Association for Information Systems
AIS Electronic Library (AISeL)

PACIS 2008 Proceedings

Pacific Asia Conference on Information Systems (PACIS)

July 2008

Supporting Document-Category Management: An Ontology-based Document Clustering Approach

Yen-Hsien Lee
National Chiayi University, yhlee@mail.ncyu.edu.tw

Ching-Yi Tu
National Chiayi University, s0951302@mail.ncyu.edu.tw

SUPPORTING DOCUMENT-CATEGORY MANAGEMENT: AN ONTOLOGY-BASED DOCUMENT CLUSTERING APPROACH

Lee, Yen-Hsien, National Chiayi University, 580 Sinmin Rd., Chiayi City 600, Taiwin, yhlee@mail.ncyu.edu.tw

Tu, Ching-Yi, National Chiayi University, 580 Sinmin Rd., Chiayi City 600, Taiwin, s0951302 @mail.ncyu.edu.tw

Abstract

Automated document-category management, particularly the document clustering, represents an appealing alternative of supporting a user's search, access, and utilization of the ever-increasing corpora of textual. Traditional document clustering techniques generally emphasize on the analysis of document contents and measure document similarity on the basis of the overlap between or among the feature vectors representing individual document. However, it can be problematic and cannot address word mismatch or ambiguity effectively to cluster document at the lexical level. To address problems inherent to the traditional lexicon-based approach, we propose an Ontology-based Document Clustering (ODC) technique, which employs a domain-specific ontology to support the proceeding of document clustering at the conceptual level. We empirically evaluate the effectiveness of the proposed ODC technique, using the lexicon-based and LSI-based document clustering techniques (i.e., HAC and LSI-based HAC) for evaluation purpose. Our comparative analysis results show ODC to be partially effective than HAC and LSI-based HAC, showing higher cluster precision across all levels of cluster recall and statistically significant in F1 measure. In addition, our preliminary analysis on the effect of granularity of concept hierarchy suggests the usage of fine-grained concept hierarchy can make ODC reach to a better performance. Our findings have interesting implications to research and practice, which are discussed together with our future research directions.

Keywords: Document-category management, Document clustering, Ontology-supported document clustering, Knowledge management

1 INTRODUCTION

The advancement and proliferation of information technology have fostered rapid creation and dissemination of information on a massive scale. As a result, voluminous information is becoming available at an explosive pace. Despite the growing popularity of multimedia, text remains the dominant form of information, particularly that available on the Internet. Availability of the vast amounts of online textual documents demands appropriate document management solutions to support a user's search, access, and utilization of the ever-increasing corpora of textual documents.

Analysis of prevalent practice suggests the common use of document category by individuals and organizations, thereby sorting documents into different folders or categories. The sheer volume of new documents and the likelihood of their assignments to appropriate categories make manual document-category management approaches prohibitively tedious and ineffective. Hence, automated document-category management represents an appealing alternative and can be greatly supported by appropriate text mining techniques. Of particular importance is document clustering, which partitions a collection of documents into distinct groups where the documents in each group share great similarity and collectively reveal a specific theme concealed in the underling document corpus (Boley et al. 1999; El-Hamdouchi & Willett 1986; Larsen & Aone 1999; Pantel & Lin 2002; Wei et al. 2006).

Traditional document clustering techniques generally emphasize on the analysis of document contents and measure document similarity on the basis of the overlap between or among the feature vectors

representing individual document. Specifically, most traditional document clustering techniques adopt a specific feature selection metric (e.g., term frequency (TF) or TF×IFD (i.e., term frequency×inverse document frequency)) (Boley et al. 1999; Larsen & Aone 1999; Pantel & Lin 2002; Roussinov & Chen 1999; Wei et al. 2006) to identify a set of representative features as the basis for document representation. Each document to be clustered is then represented as a document-feature vector according to the selected set of representative features. Subsequently, the pair-wise similarity between documents is measured on the basis of the selected features and their respective values in each document and the source documents are hereby grouped into distinct clusters. As mentioned, traditional document clustering techniques compare documents by measuring the similarity of their respective document-feature vectors (i.e., the frequency of each representative features in a document).

However, it can be problematic and cannot address the problems of word mismatch and ambiguity effectively[1] for the lexicon-based document-clustering technique (i.e., traditional document clustering technique), which performs document clustering at the lexical level. For example, the lexicon-based document-clustering technique will regard the terms "Car", "Automobile", and "Motor Vehicle" as different while they actually refer to the same concept, and hence results in the word mismatch problem. On the other hand, it cannot recognize what the term "Mouse" refers to is an animal or the computer equipment and encounters the word ambiguity problem. The problems of word mismatch and ambiguity may make lexicon-based document-clustering technique misestimate the document similarity, and further undermine its effectiveness in clustering the document corpus. Recently, research has suggested the statistical method (i.e., Latent Semantic Indexing, LSI) to address the word mismatch problem by analyzing the term correlation structure in the document corpus. LSI primarily applies Singular Value Decomposition (SVD) technique to a term-document matrix to constructs a new semantic space and represents both terms and documents in this space. The terms and documents are placed closely while they are associated closely with each other in the LSI space.

