Expert Systems with Applications 42 (2015) 2264–2275

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

A semantic approach for text clustering using WordNet and lexical chains

Tingting Wei [a], Yonghe Lu [c,*], Huiyou Chang [b], Qiang Zhou [a], Xianyu Bao [d]

[a] Department of Information Science and Technology, Sun Yat-Sen University, Guangzhou, China
[b] Department of Software, Sun Yat-Sen University, Guangzhou, China
[c] Department of Information Management, Sun Yat-Sen University, Guangzhou, China
[d] Shenzhen Academy of Inspection and Quarantine, Shenzhen, China

ARTICLE INFO

Article history:
Available online 18 October 2014

Keywords:
Text clustering
WordNet
Lexical chains
Core semantic features

ABSTRACT

Traditional clustering algorithms do not consider the semantic relationships among words so that cannot accurately represent the meaning of documents. To overcome this problem, introducing semantic information from ontology such as WordNet has been widely used to improve the quality of text clustering. However, there still exist several challenges, such as synonym and polysemy, high dimensionality, extracting core semantics from texts, and assigning appropriate description for the generated clusters. In this paper, we report our attempt towards integrating WordNet with lexical chains to alleviate these problems. The proposed approach exploits ontology hierarchical structure and relations to provide a more accurate assessment of the similarity between terms for word sense disambiguation. Furthermore, we introduce lexical chains to extract a set of semantically related words from texts, which can represent the semantic content of the texts. Although lexical chains have been extensively used in text summarization, their potential impact on text clustering problem has not been fully investigated. Our integrated way can identify the theme of documents based on the disambiguated core features extracted, and in parallel downsize the dimensions of feature space. The experimental results using the proposed framework on reuters-21578 show that clustering performance improves significantly compared to several classical methods.

1. Introduction

Text clustering is a useful technique that aims at organizing large document collections into smaller meaningful and manageable groups, which plays an important role in information retrieval, browsing and comprehension. Traditional clustering algorithms are usually relying on the BOW (Bag of Words) approach, and an obvious disadvantage of the BOW is that it ignores the semantic relationship among words so that cannot accurately represent the meaning of documents. As the rapid growth of text documents, the textual data have become diversity of vocabulary, they are high-dimensional, and carry also semantic information. Therefore, text clustering techniques that can correctly represent the theme of documents and improve clustering performance, ideally process data with a small size, are greatly needed. Recently, a number of semantic-based approaches are being developed. WordNet (Miller, 1995), which is one of the most widely used thesauruses for English, has been extensively used to improve the quality of text clustering with its semantic relations of terms (Amine, Elberrichi, & Simonet, 2010; Bouras & Tsogkas, 2012; Chen, Tseng, & Liang, 2010; Dang, Zhang, Lu, & Zhang, 2013; Fodeh, Punch, & Tan, 2011; Hotho, Staab, & Stumme, 2003; Jing, Zhou, Ng, & Huang, 2006; Kang, Kim, & Lee, 2005; Recupero, 2007; Sedding & Kazakov, 2004; Song, Li, & Park, 2009).

However, there still exist several challenges for the clustering results. (1) Synonym and polysemy problems. There has been much work done on the use of ontology to replace the original terms in a document by the most appropriate ontology concept for the solution of these problems; this process is known as word sense disambiguation (WSD). This approach, however, has not proven to be as useful as first hoped. For example, approaches that expand the feature space by replacing a term with its potential concepts only increase the feature space without necessarily improving clustering performance (Fodeh et al., 2011; Hotho et al., 2003). (2) High-dimensional term features. High dimension

* Corresponding author.
E-mail addresses: tingtingwei2011@126.com (T. Wei), luyonghe@mail.sysu.edu.cn (Y. Lu), isschy@mail.sysu.edu.cn (H. Chang), gfs_007@163.com (Q. Zhou), baoxianyu@163.com (X. Bao).

http://dx.doi.org/10.1016/j.eswa.2014.10.023
0957-4174/

of feature space may increase the processing time and diminish the clustering performance, which is a key problem in text clustering. Most current techniques usually rely on matrix operation methods such as LSI (Deerwester, Dumais, Landauer, Furnas, & Harshman, 1990), ICA (Hyvärinen & Oja, 2000), and LDA (Martínez & Kak, 2001) to deal with this problem. Unfortunately, these models need too much computation. Although there also exist a few techniques have considered semantic information (Recupero, 2007; Termier, Sebag, & Rousset, 2001), they have various weaknesses. For example, they do not explicitly and systematically consider the theme of a document. (3) Extract core semantics from texts. Existing dimension-reduced methods may remove some topic features, which results in the semantic content of a document is decomposed and cannot be reflected. It is desirable to extract a subset of the disambiguated terms with their relations (known as the core semantic features) that are "cluster-aware", which leads to improving the clustering accuracy with a reduced number of terms. (4) Assign distinguished and meaningful description for the generated clusters. In order to conveniently recognize the content of each cluster, it is necessarily to assign concise and descriptive labels to help analysts to interpret the result. Nevertheless, good solutions of assigning topic labels to clusters for ease of analysis, recognition, and interpretation are still rare.

This paper attempts to alleviate mentioned above problems, its contributions can be summarized as follows.

(1) We propose a modified similarity measure based on WordNet for word sense disambiguation. This is based on the idea that the explicit and implicit semantic relationships between synsets (concepts) in WordNet impose equally importance factors in the word similarity measure. Previous works have showed that exploiting the structural information of WordNet can improve the accuracy of similarity measurement, but the effects of adding textual data to structural information are still not very extensively researched. In this paper, we explore if the combination of the structural information and the glosses of synsets can provide a more accurate assessment of the similarity between terms for word sense disambiguation.
(2) We introduce lexical chains to capture the main theme of texts. Although lexical chains have been extensively used in text summarization, their potential impact on text clustering problem has not been fully investigated. In our work, we investigate the identification of lexical chains for text representation. We have observed that the concepts extracted from lexical chains as a small subset of the semantic features can ideally cover the theme of texts, and sufficient to reduce the dimensions of feature set, potentially leading to better clustering results.
(3) We show that our method can estimate the true number of clusters by observing the experimental results, which is valuable for determining the value of k in K-means clustering algorithms.
(4) We also demonstrate that our generated topical labels have good indicator of recognizing and understanding the content of clusters. Since lexical chains represent the most of semantic content of texts, we believe that topic labels which describe and interpret the content of a cluster should be selected from the words of lexical chains. In this paper, we use the disambiguated concepts (word senses) from lexical chains in the selection of topic labels for the generated clusters, this solution is especially important when the concept title is ambiguous.

The rest of the paper is organized as follows: Section 2 reviews some related works. Section 3 presents a modified similarity measure based on WordNet for word sense disambiguation. In Section 4, we describe how to extract core semantics by using lexical chains. Section 5 details the experiments that evaluate our method and the analysis of results. Finally, we conclude this work and show its implications in Section 6.

2. Related works

To date, text clustering has been heavily researched and a huge variety of techniques has been proposed to deal with it. The goal of the clustering process is to group the documents which are similar in contents into a single group. In order to understand our work better, some relevant works about several research fields related to our interests will be introduced and the limitations of the described approaches will be presented as well.

2.1. WordNet

WordNet is one of the most widely used and largest lexical databases of English. In general as a dictionary, WordNet covers some specific terms from every subject related to their terms. It maps all the stemmed words from the standard documents into their specifies lexical categories. In this approach the WordNet 2.1 is used which contains 155,327 terms, 117,597 senses, and 207,016 pairs of term-sense. It groups nouns, verbs, adjectives and adverbs into sets of synonyms called synsets. The synsets are organized into senses, giving thus the synonyms of each word, and also into hyponym/hypernym (i.e. Is-A), and meronym/holonym (i.e. Part-Of) relationships, providing a hierarchical tree-like structure for each term.

