Self-adaptive GA, Quantitative Semantic Similarity Measures and Ontology-based Text Clustering

Chengzhi ZHANG
Department of Information Management, Nanjing University of Science & Technology,Institute of Sci & Tech Information of China
zhangchz@istic.ac.cn

Wei SONG
Division of Electronics and Information Engineering, Chonbuk National University, Jeonju, Jeonbuk, Korea
songwei@chonbuk.ac.kr

Chenghua LI
Division of Electronics and Information Engineering, Chonbuk National University, Jeonju, Jeonbuk, Korea
chli@chonbuk.ac.kr

Wei YU
Institute of Scientific & Technical Information of China, Beijing, China
yuwei@istic.ac.cn

Abstract:

As the common clustering algorithms use vector space model (VSM) to represent document, the conceptual relationships between related terms which do not co-occur literally are ignored. A genetic algorithm-based clustering technique, named GA clustering, in conjunction with ontology is proposed in this article to overcome this problem. In general, the ontology measures can be partitioned into two categories: thesaurus-based methods and corpus-based methods. We take advantage of the hierarchical structure and the broad coverage taxonomy of Wordnet as the thesaurus-based ontology. However, the corpus-based method is rather complicated to handle in practical application. We propose a transformed latent semantic analysis (LSA) model as the corpus-based method in this paper. Moreover, two hybrid strategies, the combinations of the various similarity measures, are implemented in the clustering experiments. The results show that our GA clustering algorithm, in conjunction with the thesaurus-based and the LSA-based method, apparently outperforms that with other similarity measures. Moreover, the superiority of the GA clustering algorithm proposed over the commonly used k-means algorithm and the standard GA is demonstrated by the improvements of the clustering performance.

Keywords:

Clustering; ontology; latent semantic analysis; semantic similarity measure; genetic algorithm

1. Introduction

Due to the development of knowledge and information on the World Wide Web and the growth of large data collection, the need for the efficient, high quality partitioning of texts into previously unseen categories is a major topic for applications. Document clustering techniques have been employed frequently to support these applications. Clustering [1], [2] is a commonly used unsupervised classification technique which partitions the input space into *K* regions based on some similarity or dissimilarity metric. The partition is done such that patterns within a cluster are more similar to each other than patterns belonging to different clusters. Several clustering techniques are available in the literature. In the graph theoretic approach [3], a directed tree is constructed by estimating the density gradient at each point among the data set. The clustering is performed by searching for the valley of the density function. It is known that the quality of the result depends wholly on the quality of the estimation technique for the density gradient. K-means algorithm [4], one of the most widely used, optimize the distance criterion either by minimizing the within cluster spread, or by maximizing the inter-cluster separation. It is an iterative hill-climbing algorithm suffering from the limitation of sub optimization which is known to depend on the choice of initial clustering distribution. Since stochastic optimization approaches can effectively avoid convergence to a suboptimal solution, these approaches can be used to find a globally optimal solution. Genetic algorithms (GAs) [5], [6] are randomized search and optimization techniques guided by the principles of evolution and natural genetics. However, most of these clustering algorithms solely use vector space model (VSM) to represent text, namely, each unique word in vocabulary represents one dimension in vector space. The bag of words representation adopted for these clustering methods is often insufficient because it matches directly on keywords. Since the same concept can be described using many different terms. VSM method ignores relations between some important words which do not co-occur literally. Meanwhile, with such an intuitionistic representation, we ignore some more general concepts which can help identifying related topics. For example, a text about “crawler” may not be associated with a text about “amphibian” by traditional matching algorithms. But if we add more general concept “animal” to both documents, their semantic relationship is revealed.

In this paper an improved GA based on ontology is proposed for document clustering. We use the broad-coverage taxonomy and hierarchical structure of Wordnet [7] as thesaurus-based ontology to detect semantic relationships [8] between words. Meanwhile, a new transform based on the original latent semantic analysis (LSA) is proposed and demonstrated to construct a

corpus-based text representation which can appropriately depict the associative semantic relationship. Moreover, considering the influence between the diversity of the population and the selective pressure, a self-adaptive evolution process is put forward in this article.