Though prior research has shown the effectiveness of LSI in document clustering (Schutze & Silverstein 1997; Lerman 1999), it still cannot provide the explanation for why documents are grouped within the same clusters. In this study, we attempt to address the problems of word mismatch and ambiguity inherent to the lexicon-based document-clustering technique by proposing an Ontology-based Document Clustering (ODC) technique to support clustering documents at the conceptual rather than the lexical level. Typically, a domain-specific ontology consists of a set of related concepts, relations, and axioms (e.g., constraints) (Keet 2004) and offers a shared, common understanding of a domain that can be easily communicated between or among humans as well as application systems (Fensel 1986). With the help of a domain-specific ontology (populated with preclassified documents), our proposed ODC technique can transform a feature-represented document into a concept-represented one. Therefore, the target document corpus will be clustered in accordance with the concepts representing individual document, and thus, achieve the proceeding of document clustering at the conceptual level. To cluster documents at the conceptual level is considering as a solution to the problems of word mismatch and ambiguity. In additional, as suggested by previous research, it as well can enhance of the explanatory utilities of the resulting document categories (Hotho et al. 2001).

The remainder of this paper is organized as follows: In Section 2, we review relevant previous research on document clustering and provide an overview of ontology and ontology enrichment approach. In Section 3, we depict our proposed Ontology-based Document Clustering (ODC) technique for clustering document corpus, followed by the description of our evaluation design and the discussion of the comparative analysis results in Section 4. Finally, the conclusion and future research directions of this study are provided in Section 5.

2 LITERATURE REVIEW

[1] Word mismatch refers to the phenomenon where different words are used to describe the same concept or object, whereas word ambiguity refers to the phenomenon where a word is used to describe different concepts or objects.

In this section, we briefly review the research works relevant to our proposed Ontology-based Document Clustering (ODC) technique, including the prior research in document clustering and an overview of ontology and ontology enrichment approach.

2.1 Review of lexicon-based document clustering techniques

Document clustering groups similar documents into distinct clusters by analyzing document contents. A document in a resulting cluster exhibits maximal similarity to those in the same cluster and shares minimal similarity with the documents in other clusters. Most document clustering techniques emphasize document contents analysis and typically consist of three phases: feature extraction and selection, document representation, and clustering (Wei et al. 2002; Wei et al. 2006).

Feature extraction starts with document parsing to produce a set of features (e.g., nouns and noun phrases), excluding pre-specified non–semantic-bearing words' i.e., stopwords. Representative features are then selected from the extracted features. Feature selection is critical to clustering effectiveness and efficiency because it reduces the number of the extracted features and removes the potential biases existing in the original (untrimmed) feature set (Dumais et al. 1998; Roussinov & Chen 1999). Common feature selection metrics include term frequency (TF), term frequency and inverse document frequency (TF×IDF), and their hybrids (Boley et al. 1999; Larsen & Aone 1999).

In the subsequent document representation phase, we use the top-*k* method to choose the *k* features that have the highest selection scores to represent each document. As a result, each document (in the corpus) is represented by a feature vector and jointly defined by the *k* features selected. A review of previous research suggests several salient feature representation methods, including binary (i.e., presence versus absence of a feature in a document), within-document TF, and TF×IDF (Larsen & Aone 1999; Roussinov & Chen 1999; Wei et al. 2006). In the final clustering phase, the source documents are grouped into distinct clusters on the basis of the selected features and their respective values in each document. Common clustering approaches include partitioning-based (Boley et al. 1999; Cutting et al. 1992; Larsen & Aone 1999; Spangler et al. 2003), hierarchical (El-Hamdouchi & Willett 1986; Roussinov & Chen 1999; Voorhees 1986; Wei et al. 2006), and Kohonen neural networks (Kohonen 1989, 1995; Guerrero Bote et al. 2002; Lagus et al. 1996; Lin et al. 1999; Roussinov & Chen 1999).

A review of the extant document clustering literature suggests the predominant use of document content analysis, thereby clustering documents into distinct categories on the basis of the overlap between or among the feature vectors representing individual documents. Such lexicon-based approach operates document clustering at the lexical level would encounters the problems of word mismatch and ambiguity. In this proposed study, we attempt to address the problems inherent to the lexicon-based document clustering techniques by exploiting a domain-specific ontology to support clustering document corpus at the conceptual level.

2.2 LSI-based document clustering technique

In document clustering applications, Latent Semantic Indexing (LSI) is used as a means of dimension reduction to improve both effectiveness and efficiency (Schutze & Silverstein 1997; Lerman 1999). In general, LSI-based document clustering techniques do not have a feature selection phase, rather than representing documents as a feature-document matrix after feature extraction. Accordingly, the LSI technique is applied to this feature-document matrix to produce new, reduced dimensions that are linear combination of the original features and to further represent the target document in the new LSI space. Specifically, LSI discerns the usage patterns of terms in documents and uses statistical technique to estimate this latent structure (Deerwester et al. 1990; Berry et al. 1995). It applied Singular Value Decomposition (SVD), a statistical analysis technique to the feature-document matrix to construct a new semantic space formed by the orthogonal vectors (namely LSI dimensions). Features and documents represented in the space are close when they are associated closely.