The application of incorporating semantic features from the WordNet (Miller & Charles, 1991) lexical database has been widely used to improve the accuracy of text clustering techniques. For example, Dave et al. (Dave, Lawrence, & Pennock, 2003) employed synsets as features for document representation and subsequent clustering. However, word sense disambiguation was not performed, and WordNet synsets actually decreased clustering performance. Accordingly, Hotho et al. (2003) used WordNet in document clustering for word sense disambiguation to improve the clustering performance. Sedding and Kazakov (2004) extended this work by exploring the benefits of disambiguating the terms using their part of speech tags. The main limitation of both approaches is the increase in dimensionality of the data. Gharib, Fouad, and Aref (2010) matched the stemmed keywords to concepts in WordNet for word sense disambiguation. Their approach improves the efficiency of the applied clustering algorithms; however, it seems to over generalize the affected keywords (Bouras & Tsogkas, 2012). In the study of Amine et al. (2010), the authors accepted that the assignment of terms to concepts in ontology can be ambiguous and can lead to loss of information in their attempt to reduce dimensionality.

2.2. Semantic similarity

Semantic similarity plays an important role in natural language processing, information retrieval, text summarization, text categorization, text clustering and so on. In recent years the measures based on WordNet have attracted great concern. Many semantic similarity measures have been proposed. In general, all the measures can be grouped into four classes: path length based measures, information content based measures, feature based measures, and hybrid measures. An exhaustive overview of these approaches can be found in (Meng, Huang, & Gu, 2013). Following the cited overview, we focus on measures that are related to our work.

Rada, Mili, Bicknell, and Blettner (1989) proposed an approach based on MeSH ontology to improve text retrieval. It computed semantic similarity straightforwardly in terms of the number of edges between terms in the hierarchy. Their assumption of this approach is that the number of edges between terms in ontology is a measure of conceptual distance between terms. Wu and Palmer (1994) defined a measure of similarity between concepts based on path lengths (in number of nodes), common parent concepts, and distance from the hierarchy root. Leacock and Chodorow (1998) proposed a metric based on the count of link numbers between two set of terms or synonyms representing the same concept, and Jarmasz and Szpakowicz (2012) used the same approach with Roget's Thesaurus while Hirst and St-Onge (1998) applied a similar strategy to WordNet.

In this paper we utilize the Wu and Palmer measure and take into account the glosses of terms for word sense disambiguation. Some of the above described metrics also have been implemented for a comparison with our measure.

2.3. Lexical chains

Lexical chains derived from the research in the area of textual cohesion in linguistics (Halliday & Hasan, 2014). Cohesion involves relations between words that connect different fragments of the text. A lexical chain is a sequence of related words that give important clues about the semantic content of the text, thus, computing the lexical chains allows identification of the main topics of a document. A large number of researchers have used lexical chains for information retrieval and related areas. Morris and Hirst (1991) were the first to suggest the use of lexical chains to explore the structure of texts; they used various kinds of syntactic categories to compose lexical chains between words. Stairmand (1996) used lexical chains in the construction of both a typical IR system and a text segmentation system while Green (1997) developed a technique to automatically generate hypertext links. Hirst and St-Onge (1998) employed WordNet to study lexical chains for the detection and correction of malapropisms. Kang et al. (2005) exploited semantic relationships between words to construct concept clusters for indexing. However, these measures either have a poor performance in word sense disambiguation or inefficient to computation.

In our work we demonstrate that the lexical chains are built in terms of disambiguated terms, which not only accurately extract core semantics but also reduce the dimension of texts.

As for cluster labeling, many existing approaches generate labels with the help of external databases. For example, Tseng (2010) proposed a WordNet-based measure that first extracts category-specific terms as cluster descriptors, and then these descriptors are mapped to generic terms based on a hypernym search algorithm to create generic titles for clusters. However, this approach is very time consuming, something that leads to high execution times in order to get the required cluster labels. In contrast, our approach can generate the clusters as well as their labels reasonably fast and the assigned labels are easier to be distinguished and interpreted.

3. Word sense disambiguation

Polysemy and synonymy are two fundamental problems that affect the text representation, and they also play an important role in text clustering. Disambiguating the polysemous and synonymous nouns often yields comparable performance in document clustering (Fodeh et al., 2011). Word sense disambiguation (WSD) is a process that replacing the original terms in a document by the most appropriate sense as dictated by the surrounding context of a document. Typically, many semantic similarity measures are used for calculating the relatedness among senses. Early work varied between counting word overlaps between definitions of the word (Banerjee & Pedersen, 2003; Cowie, Guthrie, & Guthrie, 1992; Kilgarriff & Rosenzweig, 2000; Lesk, 1986) to finding distances between concepts following the structure of the LKB (Patwardhan, Banerjee, & Pedersen, 2003). As an alternative, graph-based methods have gained much attention in recent years (Agirre & Soroa, 2009; Mihalcea, 2005; Navigli & Lapata, 2010; Navigli & Velardi, 2005; Ponzetto & Navigli, 2010; Sinha & Mihalcea, 2007). Graph-based techniques are performed over the graph underlying a particular knowledge base; they first consider all the sense combinations of the words in a given context and then try to search for the relations among senses based on the whole graph. The main disadvantage of graph-based methods is their computational expense (Navigli & Lapata, 2010).

However, most previous researches exploited only one type of semantic information of knowledge base such as the structural properties. In this study, we try to explore the effect of the combination of explicit and implicit semantic relationships between synsets (concepts) on WSD. The overall procedure is presented as follows.

We adopt the WSD procedure which is given by Fodeh et al. (2011), aim to identify the most appropriate sense associated with each noun in a given document based on the assumption that one sense per discourse. The WSD approach can be described as follows. Let $N = \{n_1, n_2, \ldots, n_p\}$ denote the set of all nouns of a given document d, $n_i \in N$. Let $C_i = \{c_{i1}, c_{i2}, \ldots, c_{il}\}$ denote the set of all senses associated with the noun n_i according to the WordNet ontology. We determine the most appropriate sense of a noun n_i by computing the sum of its similarity to other noun senses in d as follows.

$$c_i = \max_{c_{ik} \in C_i} \sum_{n_j \in d} \max_{c_{jm} \in C_j} s(c_{ik}, c_{jm}) \tag{1}$$

where $s(c_{ik}, c_{jm})$ is the similarity between two senses. We restrict to the first three senses for each synset to participate in this computation for several reasons as given by Fodeh et al. (2011). First, the senses of a given noun in the WordNet hierarchy are arranged in descending order according to their common usage. Furthermore, we compare the clustering results on using only the top three senses against using all senses of a noun, the former yields similar clustering results at a reduced computation cost to the latter. This result is consistent with the experimental results obtained by Fodeh et al. (2009). In this step, the one sense of a noun that is assigned the highest score is considered the most probable sense.

There are many semantic measures have been proposed to compute the semantic similarity $s(c_{ik}, c_{jm})$ in formula (1) based on ontology hierarchy. In this work, we use the Wu–Palmer measure (Wu & Palmer, 1994) and extend it by incorporating the glosses of senses to improve the similarity measure. For the purpose of clearly present our proposed method, we first describe two related semantic similarity measures in detail and then lead to the connected use of these two methods in our method.

Wu and Palmer computed the similarity between two senses by finding the least common subsumer (LCS) node that connects their senses. For example, we can see from the red rectangle boxes in Fig. 1, the LCS of *canine* and *chap* is the lowest common node between the paths of these two senses from the root of WordNet hierarchy, *organism*. Once the LCS has been identified, the distance between two senses is computed by

$$\delta_{\text{Wu\_Palmer}}(c_p, c_q) = \frac{2d}{L_p + L_q + 2d} \tag{2}$$

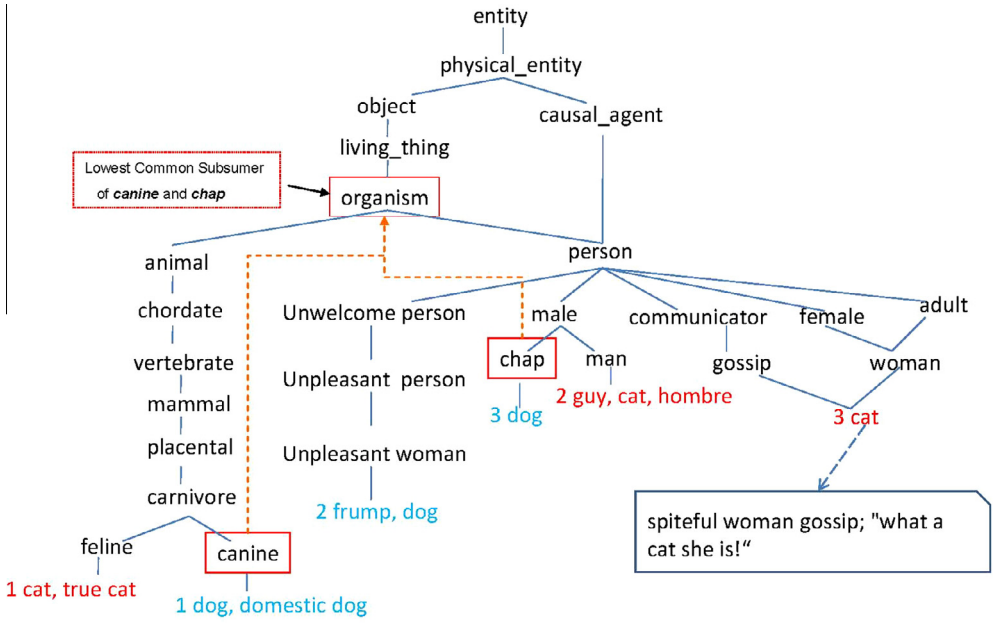

Fig. 1. A sample WordNet hierarchy.

where d is the depth of the LCS from the root, L_p is the path length between c_p and LCS, and L_q is the path length between c_q and LCS.