The remainder of this paper is organized as follow: Section 2 explains how to calculate semantic similarity from Wordnet. In section 3 the transformed LSA model are proposed for corpus-based text representation. The details of genetic algorithm for document clustering based on the ontology are described in section 4. Experiment results and analysis are given in section 5. Conclusions are given in section 6.

2. Semantic similarity based on ontology

Semantic similarity is a generic issue in the application of data mining and natural language processing fields. Semantic similarity between two words is often represented by similarity between the concepts associated with the two words. In general, the semantic similarity measures can be partitioned into two categories: thesaurus-based methods and corpus-based methods.

2.1. Wordnet

WordNet is an online lexical database of English, developed under the direction of Miller [7]. Nouns, verbs, adjectives and adverbs are grouped into sets of cognitive synonyms, named synsets, each expressing a distinct concept. Synsets are interlinked by means of conceptual-semantic and lexical relations. The version utilized in this paper is WordNet 2.0, which has 144684 words and 109377 synsets.

2.2. Semantic similarity calculation by Wordnet

We adopt the semantic similarity measure given by Li and Bandar et al [9] in this part. Two factors for calculating the similarity between two concepts are taken into account: 1) The length of the shortest path between the two concepts and 2) the depth of the subsumer in the hierarchy. That is, given two concepts c_1 and c_2, then the semantic similarity is denoted by:

$$sim(c_1, c_2) = f_1(l) \cdot f_2(h) \qquad (1)$$

where l is the length of the shortest path between concept c_1 and c_2. h is the depth of subsumer in the hierarchy semantic nets. Here, it is assumed that the influences between parameters l and h on the similarity are independent from each other. Thus the similarity function is comprised of two independent functions of f_1 and f_2.

If a word has multiple meaning, various paths may exist. So the minimum length of the path connecting two concepts is a direct approach to calculate the similarity. It is intuitive that the similarity between two concepts would nonlinear decrease as the shortest path connected them increase. Also, f_1 can be considered as an extension of Shepard's law [10], which claims that exponential-decay functions are a universal law of stimulus generalization for psychological science. Therefore, it would be reasonable to expect the similarity would decrease at an exponential rate [9] and f_1 is defined by:

$$f_1(l) = e^{-\alpha l} \qquad (2)$$

where α is a real constant between 0 and 1. From (2) we can see that when the path length decreases to zero, the similarity would monotonically increase toward the limit 1. While the path length increases infinitely, the similarity should monotonically decrease to 0.

However, only the shortest path for semantic similarity calculation may be not so accurate. To correct this problem, the shortest path length method must be revised by adding more information from the hierarchical semantic structure of Wordnet. It is intuitive that concepts at higher levels of the hierarchy have more general information, while concepts at lower levels have more concrete semantics and stronger similarity. Thus, the depth of concept in the hierarchy should be taken into account [9]. The depth h of the subsumer is derived by calculation the shortest length of links from the subsumer to the root concept of the ontology. In the light of this observation, the depth function to similarity is defined by:

$$f_2(h) = \frac{e^{\beta h} - e^{-\beta h}}{e^{\beta h} + e^{-\beta h}} \qquad (3)$$

where $\beta > 0$ is a smoothing factor. We have got the semantic similarity between two concepts based on the thesaurus method by far. The corpus-based (or information-based) method is a rather difficult issue to tackle. We can not easy to obtain it solely from the semantic nets. But it can be calculated with the help of a large corpus [11], [12]. The Brown Corpus [13] is the first of the modern, computer readable, general corpora. However, the scope of the corpus is limited due to the specific applications of the various actual datasets. Meanwhile, it takes long time to calculate the probability of encountering an instance of the concept in the large corpus. In the next section we will propose a new corpus-based semantic similarity measure.