Specifically, SVD decomposes the original the $m \times n$ dimension feature-document matrix A into a $m \times r$ dimension matrix T, $r \times r$ dimension matrix S, and $r \times n$ dimension matrix D^T. The rows of T and D indicate the basic positions of features and documents in the semantic space (i.e., the LSI space) of r dimensions, and S is a diagonal matrix containing rescaling values that rescale the axes of the LSI dimensions. To present important associative patterns among features and documents and to remove noises, the original LSI space is reduced to a k-dimensional one by keeping the first k columns of T and D, and pruning S as a $k \times k$ matrix. As the reduced LSI space is constructed, the target documents are as well represented in the new space. Finally, a clustering algorithm can be employed directly to segment the target documents into clusters. For the purpose of comparative evaluation, we will implement a traditional lexicon-based (e.g. HAC) and a LSI-based (LSI-based HAC) document clustering techniques, and the performance achieved by these two techniques (e.g., HAC) will be adopted as our evaluation benchmarks.

2.3 Overview of ontology

Ontology refers to a systematic account of existence. Philosophically, ontology entails explicit, formal specifications of how to represent objects, concepts, and other entities (including the relationships among them) commonly assumed to exist in a domain of interest. Computationally, ontology can be used to define a common vocabulary that formally represents knowledge and facilitates its sharing and reuse. In this connection, ontology describes the specification of a representational vocabulary for a shared domain of discourse, such as definitions of class, relations, functions, and other objects (Gruber 1993).

An ontology usually includes concepts, relations, instances, and axioms formally represented in a machine readable format (Keet 2004). Concepts represent a set or class of entities in a domain and can be classified as primitive or defined. *Primitive concepts* are those that have only the necessary conditions (in terms of their properties) for membership in the class, whereas *defined concepts* are those whose description is both necessary and sufficient qualifying something to be a member of the class. Relations describe the interactions between concepts or the essential properties of a concept. An axiom constrains the value of a class or an instance. Hence, the properties of a relation can be considered as axioms. In addition, axioms can be used to denote general rules. An instance is the particular thing that a concept represents. Strictly, an ontology should not contain any instances because it is designed to provide a conceptualization of a domain of interest. We can obtain a knowledge base by combining an ontology and its associated instances. The distinction between a concept and an instance can be difficult and often is application specific.[2]

Use of ontology to support knowledge sharing and reuse has been examined in an expanding array of domains that include intelligent information integration (Kohler et al. 2003; Wache et al. 2001), knowledge based systems (Perez & Benjamins 1999) and text indexing and querying (Kohler et al. 2006). The increasing applications of ontology may be partially attributed to its support of a shared, common understanding of a particular domain with great ease of communication between/among people and systems (Fensel 1986). While showing promising value in various application areas, use of ontology has been hindered by the underlying ontology engineering process which is time-consuming and knowledge-intensive (Maedche & Staab 2000). To facilitate the construction of ontology and improve its maintenance overt time, previous research has investigated the use of supervised learning techniques to discover important ontologies from a set of document concerning a particular domain (Maedche 2002; Maedche & Staab 2000; Morin 1999; Suryanto & Compton 2000; Szpakowicz 1990; Yamaguchi 2001). Most existing learning techniques focus on concept extraction or the discovery of important relations, taxonomic (Morin 1999; Suryanto & Compton 2000; Szpakowicz 1990; Yamaguchi 2001) or associative (Maedche & Staab 2000; Yamaguchi 2001).

In this study, we define ontology as a set of concepts of interest domain organized as a hierarchical (or

[2] This distinction is challenging and remains an open question in knowledge management research.

heterarchical) structure, and each of the concepts in the hierarchy is described by a set of descriptors (Hotho et al. 2001). Many professional associations have created their own concept hierarchies, but few of them have concept descriptors readily available. For example, the Computing Classification System (CCS)[3] built by the Association for Computing Machinery (ACM) is primarily used as an indexing scheme for organizing articles published in various ACM periodicals and therefore does not define concept descriptors within the hierarchy.

Previous research has proposed the automated approach to discover the important concept descriptors for the populated concept hierarchy. For example, Lee et al. (Lee et al. 2007) proposed the ontology enrichment (OE) approach to extract the descriptors for the concepts of a given concept hierarchy from a set of well-classified documents pertinent to it. OE adopted a feature weighting function to determine the discrimination power for the feature of a concept to its sibling concepts. The weighting function of a feature f_i in the concept o_j is defined as $w_c(f_i, o_j) = TF(f_i, o_j) \times pd_{ij} \times \left[\log_2 s - \left(- \sum_{h=1}^{s} \left(\frac{pd_{ih}}{\sum_{r=1}^{s} pd_{ir}} \times \log_2 \frac{pd_{ih}}{\sum_{r=1}^{s} pd_{ir}} \right) \right) \right]$, where $TF(f_i, o_j)$ denotes the term frequency of f_i in o_j, pd_{ij} is the number of documents that contain f_i in o_j over the total number of documents in o_j, and s is the number of siblings of o_j plus 1 (i.e., including o_j itself). According to the weights of the features in respective concept, OE then selected top-k_{cd} features as the descriptors of a target concept at level one and select $(k_{cd} + (n-1)\times\delta_{cd})$ descriptors for a concept at the level n. Finally, a pre-determined commonality threshold α_p was applied to remove a descriptor if it appears in more than α_p percent of the concepts in the hierarchy. In this study, we will adopt the OE approach to discover the concept descriptors for the concept hierarchy used in the empirical evaluation.