Based on the above Wu–Palmer similarity measure, we can calculate the similarity between each pair of synsets. However, this measure is based only on the explicit semantic relations that assuming the links between concepts represent distances; but such links do not cover all possible relations between synsets. For example, WordNet encodes no direct link between the synsets *car* and *tire*, although they are clearly related. Thus, different from Wu–Palmer measure, Banerjee and Pedersen (2003) presented a new measure of semantic relatedness between concepts that is based on the number of shared words (overlaps) in their definitions (glosses). When measuring the relatedness between two input synsets, this method not only looks for overlaps between the glosses of those synsets, but also between the glosses of the *hypernym*, *hyponym*, *meronym*, *holonym* and *troponym* synsets of the input synsets, as well as between synsets related to the input synsets through the relations of *attribute*, *similar-to* and *also-see*. For purposes of illustration, we define the description of concepts as below.

Definition 1 (*Description of a synset*). Let $C = \{c_1, c_2, \ldots, c_k\}$ be the set of synsets in a document, $c_i \in C$. Let Lemma (c_i) be the set of words that constitute a synset c_i. Let Gloss (c_i) be the definition and examples of usages of c_i. Let Related (c_i) be the union of the *hypernym*, *hyponym*, *meronym*, *holonym* and *troponym* synsets of c_i, as well as synsets related to the c_i through the relations of *attribute*, *similar-to* and *also-see*. Then the description of c_i is defined as

$$\mathrm{DES}(c_i) = (\mathrm{Lemma}(c_i) \cup \mathrm{Gloss}(c_i) \cup \mathrm{Gloss}(\mathrm{Related}(c_i))) \cap \neg\mathrm{stopwords} \quad (3)$$

In this study, we add the lemma of synset to the description set on the basis of Banerjee and Pedersen (2003). Fig. 2 shows an example for Definition 1.

Based on Definition 1, the scoring function of similarity can be defined as follows. Let $C = \{c_1, c_2, \ldots, c_k\}$ be the set of synsets in a document. DES (c_i) and DES (c_j) are description sets of two synsets c_i and c_j ($c_i, c_j \in C$), respectively. The longest overlap between these two strings is detected first, then removed and in its place a unique marker is placed in each of the two input strings; the two strings thus obtained are then again checked for overlaps, and this process continues until there are no longer any overlaps between them. Let N be the number of continuous words that overlapped, and let k be the number of iterations they had detected. Then the similarity between two synsets is computed by

$$\mathrm{Score}(\mathrm{DES}(c_i), \mathrm{DES}(c_j)) = \sum_k (N_k)^2 \quad (4)$$

The formula (4) is a formalized representation of the description given by Banerjee and Pedersen (2003). The score mechanism assigns an N continuous words overlapped the score of N^2, which gives an N-word overlapped a score that is greater than the sum of the scores assigned to those N words if they had occurred in two or more phrases, each less than N words long. This measure next assigns each possible sense a score by some other mechanisms; and sense with the highest score is judged to be the most appropriate sense for the target word.

The measure of Banerjee and Pedersen (2003) assumes that synsets description pair with more common words and less non-common words are more similar. However it cannot work well when there is not an overlap description set.

WordNet provides explicit semantic relations between synsets, such as through the is-a or has-part links, but links do not cover all possible relations between synsets; while overlaps provide evidence that there is an implicit relation between those uncovered synsets. In order to take full advantages of the measures mentioned above, we define a new similarity measure that combines both measures as below.

$$\delta(c_p, c_q) = \frac{2d + S}{L_p + L_q + 2d + S} \quad (5)$$

where $S = \log(\mathrm{Score}(\mathrm{DES}(c_p), \mathrm{DES}(c_q)) + 1)$, and the other parameters are similar with formula (2)–(4). This method not only reflects structure information of synsets, such as distance, but also incorporates content meaning of synsets in the ontology. It integrates well with explicit and implicit semantic between synsets in ontology.

Description of synset *car*#1

In WordNet, the first sense of *car* is shown as follows.

car#1 *car, auto, automobile, machine, motorcar - (a motor vehicle with four wheels; usually propelled by an internal combustion engine; "he needs a car to get to work")*

Lemma(car#1)={*car auto automobile machine motorcar* };
Gloss(car#1)={*a motor vehicle with four wheels usually propelled by an internal combustion engine he needs a car to get to work*};

Related:

Meronym(car#1)={automobile_engine#1,…, car_window#1};
Hyponym(car#1)={ambulance#1, … , minicar#1};
…
Gloss(Meronym(car#1))=Gloss(automobile_engine#1) ∪ … ∪Gloss(car_window#1);
Gloss(Hyponym(car#1))=Gloss(ambulance#1)∪ … ∪Gloss(minicar#1);
…
The description of synset *car* DES(car#1) **can be obtained by**
{**Lemma**(car#1)∪**Gloss**(car#1))∪**Gloss**(Meronym(car#1)))
∪**Gloss**(Hyponym(car#1)))∪ …} ∩¬ stopwords

Fig. 2. An example of synset description.

4. Core semantics extraction

As we have noted, disambiguating all nouns may increase the dimensionality of the feature space since a polysemous term can be replaced by multiple word senses from WordNet. We need to seek a way to reduce the dimensionality while achieving clustering performance that is comparable to using all nouns. Specifically, we will try to extract a small subset of semantic features (core semantics) with the help of information from WordNet. These core semantics are not only useful for clustering, but once identified, they may represent the main theme of the topics in the documents (Fodeh et al., 2011).

The process of core semantics extraction is similar to most taxonomy construction initiatives with the goal of finding out the representative terms and their relationships. Recently, there have been several attempts to learn taxonomies from text. For example, De Knijff et al. (2013) presented a framework for automatically constructing domain taxonomy from text corpora, they used a filtering method to extract terms from documents and then based on several domain-specific criteria to establish whether a term is selected as a concept. For those resulting concepts, the hierarchical relations among them are created using either the subsumption method or the hierarchical clustering algorithm. Meijer et al. (2014) extended this work by exploring the benefits of disambiguating the terms and concepts of a taxonomy by means of WSD. However, in both above measures, the concepts are captured with statistics and the hierarchical relations are created using the statistics-based methods, unlike the core semantic feature extraction approach used in this paper. Furthermore, they usually based on a specific domain.

In order to take full use of the semantic information from WordNet, in our study, we introduce lexical chains to extract a small subset of the semantic features (core semantics) which not only represent the theme of documents but also are beneficial to clustering. It is generally agreed that lexical chains represent the discourse structure of a document and provide clues about the topicality of a document (Kang et al., 2005; Morris & Hirst, 1991).

Lexical chains are identified by using relationships between word senses. In this work, we use the approach described in Kang et al. (2005) to build lexical chains, which considers only four kinds of relations – *identity*, *synonymy*, *hypernymy* (*hyponymy*), and *meronymy*. Compare to other traditional lexical chains, the adopted approach highlights the semantic importance degrees of the lexical items or lexical chains within a document, and which is helpful to identify the theme of a document; however, it builds lexical chains based on nouns without considering word sense disambiguation. In a cohesive and meaningful text, the word sense that is related with more word senses should be the correct sense. Thus, the way we build lexical chains in present work is based on the previous step that has been disambiguating all the nouns in the texts.

