Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Engineering 29 (2012) 537 – 541

Procedia Engineering

www.elsevier.com/locate/procedia

2012 International Workshop on Information and Electronics Engineering (IWIEE)

The Research of Text Mining Based on Self-Organizing Maps

Yi Ding*,Xian Fu

The college of computer science and technology Hubei normal university,Huangshi, 435002,Chinas

Abstract

New methods that are user-friendly and efficient are needed for guidance among the masses of textual information available in the Internet and the World Wide Web. This paper describes text mining has been gaining popularity in the knowledge discovery field, particularity with the increasing availability of digital documents in various languages from all around the world. In this work, we attempt to develop a language-neutral method to tackle the linguistics difficulties in the text mining process. Using a variation of automatic clustering techniques, which apply a neural net approach, namely the Self-Organizing Maps (SOM). The SOM is used to generate two maps, namely the word cluster map and the document cluster map, which reveal the relationships among words and documents respectively. The search process incorporates these two maps and effectively finds the relevant documents according to the keywords specified in the query. The conceptually associated web documents are found not only by the specific keywords but the relevant words found by the word cluster map.

*Keywords:*data mining;text mining, self-organizing maps;knowledge extraction;nformation retrieval

1. Introduction

Web text mining is a new issue in the knowledge discovery research field. It is aimed to help people discover knowledge from large quantities of semi structured or unstructured text in the web. Similar to data mining approaches, web text mining requires a technique that can automatically analyze data and extract relationships or patterns from large collections of web text by using specific algorithms. A retrieval system finds documents that help the users satisfy their information needs. However, the major problem with traditional search methods such as searching by keywords is the difficulty to devise appropriate search expressions to formulate the actual user's information needs. Even with a clear idea of

* Corresponding author. Tel.: +86-013972778052
E-mail address: teacher.dingyi@yahoo.com.cn

doi:10.1016/j.proeng.2011.12.757

the desired information it may be difficult to produce all suitable key terms and search expressions. As a result, a method of encoding the information based on conceptually related word clusters rather than individual words would be helpful. In this work, the self organizing map (SOM)[1,2] is employed as a means for automatically arranging unstructured and high dimensional textual data so that similar inputs are in principle mapped close to each other. Through some training process, the resulting map allows itself readily form word clusters and document clusters, and thus the distance relations between different data items can be beneficial for searching during the text mining process.

2. Related Work

Text mining is a new interdisciplinary field. It combines the disciplines of data mining, information extraction, information retrieval, text categorization, machine learning, and computational linguistics to discover structure, patterns, and knowledge in large textual corpora. Advances in computational resources and new statistical algorithms for text analysis have helped text mining develop as a field. Recently there have been some innovative techniques developed for text mining. Text mining by using self organizing map (SOM) techniques has already gained some attention in knowledge discovery research and the information retrieval field. One paper [3] perhaps marks the first attempt to utilize SOM (unsupervised neural networks) for information retrieval work. In this paper, however, document representation is made from 25 manually selected indexed terms and is thus not very realistic. In addition, among the most influential works we certainly have to mention WEBSOM [4-6]. Their work aims at constructing methods for exploring full-text document collections; the WEBSOM started from Honkela's suggestion of using the self-organizing semantic maps [7] as a preprocessing stage for encoding documents.

3. Developed Approach for Text Mining

3.1. Document Preprocessing

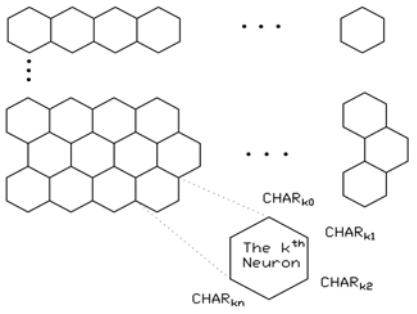
The proposed system focuses on the task of finding associations in collections of text. Based on association, similar documents, through the proposed text mining process, can be gathered in a cluster. For an English book, the document preprocessing is quite straightforward. Our approach begins with a standard practice in information retrieval (IR) to encode documents with vectors, in which each component corresponds to a different word, and the value of the component reflects the frequency of word occurrence in the document. As a result, techniques for controlling the dimensionality of the vector space are required. Such a problem could be solved by eliminating some of the most common and some of the rarest words, and by applying a numerical algorithm such as Latent Semantic Indexing (LSI) method. For a Chinese book, the document preprocessing is relatively complicated. Since a Chinese sentence is composed of characters without boundaries, segmentation is indispensable.