3 DESIGN OF ONTOLOGY-BASED DOCUMENT CLUSTERING (ODC) TECHNIQUE

We discuss the design of our proposed Ontology-based Document Clustering (ODC) technique in this section. Briefly, with the availability of a domain-specific ontology, ODC technique could transform the feature-represented documents into concept-represented ones and hereby clusters them on the basis of their pertinent concepts to address the limitations inherent to the lexicon-based document clustering techniques (i.e., the word mismatch and word ambiguity problems). ODC first measures the similarity between each document in the collection and all concepts in the domain-specific ontology. Each document will be mapped onto the concept space of the domain-specific ontology and represented as a vector of concepts using the document-concept similarities. Subsequently, a set of top-k concepts is selected as the representative concepts on the basis of their importance (or discrimination power) to the target document corpus. Once selected, each document in the target document corpus will be represented using the selected representative concepts, and thus the document corpus can be clustered at the conceptual level.

As shown in Figure 1, the overall process of our proposed ODC technique consists of five phases, including *feature extraction*, *concept mapping*, *concept selection*, *concept-based document representation*, and *clustering*. The purpose of each phase is detailed as follows:

[3] http://www.acm.org/class/1998/

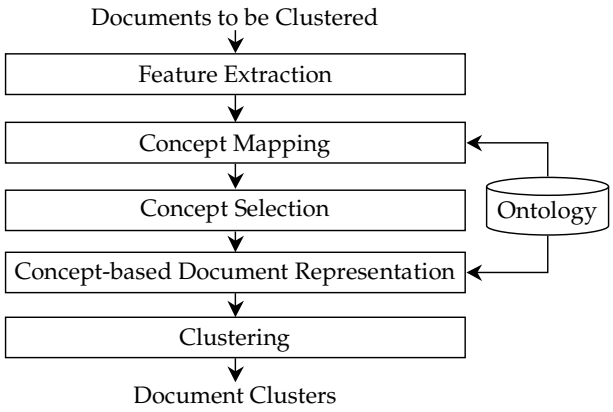

Figure 1. Overall process of ontology-based document clustering (ODC) technique

Feature Extraction: In this phase, features are extracted from the document corpus to be clustered. First of all, we use the rule-based part of speech tagger implemented by Brill (Brill 1992, 1994) to tag each word in a document. Subsequently, we follow the approach suggested by Voutilainen (1993) to develop a noun phrase parser for extracting the nouns and noun phrases from each syntactically tagged document.

Concept Mapping: The purpose of concept mapping phase is to measure the similarity between a document to be clustered d_i and each concept o_j in the domain-specific ontology and thereby converts the feature-represented documents into the concept-represented ones. In this phase, we measure the degree of relevance between a document and each concept in the concept hierarchy. The weighting function of relevance degree between a document d_i and a concept o_j is defined as $w_m(d_i, o_j) = \left\{ \sum_{f_k \in o_j} \left(w_c(f_k, o_j) \times TF(f_k, d_i) \right) \right\} \times pf_{ij}$, where f_k is one of the descriptors of the concept o_j, $w_c(f_k, o_j)$ is the weight of the descriptor f_k in the concept o_j, $TF(f_k, d_i)$ is the within-document term frequency of the descriptor f_k in the document d_i, and pf_{ij} is the percentage of the number of descriptors in the concept o_j that appears in the document d_i. According to our proposed weighting function, we sum the product of the weight of each descriptor in the concept o_j and its respective term frequency in the document d_i. The more descriptors of a concept appear in a document, the greater the confidence that the document embraces that concept. The relevance degree is then adjusted by multiplying the percentage of the number of descriptors in the concept o_j that appears in the document d_i. After measuring the degree of relevance between each document and the concepts in the hierarchy, each document will be represented as a set of weighted concepts.

Concept Selection: Upon transforming the document corpus to be clustered into the concept space of the domain-specific ontology, the concept selection phase is performed to select a set of representative concepts for the purpose of concept-based document representation. To measure the importance of a concept in relation to the whole document corpus, we adopt a revised TF×IDF measure by replacing the TF value of a concept o_j with the summation of its relevance degree appearing in all of the documents. In addition, we incorporate a small fraction to IDF (i.e., 0.001). The revised TF×IDF measure for o_j is defined as $w_s(o_j) = \left(\sum_{d_i \in D} w_m(d_i, o_j) \right) \times (\log_2 \frac{N}{n} + 0.001)$, where $w_m(d_i, o_j)$ is the relevance degree of the concept o_j in document d_i, D is the document corpus to be clustered, N is the number of documents in D, and n is the number of documents that contain concept o_j (i.e., $w_m(d_i, o_j)$ larger than 0). Finally, the top-k concepts with the highest TF×IDF scores are then selected as the global dictionary and used to represent each document in the document corpus to be clustered.