Overall, the process of extracting core semantics from texts can be decomposed into three parts. First we build lexical chains for texts based on disambiguated semantic concepts; then we adjust weights of concepts in each lexical chain by adding a weight based on the relations that this concept has with other concepts; finally, the weights of concepts in a lexical chain are added together to arrive at the score of this lexical chain, and when the score of a lexical chain pass the pre-defined requirement, the concepts in it are added to the core semantic feature sets. The three steps will be described in the subsequent sections.

Here we can view a document d as an undirected graph $G = (V,E)$ whose nodes are concepts and edges are semantic relations between concepts. Each node and edge has a weight that represents their respective degrees of semantic importance within a document. Then, a lexical chain is defined as a connected subgraph of G. Let RN = {*identity*, *synonym*, *hypernym* (*hyponym*), *meronym*} be the set of semantic relations, and let $R = \{r_1, r_2 = r_1, r_3, r_4\}$ be the set of the corresponding weight of relation in RN; weights of relations depending on the kind of semantic relationship. In this case, *identity* and *synonym* are regarded as one relation because all the nouns have been replaced by the most appropriate sense based on WordNet. For a given graph G to find the semantic relations among noun senses, and if any of these senses bear some kind of cohesive relationships, we create the appropriate links in

the graph. More formally, we perform the algorithm as depicted in Fig. 3 to construct lexical chains for document d.

For the sake of intuitively illustrating, we imitate Kang's example to show a sample of lexical chains. We apply our algorithm1 to the sample text that is extracted from reuters-21578, and we just show the generated lexical chains for simplicity. The lexical chains are shown in Fig. 4.

As we can see, a lexical chain in Fig. 4 represents semantic relations among the selected word senses of the words appearing in that lexical chain. Each node in a lexical chain is a word sense of a word, and each link can be *identity*, *synonym*, *hypernym* (*hyponym*) or *meronym* relation between two word senses. In Fig. 4, as for the form of word#*number1*#*number2*, *number1* indicates the word is used under that sense marker in this text, which reflects our constraint that one sense per discourse; *number2* is the frequencies of this word in the text, which also sums the frequencies of all words linking to this word by relation synonym.

In order to extract the core semantics, the semantic importance of word senses within a given document should be evaluated first. Generally, let $N = \{n_1, n_2, \ldots, n_p\}$ be the set of nouns in a document d, and let $F = \{f_1, f_2, \ldots, f_p\}$ be the corresponding frequency of occurrence of nouns in d. Let $C = \{c_1, c_2, \ldots, c_q\}$ be the set of disambiguated concepts that corresponding to N. Given a document d, a set of nouns N, a set of frequencies F and a set of concepts C, let $W = \{w_1, w_2, \ldots, w_q\}$ as the set of corresponding weight of disambiguated concepts in C, if c_i ($c_i \in C$) is mapped from n_k and n_m (n_k, $n_m \in N$), then the weight of c_i is computed by

$$w_i = f_k + f_m \tag{6}$$

Based on the weighted concepts, we give following definitions in terms of Kang et al. (2005).

Definition 2 (*Score of a concept*). : Let $C = \{c_1, c_2, \ldots, c_q\}$ be the set of disambiguated concepts(word senses), and let $W=\{w_1, w_2, \ldots, w_q\}$ be the set of corresponding weight of disambiguated concepts in C. Let RN = {*identity*, *synonym*, *hypernym* (*hyponym*), *meronym*} be the set of semantic relations, and let $R = \{r_1, r_2 = r_1, r_3, r_4\}$ be the set of the corresponding weight of relation in RN. Then the score of a concept c_i ($c_i \in C$) in a lexical chain is computed by

$$S(c_i) = w_i \times r_1 + \sum_{k=3}^{4}\sum_{p=1}^{q}\{w_p \times H(c_i, c_p, k) \times r_k\} \tag{7}$$

where $H(c_i, c_p, k) = \begin{cases} 1 & \text{if there exits an edge of } RN_k \text{ between } c_i \text{ and} c_p \\ 0 & \text{otherwise} \end{cases}$.

A large value of $S(c_i)$ indicates that c_i is a semantically important concept in a document. The relation weight r ($r \in R$) depending on the kind of semantic relationship and it is in the order listed: *identity*, *synonym*, *hypernym* (*hyponym*), *meronym* (thus, $r_1 = r_2 > r_3 > r_4$).

Definition 3 (*Score of a lexical chain*). Let $L = \{L_1, L_2, \ldots, L_m\}$ be a set of lexical chains of a given document, $L_i \in L$. Let $c_i = \{c_{i1}, c_{i2}, \ldots, c_{iq}\}$ be a set of disambiguated concepts in L_i. Let $S(c_{il})$ be the score of concept c_{il} ($c_{il} \in c_i$). Then, the score $S(L_i)$ of lexical chain in a document is defined as

Algorithm1. Algorithm for constructing lexical chains for document d

Input: A set of weighted disambiguated concepts $C = \{c_1, c_2, \ldots, c_q\}$ which derives from document d, a set of weighted semantic relations RS.

Output: A set of lexical chains L.

```
1. Set V=C, E=Φ, V1=V, V2=V
2. While V1 ≠ Φ:
3.     For each node v∈V1:
4.         While V2 ≠ Φ:
5.             For each node v'∈V2:
6.                 If (v≠v') and (there exists no edge between v and v') then:
7.                     {Find out and return the first relation r (r∈RS)
                        between v and v' by performing a highest-first search of
                        the semantic relations in the WordNet }
8.                     If r ≠ null then:
9.                         {Add the edge of the r relation connecting v and v'}
10.                        Set E=E ∪{v, v'}
11.            V2= V2-v'
12.    V1=V1-v
13. Return the set of connected subgraphs of G= (V, E)
```

Fig. 3. An algorithm for constructing lexical chains.

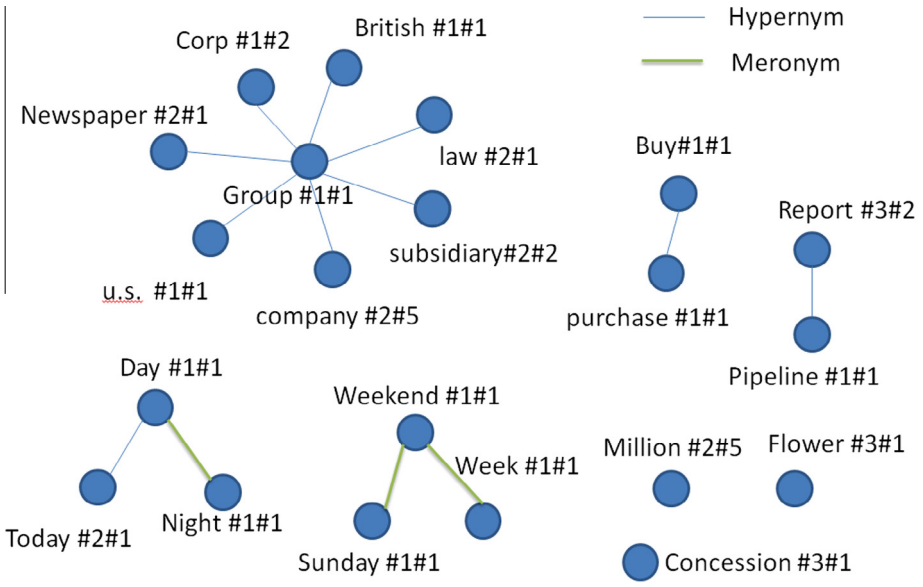

Fig. 4. A sample of lexical chains.

$$S(L_i) = \sum_{l=1}^{q} S(c_{il}) \tag{8}$$

Definition 4 (*Score of representative lexical chain*). Let $L = \{L_1, L_2, \ldots, L_m\}$ be a set of lexical chains of a given document. Let $L^R = \{L_1^R, L_2^R, \ldots, L_n^R\}$ $(n \leqslant m)$ be a set of representative lexical chains that satisfy the following criterion:

$$S(L_i^R) \geqslant \alpha \cdot \frac{1}{m} \sum_{j=1}^{m} S(L_j) \quad (i = 1, 2, \ldots, n) \tag{9}$$

where α is a weighting coefficient that is used to control the number of the representative lexical chains to be considered.

