classifying which case is probable based on current Web page. It has been found that the standard sigmoid activation function is suitable for modeling the occurrence of noise pattern using the pattern matching and learning model. During training process in neural network, we can occur one problem. In an over fitted ANN, the error (number of incorrectly classified patterns) on the training set is driven to a very small value, however, when new data is presented, the error is large. One possible solution for the over-fitting problem is to find the suitable number of training epochs by trial and error.

4. EXPERIMENTS AND RESULTS

In this section, we describe experiments on six different commercial Web sites (www.buy.com,

www.infobanc.com, www.productwiki.com, www.JandR.com, www.amazon.com and www.etsy.com) and three different News Web sites (www.foxnews.com, www.digg.com and www.zdnet.com) as data set. We implemented a three layer MLP (one hidden layer with fourteen neurons) network. One of the objectives of the present study is to evaluate the possibility of achieving the same results with this less complicated neural network structure. In this technique, the available data is divided into three subsets. The first subset is the training set, which is used for training and updating the ANN parameters. The second subset is the validation set. The error on the validation set is monitored during the training process. The third is used for testing. Table 1 shows detailed information about the number of records such as noise, data and mixture for training, validation and testing sets.

Table 1. Number of records used as data set for three subsets

Class	Training Set	Validation Set	Test Set
Noise	200	50	100
Data	50	30	50
Mixture	50	30	50

The training of the neural network was conducted using a backpropagation algorithm for 500 iterations of the selected training data. Figure (3) shows the mean square error of the training process versus the progress of training epochs. Suppose a fixed parameter θ to estimate and estimator of θ denote by β. For a given sample S, the error β is defined as $\beta(S)-\theta$, where $\beta(S)$ is the estimate for sample S, and θ is the parameter being estimated. The bias of β is defined as $B(\beta) = E(\beta) - \theta$. We calculated mean square error as MSE = variance + square of bias. So we got the relation of MSE, variance and bias in eq (2).

$$MSE(\beta) = var(\beta) + (B(\beta))^2 \qquad \text{eq (2)}$$

The error delay decreased to an outstanding level with 0.009491259 MSE (mean-squared error). Therefore, it was expected to have good classification results. The final correct classification rate on known test data is 99.8% for Noise class with 0.001466829 MSE, 98.9% for Data class with 0.021114203 MSE and 98.2% for Mixture class with 0.002909492. However, unseen data (test set) was fed to the neural network, the correct classification rate was less than result was less than 75%. The classification rate on unseen test data is 83.7% for Noise class with 0.011617541 MSE, 77.9% for Data class with 0.014450942 MSE and 65.3% for Mixture class with 0.036937428. We compared the correct percentage rate on known data and unseen data for testing process in Figure (4).

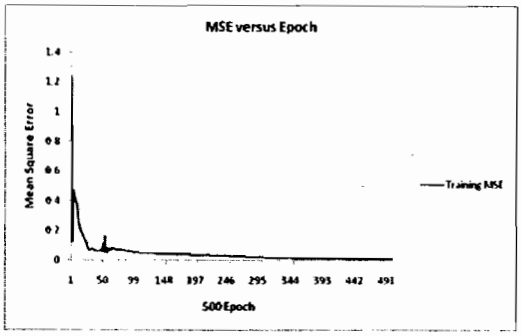

Figure 3. Mean Square Error of the training procedure versus training epochs

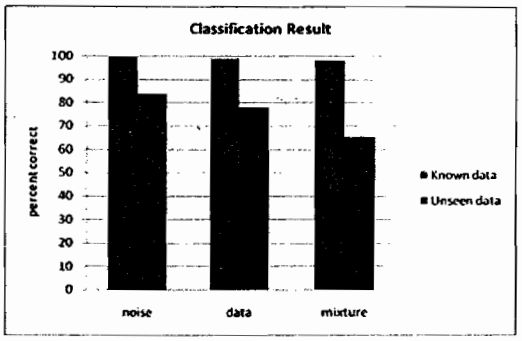

Figure 4. Comparison of correct classification result for known and unknown testing data

5. CONCLUSION

Classifying and removing noise from web pages will improve on accuracy of search results as well as search speed, and may benefit web-page organization applications (e.g., keyword-based search engines and taxonomic web page categorization applications. We can conclude that our proposed methods are feasible to clean noise data from Web pages of any Web site.

The most popular back propagation algorithm of neural network was also successfully used for the classification of correct classes. Benefit to us by learning from past experience to deal with new and unexpected situations. The implemented system solved a three class problem. Noise removing accuracy of this system is depend on the correct classification result of neural network. Therefore, we can eliminate various noise patterns for Web sites (www.productwiki.com, www.digg.com and www.infobanc.com) nearly 80% because these sites are structured with various noise patterns and data patterns seperately. However, the system can eliminate variety of noise patterns with accuracy rate less than 70% for two third of all data set which are structured by mixing noise and data regions. So,

noise removing accuracy degrade for these sites. However, its future development to several classes is straightforward. We plan to extend our approach in diverse directions. We develop the system not only to detect multiple noise patterns in Web page but also to classify the types of noise based on patterns in our future work.

6. REFERENCES

[1]. Yi, L., B. Liu, and X. Li. "Eliminating Noisy Information in Web Pages for Data Mining". In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD-2003)*. 2003. Washington, DC, USA.

[2]. Hsu, C. N. and Dung, M. T., "Generating Finite-state Transducers for Semi-structured Data Extraction from the Web," Information Systems, 23(8):521-538, 1998.

[3]. Kushmerick, N., "Wrapper Induction for Information Extraction," Ph.D. Dissertation, Department of Computer Science and Engineering, University of Washington, 1997.

[4]. Yi, L. and B. Liu. "Web page cleaning for Web mining through feature weighting". In *Proceedings of Eighteenth International Joint Conference on Artificial Intelligence (IJCAI-03)*. 2003.

[5]. S.-H. Lin and J.-M. Ho. "Discovering informative content blocks from web documents". In *KDD*, pages 588-593. ACM, 2002.

[6]. H.-Y. Kao and S.-H. Lin, "Mining Web Informative Structures and Contents Based on Entropy Analysis". In *IEEE*, 2004

[7]. B. H. Kang and Y. S. Kim. "Noise Elimination from the Web documents by using URL paths and Information Redundancy".

[8]. C.-N Ziegler and M. Skubacz. "Content Extraction From New Pages using Particle Swarm Optimization". In *IEEE*, 2007

[9]. R. Song, H. Liu, J.-R, Wen, and W.-Y. Ma. "Learning block importance models for Web pages". In WWW, 2004.

[10]. L. Yi, B. Liu, and X. Li. "Eliminating noisy information in Web pages for data mining". In KDD, 2003.

[11]. H., -Y. Kao, J. –M. Ho, and M. –S. Chen. Wisdom "Web intrapage informative structure mining based on document Object model". *IEEE* Trans KDD, 2005.

[12]. http://www.w3.org/DOM

[13]. Y. Yang and H.-J. Zhang. "HTML page analysis based on visual cues". In *Proceedings of the Sixth International Conference on Document Analysis and Recognition*, pages 859– 864, Washington, DC, USA, 2001. IEEE Computer Society.

[14]. http://infohound.net/tidy