Concept-based Document Representation: After selecting the concepts of a category, ODC represents each document as a concept vector; i.e., the concept-based document representation. We adopt a weighting scheme to represent each concept in a document by assigning a particular weight to each

concept on the basis of its importance in the document. When determining the weight of a concept, we consider not only the relevance degree of the concept in a document, but also its relevant concepts in the concept hierarchy. Two concepts may be relevant when they locate closely in the hierarchy. In this study, we measure the similarity of two distinct concepts by examining their distance in the hierarchy. Given a concept hierarchy of t-level hierarchical structure, we define $(1/2)^{t-(s-1)}$ as the concept similarity at the level s in relation to its parent concept, and $(1/2)^{t-(s-1)} \times (1/2)^{t-((s-1)-1)}$ as the similarity in relation to its grandparent node. The similarity of two concepts can then be calculated using the product of the similarity of the respective concepts in relation to their closest common ancestor. In situations where the closest common ancestor of two concepts is the root node of the hierarchy, the similarity of these concepts is set to 0. In addition, we define the similarity between a concept and itself to be 1. We use the concept similarity measure defined above to adjust the weight of a concept o_j in a document d_i by the maximal product of the relevance degree of o_h in d_i and the similarity between o_h and o_j for every $o_h \in O$ where O is the set of the selected concepts. Using the matrix representation, we formally define the weight of each concept in a document as $P_{|D|\times|O|} \otimes Q_{|O|\times|O|} = R_{|D|\times|O|}$, where $P_{m\times n}$ is the document-concept matrix in which each element $p_{ij} = w_m(d_i, o_j)$ is the relevance degree of o_j in d_i, $Q_{n\times n}$ is the similarity matrix of concepts in which each element q_{ij} is the similarity between the concepts o_i and o_j, $R_{m\times n}$ is the document-concept matrix in which each element r_{ij} is the weight of the concept o_j in d_i and defined as $r_{ij} = \underset{k=1}{\overset{|O|}{\text{Max}}} (p_{ik}\times q_{kj})$, where $|O|$ is the number of the selected concepts.

Clustering: In the final phase, ODC will categorize the target document corpus into distinct clusters on the basis of the representative concepts and their respective values in each document. As mentioned, the common clustering approaches include partitioning-based, hierarchical, and Kohonen neural network. Among which, hierarchical clustering is popular and has an advantage over partitioning-based, in that the number of clusters need not be prespecified and can be decreased (or increased) by adjusting the intercluster similarity threshold (Roussinov & Chen 1999). Therefore, we employ the hierarchical clustering approach (specifically, the HAC algorithm) as the underlying clustering algorithm for our proposed ODC technique.

4 EVALUATION DESIGN AND RESULTS

4.1 Evaluation design

Evaluation Document Corpus: For the purpose of concept descriptor learning, we obtained source documents from ACM and used the ACM CCS classification structure as the concept hierarchy for learning concept descriptors. In our evaluation, we removed the first two level-one nodes, A (i.e., General Literature) and B (i.e., Hardware), and their child nodes from the concept hierarchy because of their irrelevance to the documents used in our evaluation experiment. Furthermore, the General and Miscellaneous nodes at level-two and level-three do not depict concrete concepts and therefore were excluded from the hierarchy used in our evaluation. To discover important concept descriptors, we randomly selected a total of 14,729 abstracts of research articles from the ACM digital library. Each article is indexed by one or more designations to indicate its subject area(s) within the CCS classification structure. We removed these nodes in which had only one abstract and their child nodes from the hierarchy because the number of documents is not sufficient for generating descriptors representative of such nodes. The nodes which do not have siblings were also removed from the hierarchy, because we cannot measure the relative importance of the features by the concept descriptor weighting function proposed by (Lee et al. 2007). As a result, a total of 1,032 nodes were retained in the hierarchy, including 9 nodes at level one, 49 at level two, 263 at level three, and 711 at level four.

For the evaluation purpose, we collected 433 research articles in information systems and technology

from a digital library website that specializes in the science literature.[4] Choice of our document corpus is appropriate because most standard document sets, including Reuters RCV1 and Reuters 21578, do not support the use of an established ontology, a distinct focus of our evaluation. A senior faculty of Management Information Systems reviewed all the selected articles and classified them into 17 categories. To maintain a comparable number of categories, we chose 12 categories, each of which has a minimum of 10 documents. As a result, a total of 400 articles were used in our evaluation, spanning across 12 categories and having an average of 138 words in an article. For each article (document) in the corpus, we used only its abstract in the evaluation.

Evaluation Procedure: For each document, we consider the category specified in the document corpus to be accurate; i.e., true category. To expand the number of trials, 80% of the documents are randomly selected from each true category to form a synthetic document set respectively. The synthetic document corpus will then be clustered by ODC technique as well as its benchmark technique; i.e., HAC and LSI-based HAC. To avoid the biased estimates, the random selection-and-clustering process is repeated 30 times. We evaluate the effectiveness of each investigated technique using its average performances across the 30 random trials.