We extract the weighted concepts in the lexical chains L^R composing the set of core semantic features for the given document. It is these concepts can then be used to cluster the documents.

5. Experiments and analysis

In this section, we experimentally evaluate the performance of the proposed method. All the experiments have been performed on an Intel I5 Processor, Windows 7 OS machine with 4 GB memory. We choose WordNet version 2.0 for our experiment.

5.1. Comparison of term similarity measures

In order to evaluate our modified similarity measure, the measure of formula (2) used by Wu and Palmer (1994) and the Pedersen's extended gloss overlap (Banerjee & Pedersen, 2003) measure which we implemented it with formula (4) are conducted to compare with our work.

As for the experimental corpus in this section, there was an experiment carried out by Miller and Charles (1991) is commonly used to evaluate methods of computing the semantic similarity between words. The authors provided 30 pairs of words prior and then students were asked to rate these words for similarity in meaning on a scale from 0 (dissimilar) to 4 (highly similar). The rating scores are shown in Table 1 as the column MC.

In this experiment, we focus primarily on the improvement of our method against the classical measures, and we just calculate the similarities of many noun pairs but do not take account of their context. Furthermore, as we have noted in Section 3, the senses of a given noun in the WordNet hierarchy are arranged in descending order according to their common usage. Given the computational cost of using large graph, thus, for the purpose of simplification, we select the first sense of nouns in WordNet to build the on-line tree-like hierarchy for the given terms.

Table 1 lists the human judgments' results and our experimental results. The column of MC is the rating scores of human. The results of Wu and Palmer (1994) and our modified method are listed in columns WP and WGL, respectively; and the result of

Table 1
Miller's human judgments' result and our experimental result.

Noun pair	MC	WP	GL_{ours}	WGL
Car–automobile	3.92	0.9900	999,999	0.9900
Gem–jewel	3.84	0.3064	130.0	0.5064
Journey–voyage	3.84	0.8632	7	0.8768
Boy–lad	3.76	0.8726	19	0.8953
Coast–shore	3.7	0.9087	33	0.9316
Asylum–madhouse	3.61	0.6378	1	0.6538
Magician–wizard	3.5	0.7120	2	0.7334
Midday–noon	3.42	0.9900	999,999	0.9900
Furnace–stove	3.11	0.5098	33	0.6000
Food–fruit	3.08	0.3561	7	0.4617
Bird–cock	3.05	0.2504	2	0.3047
Bird–crane	2.97	0.5674	0	0.5674
Tool–implement	2.95	0.9279	66	0.9442
Brother–monk	2.82	0.6378	2	0.6625
Crane–implement	1.68	0.4030	0	0.4030
Lad–brother	1.66	0.6730	0	0.6730
Journey–car	1.16	0.1025	71	0.2782
Monk–oracle	1.1	0.6378	1	0.6538
Cemetery–woodland	0.95	0.4617	1	0.4928
Food–rooster	0.89	0.2671	1	0.3033
Coast–hill	0.87	0.6881	41	0.7721
Forest–graveyard	0.84	0.1683	0	0.1683
Shore–woodland	0.63	0.6378	29	0.7400
Monk–slave	0.55	0.7552	5	0.7847
Coast–forest	0.42	0.2152	1	0.2636
Lad–wizard	0.42	0.7552	1	0.7676
Chord–smile	0.13	0.4030	0	0.4030
Glass–magician	0.11	0.3297	0	0.3297
Rooster–voyage	0.08	0.0900	0	0.0900
Noon–string	0.08	0.1347	1	0.1736

extended gloss overlap is listed in column GL_{ours}, which is calculated by formula (4). In GL_{ours} measure, we assign a very large number 999,999 to the score of similarity between two identical senses.

We obtain a better performance than the other measures in the same experimental setups. We get a Pearson's correlation value 0.33 between MC and GL_{ours} and a correlation value 0.536 between MC and WP. The correlation between MC and WGL is 0.579, which indicates that our integrated measure is most coinciding with the human judgments. When we add the gloss to the path-based method, it pays a positive role to the correlation. We can see that the explicit and implicit semantic relations together can reveal hidden similarity between terms, potentially leading to better performance. Through the results, we can conclude that our method is effective.

5.2. Clustering results on text dataset

In this section, we evaluate our approach by different setups and configurations, compare the results of our method with other similar measures, and discuss insights gained.

5.2.1. Dataset

We use the reuters-21578 corpus in our experiments, which has been widely used for evaluating document clustering algorithms. The characteristic of the corpus reuters-21578 is that each text is labeled with zero, one, or more of the 135 pre-defined classes. However, the class distribution is not uniform. The size of some classes such as *earn* and *acquisition* is relatively large, while others such as *reserve* and *veg-oil* have few documents (Fodeh et al., 2011). In this dataset, we discard the unlabeled and multiple labels documents. To maintain size among the classes, we sample several subsets of reuters-21578 according to the number of documents in classes. For example, if there is a subset that the minimum size of classes in it is 15 and its maximum size is 20, we label this subset as "RC-min15-max20". Table 2 summarizes the characteristic of these subsets.

5.2.2. Evaluation metrics

In our experiment, cluster quality is evaluated by three metrics, purity, F1-measure, and entropy. Purity assumes that all the texts of a cluster we obtained are the members of the actual class for that cluster. F1-measure combines the information of precision and recall. Entropy is the sum of individual cluster entropies weighted by the cluster size. The detail of their definition can be seen in (Jing, Ng, & Huang, 2010). Note that the values for purity and similarity are percentages, and thus limited to the range between 0 and 1. The smaller the entropy value, the better the clustering result, and the larger purity and F1-measure values the better the clustering result.

In addition, in order to evaluate one method's ability to deal with high dimensionality, we introduce another criterion from research Fodeh et al. (2011) as follows. Given a baseline method B, the percentage of reduction in the number of input features can be computed by

$$\text{Reduction} = \frac{\text{Features}(B) - \text{Features}(C)}{\text{Features}(B)} \tag{10}$$

where C denotes a method of decreasing the number of features, and the function *Features* denotes the total number of features which are derived from that method.

5.2.3. Clustering schemes under comparison

In our experiments, we need to investigate the following aspects.

(1) We compare the relative changes in clustering performance after disambiguating the nouns against all nouns.
(2) In order to evaluate the effect of lexical chain features used in text clustering, we investigate the importance of core semantics in the clustering performance, and compare our method to its similar measures such as ASG03 (Hotho et al., 2003), LMJ10 (Jing et al., 2010) and CSF11 (Fodeh et al., 2011).
(3) We verify if our method can correctly estimate the number of clusters in a dataset by observing the F1-measure values of test cases.
(4) We verify if our method can generate better description labels for derived clusters.

In an attempt to gain insight into the above aspects, we perform our experiments on different setups as shown in Table 3.

5.2.4. Clustering results and analysis

We adopt the Bisecting K-means as our underlying clustering algorithm, which has been proved to be very robust in a wide variety of experiments (Hotho et al., 2003; Steinbach, Karypis, & Kumar, 2000); the clustering parameters used are the same for all methods. The parameter k is set to the known number of classes for these datasets. We evaluate the value of α in formula (9) using a brute force approach by incrementing it with a step size of 0.5 in the range between 0 and 2 on all datasets. The optimal setting is $\alpha = 1.5$, and which is a tradeoff between clustering performance and dimensionality reduction. The weights of the relations used in the clustering are set as $r_1 = r_2 = 0.8$, $r_3 = 0.5$ and $r_4 = 0.3$. We repeat each experiment 20 times and report their average values.

Table 4 shows the evaluations of all clustering results on four datasets. The best result obtained in each metric is marked in bold face. The column #Docs and #Features indicate the total number of documents and features which are derived from the corresponding method, respectively. The number of documents of CSF11 is less than the others due to there are a number of documents derived from this method that do not contain any of the core semantic features; this issue is discussed in more details later in this section. As the experiment using CSF11 on dataset RC-min408-max3945 is very time-consuming, we only conduct this method on the other three datasets.

From the Table 4, we can make the following observations.