Because a word may have multiple meaning, the semantic similarity between words is then represented by the maximum value of the similarity of concepts which are related to the words. Assuming word w_1 is represented by a number of a concepts ($c_{1,1}, c_{1,2}, \ldots c_{1,a}$) and word w_2 is represented by a number of b concepts ($c_{2,1}, c_{2,2}, \ldots, c_{2,b}$), the semantic similarity between these two words is assessed by:

$$sim(w_1, w_2) = \max\{sim(c_1, c_2)\}$$
$$c_1 \in \{c_{1,1}, c_{1,2}, \ldots, c_{1,a}\}, c_2 \in \{c_{2,1}, c_{2,2}, \ldots, c_{2,b}\} \qquad (4)$$

And then, the semantic similarity between these two documents is defined by:

$$sim_{ONTO}(d_1, d_2) = (\sum_{i=1}^{m} \sum_{j=1}^{n} sim(w_{1,i}, w_{2,j})) / mn \quad (5)$$

Where m and n are the number of Wordnet lexicon words included in documents d_1 and d_2 respectively. If we only apply such semantic similarity in our system, some problems may occur in the actual applications. For example, a specialized topic may seldom contain the Wordnet lexicon words or some formal words are broken up to incomplete forms after stemmer and will not be found in Wordnet lexicon.

Strategy 1. A hybrid model practically combines the ontology-based and VSM-based similarity measures. Therefore, the similarity between two documents can still be effectively evaluated even in case the ontology-based measure does not work well. The hybrid model is given by:

$$sim(d_1, d_2) = \lambda sim_{VSM}(d_1, d_2) + (1-\lambda) sim_{ONTO}(d_1, d_2) \quad (6)$$

where λ is a real constant between 0 and 1. So in strategy 1 the semantic similarity between two documents is the weighted summation of VSM-based and ontology-based measures.

We utilize cosine measure to compute the similarity between two documents in vector space model. Here we outline the basic formula to calculate cosine similarity in VSM. After normalization, document d_1 is represented by ($w_{1,1}$, $w_{1,2}$,...$w_{1,n}$) and document d_2 is represented by ($w_{2,1}$, $w_{2,2}$,...$w_{2,n}$) where w is the weighted value for each term. So the cosine similarity between documents d_1 and d_2 is defined by:

$$sim_{VSM}(d_1, d_2) = (\sum_{k=1}^{n} w_{1,k} \cdot w_{2,k}) / (\sqrt{\sum_{k=1}^{n} w_{1,k}^2} \cdot \sqrt{\sum_{k=1}^{n} w_{2,k}^2}) \quad (7)$$

In strategy 1 we take advantage of the ontology-based measure in conjunction with VSM-based method as the hybrid strategy to calculate the semantic similarity between documents. Whereas, due to the lack of a well-formed corpus, which is necessary for the calculation of corpus-based approach, in next section we propose a modified corpus-based semantic measure which utilizes latent semantic analysis technology to reveal associated relationships between documents.

3. Semantic similarity calculation by latent semantic analysis

Latent semantic analysis (LSA) is an automatic approach which can overcome the problems by using statistically derived conceptual indices instead of individual words. It utilizes singular value decomposition (SVD) to decompose the large term-by-document matrix into a set of k orthogonal factors [14]. SVD is an elaborate mathematic concept which extracts dominant features of large data sets and reduces the dimensionality. Thus, in this semantic and dominant structure, we can find the associative relationships, even two documents don't share any common words, because the similar semantic contexts in the texts will have similar vectors in the semantic feature space.

3.1. Singular value decomposition

The term-by-document matrix can be initially represented as $A(m \times n)$ matrix, where m is the number of distinct terms and n is the number of documents in data set. The singular value decomposition of A is given by:

$$A = U \Sigma V^T \quad (8)$$

where U and V are the matrices of term vectors and document vectors associated to the original matrix A. Σ is the diagonal matrix of singular values. To reduce dimensions, we can simply choose the k largest singular values, so the approximation matrix A_k is given by:

$$A_k = U_k \Sigma_k V_k^T \quad (9)$$

3.2. The transformed LSA for document representation

In this study we propose a transformed LSA to create a corpus-based document representation which can hopefully reveal the true semantic relationship between documents. A document d is initially represented as a $m \times 1$ matrix, where m is the number of terms. Because matrix U in (8) represents the matrix of terms vectors and the proper rank U_k spans the basis vectors. In our approach we use the multiplying of matrices d^T and U_k to represent the document vector. So each document vector in our method is defined by:

$$\hat{d} = d^T U_k \quad (10)$$

And then, the semantic corpus can be organized by:

$$C = D U_k \quad (11)$$

where D is the document-by-term matrix. The whole dimension of matrix C can precisely imitate the original document-by-term matrix, which will be proven in experiments in section 5.

Once the new corpus is given, the documents similarity can be computed by cosine similarity measurement. Assuming two transformed documents d_1 and d_2 are represented by ($c_{1,1}$, $c_{1,2}$,...$c_{1,n}$) and ($c_{2,1}$, $c_{2,2}$,...$c_{2,n}$), respectively. The cosine similarity between d_1 and d_2 in the transformed LSA method is defined by:

$$sim_{LSA}(d_1,d_2)=(\sum_{p=1}^{k}c_{1,p}\cdot c_{2,p})/(\sqrt{\sum_{p=1}^{k}c_{1,p}^2}\cdot\sqrt{\sum_{p=1}^{k}c_{2,p}^2}) \quad (12)$$

In our transformed LSA method different ranks k of corpus C are chosen to compute the similarity between each pair of documents. Then the best rank k is selected in our experiment.

Strategy 2. We propose an improved hybrid model which combines the ontology-based and LSA-based measures for documents similarity calculation. The new document similarity measure is given by:

$$sim'(d_1,d_2)=\delta sim_{LSI}(d_1,d_2)+(1-\delta)sim_{ONTO}(d_1,d_2) \quad (13)$$

where δ is a real constant within the range from 0 and 1. From the definition of (13) we can see that the value of δ relies on the quality of ontology and the parameter k selected in LSA. In the next section a self-adaptive genetic algorithm based on the ontology proposed is implemented for text clustering.

4. Genetic algorithm for document clustering based on the ontology

Genetic algorithms (GAs) are known to provide significant advantages over conventional search method by using the principals of natural selection and heuristics. They are efficient, adaptive and robust search processes which can provide near-optimal solutions for objective or fitness function of an optimization problem. Some important factors which affect the success of GAs include a continuous balance between selection pressure and diversity, and a global-wide search. In GAs, each individual is encoded in the form of chromosome. A collection of chromosomes is called a population and first of all a randomly distributed population is created. A fitness function is defined to measure the relative degree of fitness for each chromosome. Biologically inspired evolution operators continue several generations till the termination criterion is satisfied.

There is a strong influence existing between selection pressure and diversity. An enhancement of the selection pressure might expand the proportion of chromosomes directly replicated from last generation and decrease the diversity of population which may lead a premature convergence to a suboptimal solution. On the contrary, an enhancement of diversity might reduce the number of chromosome inherited and cause too much time to evolve excellent offspring. In this paper we propose a self-adaptive GA which can adjust the influence between these two factors.

4.1. Chromosome encoding

Each chromosome is a sequence of real numbers representing cluster centers. Suppose a chromosome ch_i is initially comprised of K centers.

$$ch_i=\{center_{i,1},center_{i,2},...,center_{i,K}\} \quad (14)$$

For strategy 1 each center $center_i$ is initialized by a random selected document in VSM.

$$center_i=\{w_{i1},w_{i2},...,w_{im}\} \quad (15)$$

where m is the number of terms in the document after normalization. In contrast, for strategy 2 each center $center_i'$ is initialized by a randomly selected document from corpus C in (11).

$$center_i'=\{c_{i,1},c_{i,2},...,c_{i,n}\} \quad (16)$$

where n is the number of documents in data set. From the definition of our LSA-based model the dimensions of each center can be reduced from n to k ($k<n$).

$$center_i''=\{c_{i,1},c_{i,2},...,c_{i,k}\} \quad (17)$$

4.2. Self-adaptive evolution operators

Biologically inspired evolution operators, like selection, crossover and mutation, are utilized to generate new children. The selection process is directed under the concept of roulette wheel. The proportion of selection is s. Two types of changes can occur to the survivors: crossover and mutation. A classical single-point crossover is used in this paper. Here we assume the proportion of crossover is c. In the light of the concept of Gaussian mutation [15], [16], each chromosome yields its offspring as survival to next generation. The proportion of mutation is m.