3.2. Self-Organizing Maps

The self-organizing map (SOM) is one of the major unsupervised artificial neural network models. It basically provides a way for cluster analysis by producing a mapping of high dimensional input vectors onto a two dimensional output space while preserving topological relations as faithfully as possible. After appropriate training iterations, the similar input items are grouped spatially close to one another. As such, the resulting map is capable of performing the clustering task in a completely unsupervised fashion. In this work we employ the SOM method to produce two maps for text mining, namely the word cluster map and the document cluster map.

3.3. The Word Cluster Map and Document Cluster Map

The word cluster map that is employed for document encoding is produced according to word similarities, measured by the similarity of the co-occurrence of the words. Conceptually related words tend to fall into the same or neighboring map nodes. By means of the SOM algorithm, word clusters can be ordered and organized as nodes on the map. Let $x_i \in R^N, 1 \le i \le M$,be the feature vector of the *i*th document in the book, where N is the number of indexed terms and M is the number of documents. We used these vectors as the training inputs to the map. The map consists of a regular grid of processing units called neurons. Each neuron in the map has N synapses. Let $CHAR_k = \{CHAR_{kn} | 1 \le n \le N\}, 1 \le k \le K$, be the synaptic weight vector of the *k*th neuron in the map, where K is the number of neurons on the map. Fig.1.(a) depicts the formation of the map. We trained the map by the SOM algorithm:



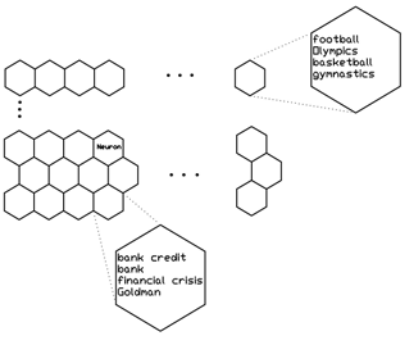


Fig. 1 (a)The formation of neurons in the map. (b) The word cluster map

Step 1. Randomly select a training vector x_i from the book.

Step 2. Find the neuron j with synaptic weights $CHAR_k$ which is closest to x_i

$$\|x_i - x_k\| = \min_j \|x_i - CHAR_j\| \quad (1)$$

Step 3. For every neuron l in the neighbor of node k, update its synaptic weights by

$$CHAR_l^{new} = CHAR_l^{old} + \alpha(t)\left(x_i - CHAR_l^{old}\right) \quad (2)$$

where $\alpha(t)$ is the training gain at time stamp t.Step 4. Increase time stamp t. If t reaches the preset maximum training time T, halt the training process; otherwise decrease $\alpha(t)$ and the neighborhood size, and go to Step 1.

The training process stops after time T which is sufficiently large that every feature vector may be selected as training input for certain times. The training gain and neighborhood size both decrease when t increases. After the training process, the map forms a Word Cluster Map by labeling each neuron with certain words. For the *n*th word in the book we construct an N-dimensional vector v_n in which only the *n*th element is non-zero. To label the neurons, we present each v_n to the map and find the best matching neuron. Fig.1.(b) depicts the schematic drawing of the word cluster map.

4. Discovery Algorithm

In this section we explain how the word cluster map and document cluster map effectively model the relationship between the words and documents. We transform a document to a vector of word occurrence.

After the self-organizing process, two documents will map to near neurons if they contain similar word occurrences. When different words are labeled on the same neuron or near neurons on the word cluster map, they tend to occur in a restricted set of documents. On the other hand, if two words seldom co-occur in any document, they should not be labeled on near neurons. This is because the neuron may be viewed as representing a virtual document containing those words labeled on it. Two words will be mapped to the same neuron if, and only if, they often co-occur in the same document, otherwise the virtual document may not contain these words simultaneously. Neighboring neurons in the word cluster map represent word clusters containing similar words, i.e. words tend to co-occur in the same document. Hence the self-organizing map may measure the word co-occurrence similarity among documents. We define the similarity between the p th word and the q th word as follows:

$$D_1(p,q) = \left(1 + 2^{\left\| G(N_p) - G(N_q) \right\|}\right)^{-1} \quad (3)$$

where N_p is the neuron labeled by the pth word and $G(N_p)$ is the two-dimensional grid location of N_p. Such similarity measures the likelihood of the co-occurrence of words. Large similarity reveals that the two words often co-occur in the same set of documents, which may be considered as a kind of association pattern among the words.