We can see that the DC experiment results are always better than the Base, which indicates that the disambiguated concepts produced by our measure can improve the clustering performance. In addition, we note that the number of features derived from DC is higher than the Base for all datasets. The reason as we have described in Section 3 that is due to the disambiguation of each polysemous term into multiple word senses from WordNet. This result further supports the previous assumption that when

Table 2
Characteristic for our test datasets.

Datasets	Documents total	Classes total	Characteristic
RC-min15-max20	193	10	Each class has few documents and the class distribution is uniform
RC-min15-max500	1726	20	The size of classes is of large difference and the class distribution is not uniform
RC-min110-max150	1240	9	Each class has large documents and the class distribution is uniform
RC-min408-max3945	6714	3	The three biggest classes in reuters-21578

Table 3
Descriptions of methods.

Measure	Explanation
Base (all nouns)	All basic preprocessing techniques are used, i.e. extract the term set, remove all stop words, word stemmer and identify nouns from term set. WSD is not performed
DC (disambiguated concepts)	Identical to the Base, but WSD is performed according to the formula (1)
DCS (disambiguated core semantics)	Identical to the DC, but adds the process of core semantics extraction
ASG03 (Hotho et al., 2003)	Different from our disambiguation strategy in formula (1), when calculate the similarity between senses, this method selects the first sense of each noun rather than the first three senses. Moreover, it does not perform dimensionality reduction processing
LMJ10 (Jing et al., 2010)	This method mainly readjusts the term weight according to the similarity measure between terms. The term similarity is calculated with WordNet-based similarity measure. It also does not perform dimensionality reduction processing
CSF11 (Fodeh et al., 2011)	Our disambiguation strategy is similar with the one used in this study. However, it uses information gain to extract the core semantics, which is different from our lexical chains measure

Table 4
Clustering results.

Method	F1-measure	Purity	Entropy	#Docs	#Features	Reduction
RC-min15-max20						
Base	0.395	0.373	2.205	193	600	0
DC	0.43	0.389	2.132	193	754	−0.256
DCS	**0.555**	**0.523**	**1.819**	193	366	0.39
ASG03	0.359	0.326	2.506	193	579	0.035
LMJ10	0.281	0.264	2.803	193	600	0
CSF11	0.409	0.401	2.223	162	64	0.893
RC-min15-max500						
Base	0.367	0.447	2.444	1726	1677	0
DC	**0.458**	**0.524**	2.196	1726	2269	−0.353
DCS	0.447	0.523	**2.118**	1726	1458	0.135
ASG03	0.375	0.455	2.36	1726	1590	0.051
LMJ10	0.205	0.299	3.381	1726	1677	0
CSF11	0.329	0.387	2.885	1711	434	0.741
RC-min110-max150						
Base	0.522	0.452	1.745	1240	2854	0
DC	0.644	0.591	1.387	1240	3376	−0.183
DCS	**0.678**	**0.63**	**1.253**	1240	2076	0.273
ASG03	0.52	0.45	1.773	1240	2659	0.068
LMJ10	0.239	0.203	2.958	1240	2854	0
CSF11	0.488	0.46	2.159	1132	216	0.924
RC-min408-max3945						
Base	0.62	0.623	1.176	6714	3389	0
DC	0.685	0.676	1.086	6714	3963	−0.169
DCS	**0.728**	**0.733**	**0.818**	6709	2553	0.247
ASG03	0.589	0.589	1.142	6714	3157	0.068
LMJ10	0.578	0.587	1.206	6714	3389	0

replacing a term with its potential concepts may increase the feature space.

The performance of DCS is better than the Base across all datasets and in most cases is better than the DC. The reason for the relatively poor performance compare to DC is due to the dataset RC-min15-max500 has a wide range of topics and the size of classes in it is of large difference (as described in Table 2), and the core semantic features as a small portion of the total feature set might not cover all the topics in this dataset. However, we achieve a feature reduction of at least 13.5% using the DCS approach on all datasets in terms of all nouns (Base). These results suggest that using lexical chain features (core semantics) to represent the documents not only reduce the feature set dimensionality but also improve the clustering performance for many of the datasets.

Comparing DSC, ASG03, LMJ10 and CSF11 with each other, in all cases DCS gives the best results. For datasets RC-min15-max20 and RC-min110-max150, the cluster purity using CSF11 is higher than using Baseline, ASG03 and LMJ10; but CSF11 also performs poor in dataset RCmin15-max500 as our DCS measure, the reason has been described above. Moreover, although in most cases ASG03 scheme can also improve clustering results compared to the Base, the improvement are not significant. Sometimes it even obtains poor performance than the Base. We think the reason is that it selects the first sense of concept in WordNet in case of a tie between two or more senses. In all six schemas, the performance of LMJ10 is the worst, which is partly due to it does not remove some semantic noise but increase their weights. As for the aspect of dimensionality reduction, the number of features derived from ASG03 is slightly lower than the number of all nouns, the reduction of our DCS approach is between 10% and 40%, and the reduction of CSF11 is up to more than 74%.

Although the CSF11 can greatly reduce the dimensionality, it also loses much semantic information. Moreover, the greatly dimensionality reduction may lead to only a subset of the documents will be clustered, and which we call "covered documents". The documents that do not contain any of the core semantic features become "uncovered" (Fodeh et al., 2011). Table 5 lists the number of uncovered documents for three methods. The number of "uncovered" documents produced by CSF11 is higher than that produced by the other methods. Since the CSF11 measure loses much information and does not cover all documents in a dataset during the process of core semantic features extraction, it obtains poor performance compared to our DCS approach, even though we are in the same way for WSD. This result suggests the lexical chain measure is able to identify the theme of documents for clustering without losing much semantic information, which indicates that lexical chain is effectiveness in text clustering. As for the "uncovered" documents in our approach, we map them to one of the existing core feature centroids based on "closeness" of those centroids.

In short, the results in Table 4 show that the cluster quality obtained using the core semantic features is better than using all nouns and using the disambiguated concepts (or at least comparable to). The performance of using the disambiguated concepts is better than using all nouns, which suggests that our disambiguation measure can resolve the synonymous and polysemous problems commendably and improve the quality to some extent. The lexical chains features (core semantic features) produced by our DCS approach not only reduce the number of semantic concepts without losing much information; it also sufficiently captures the main theme of a document that is helpful to clustering.

5.2.5. Identify the number of clusters

In this section, we verify if our method can correctly estimate the number of clusters in a dataset by observing the experimental results with varying the number k of clusters for the parameter. As the values of purity and entropy reflect good performance of clustering with the increasing value of k, these two metrics cannot use to identify the correct number of clusters. F1-measure is a multiple evaluation method that combines recall and precision measures,

Table 5
Comparison of the number of uncovered documents.

Dataset	Documents total	ASG03	CSF11	DCS
		#Uncovered	#Uncovered	#Uncovered
RC-min15-max20	193	0	31	0
RC-min15-max500	1726	0	15	0
RC-min110-max150	1240	0	108	0
RC-min408-max3945	6714	0	N	5

ideally, the nearer the value of k to the real number of groups, the higher the F1-measure obtained, due to the higher accuracy of the clusters mapping to the original classes. Therefore, we observe the experimental results with varying the number k of clusters on F1-measure. To find the best partition, we use the Bisecting K-means algorithm with its input parameters k changes within some limit. As the number of clusters of dataset RC-min408-max3945 is too small, the experiments are conducted on the other three datasets only. The comparison results on different datasets are shown in Figs. 5–7. The true number k of clusters in a dataset is shown after the corpus name (i.e. RC-min15-max20 (k = 10)).

From Fig. 5, we see that the values of Base, ASG03 and DCS appear to increase with increasing the value of k and then tend to be constant after k is set equal to or greater than 9, from these results we can estimate the number of clusters is in a vicinity of 9 (the correct number is 10). However, it is difficult to estimate the correct number of clusters from the changes of curves for CSF11, LMJ10 and DC.

Fig. 6 presents the results on dataset RC-min15-max500. In terms of the curve of our DCS approach, we pinpoint that for the case of 20 clusters, the results begin to decrease over the rest of the cases which can be interpreted as a viable indication of the actual number of clusters our data set seems to have. Indeed, the actual number of clusters in this dataset is 20. However, the results of DC, ASG03, Base and LMJ10 show stable curve in different values of k, which makes the estimation of the correct number of clusters become a difficult job. The performance of CSF11 appears unstable when the number of clusters increases.