Evaluation Metrics: We evaluate the effectiveness of each investigated technique in terms of cluster recall and cluster precision, both of which anchor the analysis of the association of a document pair that pertains to the same cluster (Roussinov & Chen 1999; Wei et al. 2005). To assess the inevitable tradeoff between cluster precision and cluster recall, we analyze the precision/recall trade-off (PRT) curve which depicts the effectiveness of an investigated technique under different merging thresholds; i.e., inter-cluster similarity threshold for ODC, HAC, LSI-based HAC techniques. In this study, we examine the merging threshold for each technique over the range of 0 and 1, in increments of 0.02. Evidently, PRT curves closer to the upper-right corner are more desirable than those closer to the point of origin.

4.2 Evaluation results

Prior to our comparative evaluation, we take a computational approach to tune parameters critical to the OE technique, which is employed to discover the concept descriptors in our study, as well as the investigated techniques; i.e., ODC, HAC, and LSI-based HAC techniques. Three parameters need to be determined their appropriate values in the OE technique, including the number of descriptors for each concept at level one (k_{cd}), the increment of descriptors for each concept at the next level (δ_{cd}), and a pre-specified commonality threshold required in concept refinement (α_p). On the other hand, for the ODC, HAC, and LSI-based HAC techniques, we need to tune the number of concepts (k_c), the number of representative features (k_f), and the number of spaces (k_s), respectively. We examine k_c, k_f, and k_s, ranging from 20 to 200 in increments of 20, and based on our experimental tuning results, we set k_{cd} at 20, δ_{cd} at 10, α_p at 6%, k_c at 200, k_f at 200, and k_s at 200 in the subsequent comparative evaluation experiments.

Using the parameter values selected based on our parameter-tuning analyses, we design and conduct evaluations to compare the effectiveness of ODC, HAC, and LSI-based HAC techniques achieved. As shown in Figure 2, the cluster precision of ODC noticeably outperforms that of HAC and LSI-based HAC across all levels of cluster recall. However, it's surprising that the effectiveness of the traditional feature-based HAC is advantageous over the LSI-based HAC technique. Further, we statistically test the performance differences between ODC, HAC, and LSI-based HAC by the F_1 measure, which aggregates cluster recall and cluster precision and is derived as $F_1 = \frac{2 \times CR \times CP}{CR + CP}$. For a comparison basis, we calculate the F_1 values of the clustering results each round attained by these techniques as

[4]CiteSeer Scientific Literature Digital Library, http://citeseer.nj.nec.com/

the number of clusters arrives at the number of true categories (i.e., 12). The average F_1 values of 30 trials for ODC, HAC, and LSI-based HAC techniques are 0.544, 0.418, and 0.260, respectively, and the two-tailed *t*-test result shows that ODC statistically significant advantage over HAC and LSI-based HAC in F_1 measure at the *p*-value less than 0.01. Overall, our proposed ODC outperforms HAC and LSI-based HAC techniques.

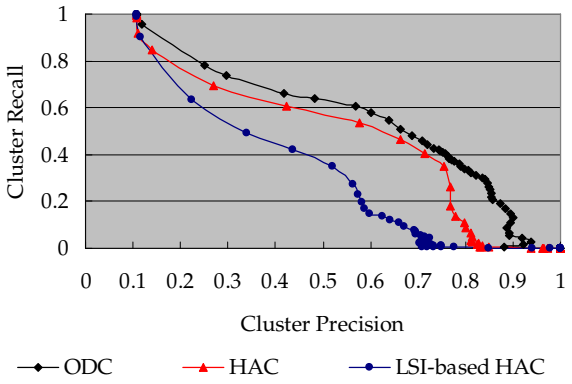

Figure 2. Comparative Evaluation Result

4.3 Effects of the granularity of the concept hierarchy

Generally, the concept hierarchy of a domain-specific ontology is defined manually and possibly coarse-grained; that's, the concept hierarchy is merely described by the high-level concepts. Therefore, we further investigate the effects of the granularity of the concept hierarchy to the effectiveness of our proposed ODC technique. To get a coarse-grained concept hierarchy, we remove all the level-four nodes (i.e., leaf nodes) from the ACM CCS classification structure to form a three level concept hierarchy, consists of 321 higher-level concepts. We conduct the evaluations to compare the effectiveness of ODC technique when adopting different granularities of concept hierarchy. As showed in Figure3, while using the fine-grained concept hierarchy (i.e., the whole ACM CCS classification structure), ODC appear more effective than that using coarse-grained concept hierarchy across all the different levels of merging threshold. Our analysis results suggest that to support the proceeding of document clustering at conceptual level, a fine-grained concept hierarchy will be advantageous over a coarse-grained one.