On the document cluster map a neuron represents a document cluster that contains documents of similar meaning, which is defined by the set of highly co-occurring words they contain. Since we train the map using the encoded document feature vectors as input, the weight vector of a neuron represents the occurrence of the words in a virtual document. On the document cluster map only those documents containing overlapping words may map to the same neuron. Documents containing non-overlapping words may map to distant neurons. Neighboring neurons represent document clusters with similar (overlapping) sets of words; thus the co-occurrence of words may be determined by the neighborhood spatially. For any document, we can find its similar documents by examining its mapped neuron and neighboring neurons in the document cluster map. The similarity between the *i*th and *k*th document is defined as follows:

$$D_2(i,k) = \left(1 + 2^{\left\| G(N_i) - G(N_k) \right\|}\right)^{-1} \quad (4)$$

where Ni here is the neuron labeled by the ith document and $G(N_i)$ is grid location of N_i as in (3).

Documents which contain the same sets of words will definitely map to the same neuron, resulting in high similarity defined in (4). Moreover, even if these documents do not contain exactly the same set of words, we may still say that they are conceptually similar because (1) they still contain common words that often co-occur in these documents, and (2) the dissimilar words are likely occurring in documents mapped to the nearby neurons. Document cluster maps provide an effectively way to form document clusters. A neuron on the map represents a document cluster. We can also define the similarity between two clusters by the distance of their corresponding neurons:

$$D_3(j,l) = \left(1 + 2^{\left\| G(N_j) - G(N_l) \right\|}\right)^{-1} \quad (5)$$

where j and l are the neuron indices of the two clusters.

The similarities defined in (3)–(5) provide some knowledge about the documents in the book. We use (3) to discriminate words based on the knowledge discovered from the book. This is also true for (4) and (5) to discriminate documents and clusters respectively. Word associations, as well as document associations, are clearly defined by such similarities. It is natural to apply such associations to applications of document retrieval, indexing, and clustering.

We perform a search through the word cluster map to obtain the labeled neurons of the keyword or combinations of keywords. The documents labeled on the corresponding neurons on the document cluster map are selected and presented to the user. Moreover, all documents may also be presented by document

similarity defined in (4). Documents with higher similarities appear earlier in the query result. The user gets a list of relevant documents even when the query words may not occur in the documents. Important information and new knowledge related to the user's query may be revealed by our method.

5. Conclusions

Web text mining is aimed to help people discover knowledge from large quantities of semi-structured or unstructured text in the web. In this paper, a novel SOM-based method for text mining is presented. The documents were first transformed to a set of feature vectors, in which each component corresponds to a different word and the value of the component reflects the word- occurrence frequency in the document. The vectors were used as input to train the self-organizing map. Two maps, namely the word cluster map and the document cluster map, were obtained by labeling the neurons in the map with words and documents respectively. The search process incorporates the word cluster map and the document cluster map to effectively reveal the relationships among the documents. The resulting documents for a specific query may contain not only the keywords specified in the query but similar words which are often co-occurred in the documents in the book. Relevant documents can thus be easily found through the search process.

References

[1] Dagan, R. Feldman, and H. Hirsh, "Keyword-based browsing and analysis of large document sets," in Proceedings of the Fifth Annual Symposium on Document Analysis and Information Retrieval (SDAIR), Las Vegas, NV, 2004, pp. 191–208.

[2] R. Feldman and I. Dagan, "KDT—knowledge discovery in texts," in Proceedings of the First Annual Conference on Knowledge Discovery and Data Mining (KDD), AAAI Press: Montreal, 2005, pp. 112–117

[3] R. Feldman,W. Klosgen, and A. Zilberstein, "Visualization techniquesto explore data mining results for document collections,"in Proc. Third Annual Conference on Knowledge Discovery andData Mining (KDD), Newport Beach, 1997, pp. 16–23.

[4] R. Feldman, I. Dagan, and H. Hirsh, "Mining text using keyword distributions," Journal of Intelligent Information Systems, vol. 10, pp. 281–300, 2002.

[5] T. Honkela, S. Kaski, K. Lagus, and T. Kohonen, "Newsgroupexploration with WEBSOM method and browsing interface,"Laboratory of Computer and Information Science,Helsinki University of Technology, Technical Report A37,Espoo, Finland, 2001.

[6] T. Kohonen, "Self-organization of very large document collections:State of the art," in Proceedings of ICANN98, the 8thInternational Conference on Artificial Neural Networks, editedby L. Niklasson, M. Boden, and T. Ziemke, London, Springer,1998, vol. 1, pp. 65–74.

[7] S. Kaski, T. Honkela, K. Lagus, and T. Kohonen, "WEBSOM—self-organizing maps of document collections," Neurocomputing,vol. 21, pp. 101–117,2006.