In Fig. 7, the trend of curve changes of ASG03, Base, DC and DCS is almost the same, which looks smooth when k is set between 8 and 10 and then is upward with increasing the value of k; from the results of these four methods we can estimate the number of clusters is in a vicinity of 10 (the actual number is 9). The results obtained by LMJ10 suggest better performance, the curve of its F1-measure values looks smooth until k is set to 9 then it declines with the increasing the value of k. From the results we can correctly estimate the number of clusters is 9 (the actual number is 9). However, the curve of CSF11 does not show regularity with the change of the value of k.

In all, our empirical results show that the testing curve of our DCS approach is very close to the ideal case, so we can easily identify the number of clusters and that is at least near or equal to the ground truth number. Moreover, our DCS approach always gets the best results across all datasets, regardless of the number of clusters.

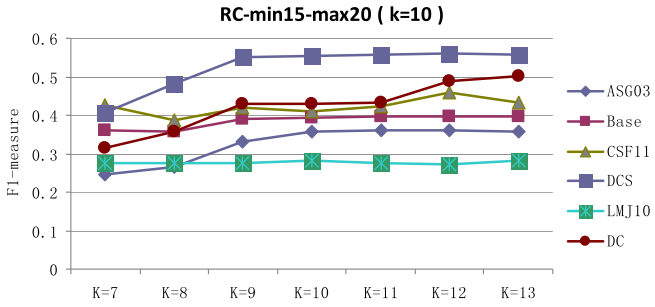

Fig. 5. Comparison results on RC-min15-max20.

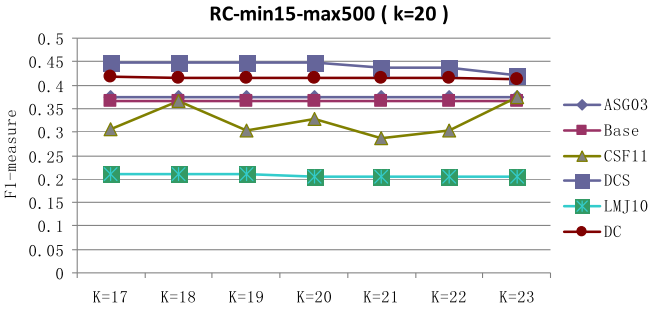

Fig. 6. Comparison results on RC-min15-max500.

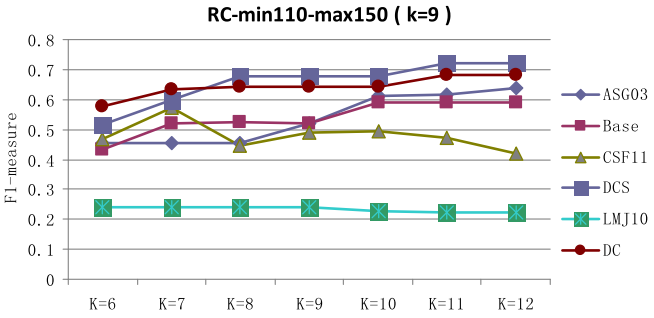

Fig. 7. Comparison results on RC-min110-max150.

5.2.6. Extract the topic labels for clusters

Labeling a clustered set of documents is an inevitable task after clustering is performed. Automatic labeling methods mainly rely on extracting significant terms from clustered documents. In this study, we extract the top-ten highest-weighted features as the cluster labels, since the weighted concepts in the extracted representative lexical chains are semantically important terms in clusters.

To verify if our method can generate better description labels for derived clusters, we compare the topic labels obtained using Bisecting K-means clustering on all nouns against the topic labels obtained from core semantic features, and investigate the correlation between our topic labels and the human labels. Due to space limitations, the results are shown for dataset RC-min408-max3945 only. This dataset contains three topics, and which are described in Table 6. Tables 7 and 8 show the list of top 10 features derived from the Base and DCS methods, respectively. In both tables, the columns 1, 2 and 3 correspond to the class *acq*, *crude* and *earn* mentioned in Table 6, separately. In addition, the features in each column are listed in decreasing order of their weights which is obtained by using formula (8).

Table 6
Descriptions of the topics contained in dataset RC-min408-max3945.

Class	Documents total	Topic descriptions
acq	2362	The label 'acq' is the short of acquisitions, and this class is comprised of documents that mainly describe the concept of stock investments and the reform of stock market
crude	408	The documents in this group are mainly talking about international oil
earn	3945	The documents in this group focus on the company's revenue

Table 7
Top 10 features of the clusters obtained by the Base method.

Group 1	Group 2	Group 3
Stake	Daily	ct
Commission	Newspaper	net
Exchange	Circulation	loss
Security	News	rev
Pct	Lord	profit
Filing	Telegraph	year
Common	Tribune	sale
Share	Sun	note
Investment	South	corp
Total	Publisher	share

Table 8
Top 10 features of the clusters obtained by the DCS method.

Group 1	Group 2	Group 3
Percentage	Uruguayan_peso	computerized_tomography
Share	Dias	net_income
Company	Panel	loss
Stock	Parcel	revolutions_per_minute
Corporation	Delaware	note
Parcel	Informant	prior
Group	Compromise	year
Million	Back	wage
Park	Philippine	April
Stockholder	Prayer	loss

From the Table 7, we observe that the features in first group (group 1) and the third group (group 3) either the same (e.g., common features 'share') or with similar meaning (e.g., 'stake' in group 1 and 'profit' in group 3, 'exchange' in group 1 and 'sale' in group 3), which make us difficult to distinguish between them. On the other hand, the top features of the group 2 have nothing to do with the topic of *crude* as which has been described in Table 5.

Using the core semantic concepts as features (Table 8), most of the top features clearly identify the topic of the class. For example, the first group contains concepts, such as 'share', 'company', 'stock', and 'stockholder', which all related to the 'acq' class; and the concepts in the group 3 are also consistent with the 'earn' class ('net_income', 'loss' and 'wage' are related to 'revenue'; 'prior', 'year' and 'April' are commonly presented in finance report of companies). Although the top features of group 2 do not include any concepts that directly related to 'crude', there still exist several concepts such as 'Uruguayan_peso' (which relies on crude oil import), 'Delaware' (its petroleum chemical industry occupies the first place in the USA), 'panel' and 'compromise' have something to do with international oil.

From the above comparison and analysis, we see that despite its simplicity, our DCS method produces labels that are almost coincide with the original given labels, and yields better results than the Base method, showing that the effectiveness of the DCS method in generating the meaningful topical labels. These topical labels have good indicator of recognizing and understanding the content of clusters. Importantly, we also record the senses along with their corresponding definitions in WordNet for ease of analysis and understanding. We therefore argue that our extracted topical labels are feasible in recognizing and interpreting the main topics of clusters.

6. Conclusions

This paper presents a methodology for clustering using WordNet and lexical chains. A modified WordNet-based semantic similarity measure is proposed for word sense disambiguation, and lexical chains are employed to extract core semantic features that express the topic of documents. We have mainly solved four problems in document clustering. The problems are disambiguating the polysemous and synonymous words, overcoming high dimensionality, determining the number of clusters, and assigning appropriate description for the generated clusters. Most previous researches tried to address only one of these four problems. But, we study on a hybrid method for solving these problems in text clustering at the same time. We show that the combination of explicit and implicit semantic relationships in WordNet pays a positive role to the assessment of word sense similarity. Furthermore, our proposed approach can estimate the true number of clusters by observing the obtained results, which is valuable for deciding the value of k in K-means clustering algorithms. In addition, we can use the top ranked concepts of each cluster to define the clusters for ease of human recognition and analysis. More importantly, we show that the lexical chain features (core semantics) can improve the quality significantly with a reduced number of features in the document clustering process. Although lexical chains have been widely used in many application domains, this study is one of the few researches which try to investigate the potential impact of lexical chains on text clustering.

However, there are still some limitations in our research. Some important words which are not included in WordNet lexicon will not be considered as concepts for similarity evaluation. In addition, the proposed method can obtain better clustering results only if the explicit and implicit relationships between words are thoroughly represented in WordNet.

In future work, we would like to perform our method on a larger knowledge base, such as Wikipedia. Moreover, since we have demonstrated that the lexical chains can lead to improvements in text clustering, the next work we plan to explore the feasibility of lexical chains in the text mining task.