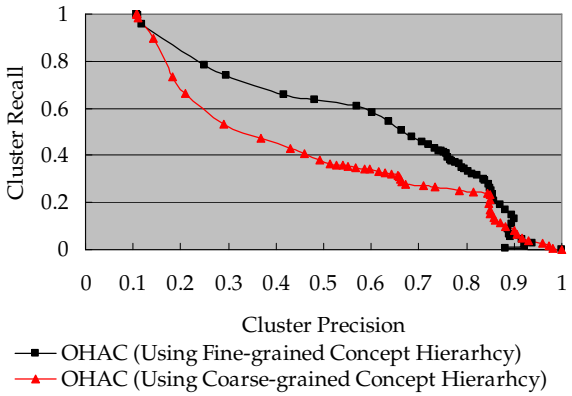

Figure 3. Effect Analysis on Granularity of Concept Hierarchy

5 CONCLUSION AND FUTURE RESEARCH DIRECTIONS

Document clustering techniques represent the appealing alternatives to supporting the management of vast amounts of textual documents. Most traditional document clustering techniques emphasize on the analysis of document contents when partitioning a set of documents into distinct groups. Such approach can be problematic and cannot address the problems of word mismatch and ambiguity effectively. Motivated by the need of more effective and advanced document management approach, we propose ODC by addressing the problems of word mismatch and ambiguity arisen when performing document clustering on the lexical level. With the use of a domain-specific ontology, ODC technique is able to categorize documents on the basis of their belonging concepts, respectively. Our empirical evaluation results reveal that the effectiveness of ODC outperforms its benchmark (i.e., the lexicon-based and LSI-based document clustering techniques, HAC and LSI-based HAC).

The research contributions of this study are twofold. First, we have contributed to document clustering research by advancing the clustering of documents on the basis of document concepts to addressing the problems of word mismatch and ambiguity faced by lexicon-based document clustering techniques. Second, the proposed ontology-based approach also contributes to document management research. The comparative evaluation results suggest that our proposed ODC technique partially outperforms the lexicon-based and LSI-based ones. Thus, the process of ODC technique can provide an illustration for ontology-based document management. On the other hand, the effect analysis to the granularity of concept hierarchy provides as well a suggestion to the depth of concept hierarchy adopted by the ontology-based document management approaches.

This study can be extended in several directions. First, we evaluate our proposed ODC technique by a domain-specific ontology (i.e., ACM CCS classification structure) and a set of relevant documents in this study. To alleviate the possible evaluation bias, more relevant document corpora are suggested to collected to evaluate our proposed technique. In addition, the adaptability of ODC to the documents of other domains should be further examined as well. Second, this study focuses on categorizing document corpus into a flat set of document clusters. Actually, documents are usually grouped into category hierarchy rather than a flat set of categories in many real-world situations. This requires the document clustering techniques capable of dealing with hierarchical structure of and relations between document categories. Finally, the proposed ODC technique provides a basis for continued ontology-based document management research. The development and evaluation of advanced ontology-based techniques for text categorization and document clustering represent interesting and essential future research directions.

References

Berry, M. W., Dumais, S. T., and O'Brien G. W. (1995). Using Linear Algebra for Intelligent Information Retrieval. SIAM Review, 37 (4).

Boley, D., Gini, M., Gross, R., Han, E., Hastings, K., Karypis, G., Kumar, V., Mobasher, B., and Moore, J. (1999). Partitioning-based Clustering for Web Document Categorization. Decision Support Systems, 27 (3), 329-341.

Brill, E. (1992). A Simple Rule-based Part of Speech Tagger. Proceedings of the Third Conference on Applied Natural Language Processing, Trento, Italy, Association for Computational Linguistics, 152-155.

Brill, E. (1994). Some Advances in Rule-based Part of Speech Tagging. Proceedings of the Twelfth National Conference on Artificial Intelligence (AAAI-94), Seattle, WA, AAAI Press, 722-727.

Cutting, D., Karger, D., Pedersen, J., and Tukey, J. (1992). Scatter/Gather: A Cluster-based Approach to Browsing Large Document Collections. Proceedings of 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 318-329.

Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., and Harshman, R. A. (1990). Indexing by Latent Semantic Analysis. Journal of the American Society for Information Science. 41 (6).

Dumais, S., Platt, J., Heckerman, D., and Sahami, M. (1998). Inductive Learning Algorithms and Representations for Text Categorization. Proceedings of the ACM 7th International Conference on Information and Knowledge Management, 148-155.

El-Hamdouchi, A. and Willett, P. (1986). Hierarchical Document Clustering Using Ward's Method. Proceedings of ACM Conference on Research and Development in Information Retrieval, 149-156.

Fensel, D. (2000). Ontologies: Silver Bullet for Knowledge Management and Electronic Commerce. Springer-Verlag, Berlin.

Gruber, T. R. (1993). A Translation Approach to Portable Ontology Specifications. Knowledge Acquisition, 5, 199-220.

Guerrero Bote, V. P., Moya Anegón, F., and Herrero Solana, V. (2002). Document Organization Using Kohonen's Algorithm. Information Processing and Management, 38 (1), 79-89.

Hotho, A., Maedche, A., and Staab, S. (2001). Ontology-Based Text Clustering. Proceedings of the IJCAI 01 Workshop on Text Learning: Beyond Supervision. August, Seattle, WA.