Acknowledgements

This research is supported by National Key Technology R&D Program for the 12th five-year plan (Grant No. 2012BAK17B08), National Natural Science Foundation of China (Grant No. 71373291), and the Science and Technology Plan Projects of Guangdong Province (Grant No. 2012A020100008).

References

Agirre, E., & Soroa, A. (2009). Personalizing pagerank for word sense disambiguation. In *Paper presented at the proceedings of the 12th conference of the European chapter of the association for computational linguistics*.

Amine, A., Elberrichi, Z., & Simonet, M. (2010). Evaluation of text clustering methods using WordNet. *The International Arab Journal of Information Technology, 7*(4), 349–357.

Banerjee, S., & Pedersen, T. (2003). Extended gloss overlaps as a measure of semantic relatedness. In *Paper presented at the IJCAI*.

Bouras, C., & Tsogkas, V. (2012). A clustering technique for news articles using WordNet. *Knowledge-Based Systems, 36*, 115–128.

Chen, C.-L., Tseng, F. S., & Liang, T. (2010). An integration of WordNet and fuzzy association rule mining for multi-label document clustering. *Data & Knowledge Engineering, 69*(11), 1208–1226.

Cowie, J., Guthrie, J., & Guthrie, L. (1992). Lexical disambiguation using simulated annealing. In *Paper presented at the proceedings of the 14th conference on computational linguistics* (Vol. 1).

Dang, Q., Zhang, J., Lu, Y., & Zhang, K. (2013). WordNet-based suffix tree clustering algorithm. In *Paper presented at the 2013 international conference on information science and computer applications (ISCA 2013)*.

Dave, K., Lawrence, S., & Pennock, D. M. (2003). Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. *In Paper presented at the proceedings of the 12th international conference on World Wide Web*.

De Knijff, J., Frasincar, F., & Hogenboom, F. (2013). Domain taxonomy learning from text: The subsumption method versus hierarchical clustering. *Data & Knowledge Engineering, 83*, 54–69.

Deerwester, S. C., Dumais, S. T., Landauer, T. K., Furnas, G. W., & Harshman, R. A. (1990). Indexing by latent semantic analysis. *JASIS, 41*(6), 391–407.

Fodeh, S. J., Punch, W. F., & Tan, P. -N. (2009). Combining statistics and semantics via ensemble model for document clustering. In *Paper presented at the proceedings of the 2009 ACM symposium on applied computing*.

Fodeh, S., Punch, B., & Tan, P.-N. (2011). On ontology-driven document clustering using core semantic features. *Knowledge and Information Systems, 28*(2), 395–421.

Gharib, T. F., Fouad, M. M., & Aref, M. M. (2010). *Fuzzy document clustering approach using WordNet lexical categories advanced techniques in computing sciences and software engineering*. Springer (pp. 181–186).

Green, S. J. (1997). Automatically generating hypertext by computing semantic similarity. University of Toronto.

Halliday, M. A. K., & Hasan, R. (2014). *Cohesion in English*. Routledge.

Hirst, G., & St-Onge, D. (1998). Lexical chains as representations of context for the detection and correction of malapropisms. *WordNet: An electronic Lexical Database, 305*, 305–332.

Hotho, A., Staab, S., & Stumme, G. (2003). WordNet improves text document clustering. In *Paper presented at the in SIGIR international conference on semantic Web Workshop*.

Hyvärinen, A., & Oja, E. (2000). Independent component analysis: Algorithms and applications. *Neural Networks, 13*(4), 411–430.

Jarmasz, M., & Szpakowicz, S. (2012). Roget's thesaurus and semantic similarity. arXiv preprint arXiv:1204.0245.

Jing, L., Zhou, L., Ng, M. K., & Huang, J. Z. (2006). Ontology-based distance measure for text clustering.

Jing, L., Ng, M. K., & Huang, J. Z. (2010). Knowledge-based vector space model for text clustering. *Knowledge and Information Systems, 25*(1), 35–55.

Kang, B.-Y., Kim, D.-W., & Lee, S.-J. (2005). Exploiting concept clusters for content-based information retrieval. *Information Sciences, 170*(2), 443–462.

Kilgarriff, A., & Rosenzweig, J. (2000). English senseval: Report and results. In: *Paper presented at the LREC*.

Leacock, C., & Chodorow, M. (1998). Combining local context and WordNet similarity for word sense identification. *WordNet: An Electronic Lexical Database, 49*(2), 265–283.

Lesk, M. (1986). Automatic sense disambiguation using machine readable dictionaries: How to tell a pine cone from an ice cream cone. In *Paper presented at the SIGDOC '86: proceedings of the 5th annual international conference on systems documentation*.

Martínez, A. M., & Kak, A. C. (2001). PCA versus LDA. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 23*(2), 228–233.

Meijer, K., Frasincar, F., & Hogenboom, F. (2014). A semantic approach for extracting domain taxonomies from text. *Decision Support Systems, 62*, 78–93.

Meng, L., Huang, R., & Gu, J. (2013). A review of semantic similarity measures in wordnet. *International Journal of Hybrid Information Technology, 6*(1), 1–12.

Mihalcea, R. (2005). Unsupervised large-vocabulary word sense disambiguation with graph-based algorithms for sequence data labeling. In *Paper presented at the proceedings of the conference on human language technology and empirical methods in natural language processing*.

Miller, G. A. (1995). WordNet: A lexical database for English. *Communications of the ACM, 38*(11), 39–41.

Miller, G. A., & Charles, W. G. (1991). Contextual correlates of semantic similarity. *Language and Cognitive Processes, 6*(1), 1–28.

Morris, J., & Hirst, G. (1991). Lexical cohesion computed by thesaural relations as an indicator of the structure of text. *Computational Linguistics, 17*(1), 21–48.

Navigli, R., & Lapata, M. (2010). An experimental study of graph connectivity for unsupervised word sense disambiguation. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 32*(4), 678–692.

Navigli, R., & Velardi, P. (2005). Structural semantic interconnections: A knowledge-based approach to word sense disambiguation. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 27*(7), 1075–1086.

Patwardhan, S., Banerjee, S., & Pedersen, T. (2003). Using measures of semantic relatedness for word sense disambiguation. In *Paper presented at the proceedings of the fourth international conference on intelligent text processing and computational linguistics (CICLING-03)*, Mexico City.

Ponzetto, S. P., & Navigli, R. (2010). Knowledge-rich word sense disambiguation rivaling supervised systems. In *Paper presented at the proceedings of the 48th annual meeting of the association for computational linguistics*.

Rada, R., Mili, H., Bicknell, E., & Blettner, M. (1989). Development and application of a metric on semantic nets. *IEEE Transactions on Systems, Man and Cybernetics, 19*(1), 17–30.

Recupero, D. R. (2007). A new unsupervised method for document clustering by using WordNet lexical and conceptual relations. *Information Retrieval, 10*(6), 563–579.

Sedding, J., & Kazakov, D. (2004). WordNet-based text document clustering. In *Paper presented at the proceedings of the 3rd workshop on robust methods in analysis of natural language data*.

Sinha, R. S., & Mihalcea, R. (2007). Unsupervised Graph-based word sense disambiguation using measures of word semantic similarity. In *Paper presented at the ICSC*.

Song, W., Li, C. H., & Park, S. C. (2009). Genetic algorithm for text clustering using ontology and evaluating the validity of various semantic similarity measures. *Expert Systems with Applications, 36*(5), 9095–9104.

Stairmand, M. (1996). A computational analysis of lexical cohesion with applications in information retrieval. The University of Manchester.

Steinbach, M., Karypis, G., & Kumar, V. (2000). A comparison of document clustering techniques. In *Paper presented at the KDD workshop on text mining*.

Termier, A., Sebag, M., & Rousset, M. -C. (2001). Combining statistics and semantics for word and document clustering. In *Paper presented at the workshop on ontology learning*.

Tseng, Y.-H. (2010). Generic title labeling for clustered documents. *Expert Systems with Applications, 37*(3), 2247–2254.

Wu, Z., & Palmer, M. (1994). Verbs semantics and lexical selection. In *Paper presented at the proceedings of the 32nd annual meeting on association for computational linguistics*.