Keet, C. M. (2004). Aspects of Ontology Integration. Literature research & background information for the PhD proposal, School of Computing, Napier University, Scotland.

Kohler, J., Philippi, S., and Lange, M. (2003). SEMEDA: ontology based semantic integration of biological databases. Bioinformatics, 19 (18), 2420 – 2427.

Kohler, J., Philippi, S., Specht, M., and Ruegg, A. (2006). Ontology based text indexing and querying for the semantic web. Knowledge-Based Systems, 19, 744-754.

Kohonen, T. (1989). Self-Organization and Associative Memory, New York: Springer-Verlag.

Kohonen, T. (1995). Self-Organizing Maps, New York: Springer-Verlag.

Lagus, K., Honkela, T., Kaski, S., and Kohonen, T. (1996). Self-organizing Maps of Document Collections: A New Approach to Interactive Exploration. Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining.

Larsen, B. and Aone, C. (1999). Fast and Effective Text Mining Using Linear-time Document Clustering. Proceedings of the 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 16-22.

Lee, Y. H., Wei, C., and Hu, P. (2007). Preserving User Preferences in Document-Category Management: An Ontology-based Evolution Approach. Proceedings of 11th Pacific Asia Conference on Information Systems (PACIS), Auckland, New Zealand.

Lerman, K. (1999). Document Clustering in Reduced Dimension Vector Space. Unpublished, available at http://www.isi.edu/%7Elerman/papers/Lerman99.pdf

Lin, C., Chen, H., and Nunamaker, J. F. (1999). Verifying the Proximity and Size Hypothesis for Self-Organizing Maps. Journal of Management Information Systems, 16 (3), 57-70.

Maedche, A. (2002). Ontology Learning for the Semantic Web, Kluwer Academic Publishers.

Maedche, A. and Staab, S. (2000). Semi-Automatic Engineering of Ontologies from Text. Proceedings of the 12th International Conference on Software and Knowledge Engineering, Chicago, IL.

Morin, E. (1999). Automatic Acquisition of Semantic Relations between Terms from Technical Corpora. Proceedings of the Fifth International Congress on Terminology and Knowledge Engineering (TKE 99).

Pantel, P. and Lin, D. (2002). Document Clustering With Committees. Proceedings of the 25th International ACM SIGIR Conference on Research and Development in Information Retrieval, Tampere, Finland: ACM Press, 199-206.

Perez, A.G. and Benjamins, V. R. (1999). Overview of Knowledge Sharing and Reuse Components: Ontologies and Problem-Solving Methods. Proceedings of the IJCAI-99 Workshop on Ontologies and Problem-Solving Methods (KRR5), Stockholm, Sweden.

Roussinov, D. and Chen, H. (1999). Document Clustering for Electronic Meetings: An Experimental Comparison of Two Techniques. Decision Support Systems, 27 (1), 67-79.

Schutze, H. and Silverstein, C. (1997). Projections for Efficient Document Clustering. Proceedings of the 20th Annual Internationally ACM SIGIR Conference on Research and Development in Information Retrieval, Philadelphia, PA, USA, 74-81.

Spangler, S., Kreulen, J. T., and Lessler, J. (2003).Generating and Browsing Multiple Taxonomies Over A Document Collection. Journal of Management Information Systems, 19 (4), 191-212.

Suryanto, H. and Compton, P. (2000). Learning Classification Taxonomies from A Classification Knowledge Based System. Proceedings of the Workshop on Ontology Learning, Berlin, Germany.

Szpakowicz, S. (1990). Semi-automatic Acquisition of Conceptual Structure from Technical Texts. International Journal of Man-Machine Studies, 33.

Voorhees, E. M. (1986). Implementing Agglomerative Hierarchical Clustering Algorithms for Use in Document Retrieval. Information Processing and Management, 22, 465-476.

Voutilainen, A. (1993). NPtool: A Detector of English Noun Phrases. Proceedings of Workshop on Very Large Corpora.

Wache, H., Vgele, T., Visser, U., Stuckenschmidt, H., Schuster, G., Neumann, H., Hbner, S. (2001). Ontology-Based Integration of Information-A Survey of Existing Approaches. Proceedings of the 17th International Joint Conference on Artificial Intelligence (IJCAI-01), Seattle, USA, 108-117.

Wei, C. and Hu, P. J., and Dong, Y. X. (2002). Managing Document Categories in E-commerce Environments: An Evolution-based Approach. European Journal of Information Systems, 11 (3), 208-222.

Wei, C. P., Hu, P., and Lee, Y. H. (2005). An Evolution-based Approach to Preserving User Preferences in Document Category Management. Proceedings of 9th Pacific-Asia Conference on Information Systems (PACIS).

Wei, C., Yang, C. S., Hsiao, H. W., and Cheng, T. H. (2006). Combining Preference- and Content-based Approaches for Improving Document Clustering Effectiveness. Information Processing and Management, 42 (2), 350-372.

Yamaguchi, T. (2001). Acquiring Conceptual Relationships from Domain-specific Texts. Proceedings of the Second Workshop on Ontology Learning (OL 2001), Seattle, WA.