A self-adaptive GA with dynamic evolution operators is proposed in this study. When the iteration of the best individual without improvement reaches consecutive $n_{\max}$, the diversity of the population need to be enhanced by increasing the proportions of crossover and mutation, while the proportion of selection is decreased. Otherwise, the parameter for each operator is kept as its original value. Moreover, we should ensure that the proportions of crossover and mutation would monotonically increase toward a limit. Here we empirically set the limit as 0.5. The dynamic evolution operators are defined by:

$$c'=\frac{e^{\tau g}-e^{-\tau g}}{2(e^{\tau g}+e^{-\tau g})} \quad (18)$$

$$m'=\frac{e^{\sigma g}-e^{-\sigma g}}{2(e^{\sigma g}+e^{-\sigma g})} \quad (19)$$

$$s'=1-\frac{e^{\tau g}-e^{-\tau g}}{2(e^{\tau g}+e^{-\tau g})}-\frac{e^{\sigma g}-e^{-\sigma g}}{2(e^{\sigma g}+e^{-\sigma g})} \quad (20)$$

$$n_{\max}=\max\{\frac{1}{2\tau}\ln\frac{1+2c}{1-2c},\frac{1}{2\sigma}\ln\frac{1+2m}{1-2m}\} \quad (21)$$

where τ and σ are two constants. g is the number of the consecutive iterations without being improved. After consecutive $n_{\max}$ iterations without enhancement, the dynamic progress is performed and the proportions of crossover and mutation must greater than their original values, respectively.

4.3. Fitness function and termination criterion

The fitness function is defined as $1/DB$, where DB is Davies-Bouldin index [17], [18]. The algorithm is terminated when the iterations of the best individual without improvement reach consecutive $N_{\max}$ ($N_{\max} > n_{\max}$).

5. Experiments results and analysis

In order to measure the performance of our system, we implement our clustering algorithm to the Reuters-21578 test collection, which is one of the most-widely used benchmark data set in text mining and information retrieval fields. We use the subset of 200 documents from topics coffee, crude, sugar and trade for test. After being preprocessed, there are 3318 indexing terms. We firstly compare the validity of each semantic similarity measurements and then, our improved genetic algorithm is implemented for text clustering, along with its comparison with standard GA and k-means with the same similarity measurement.

5.1. The comparison of the various similarity measurements

Three similarity measurements provided by sim_{VSM} (7), sim_{LSA} (12) and sim_{onto} (5) are compared in this part to demonstrate the effectiveness of the proposed similarity measure in this study. Our method is implemented with the following parameters: α =0.08, β =0.6, λ =0.25, δ =0.45, τ =0.5, σ =0.3, S=0.6, c =0.3, m =0.1 and the number of consecutive iterations $N_{\max}$ for termination criterion is 20. The simple comparisons of similarities are illustrated in Table 1 and Table 2, respectively.

Table 1. The partial results of the various similarity measurements for homogeneous documents (coffee).

	Ⅰ. $\mathrm{Sim_{VSM}}$					Ⅱ. $\mathrm{Sim_{LSA}}$ (k =200)				
Doc#	Cof1	Cof2	Cof3	Cof4	Cof5	Cof1	Cof2	Cof3	Cof4	Cof5
Cof1	1.0	0.7258	0.4339	0.4145	0.4773	1.0	0.7258	0.4339	0.4145	0.4773
Cof2	0.7258	1.0	0.3211	0.4076	0.4451	0.7258	1.0	0.3211	0.4076	0.4451
Cof3	0.4339	0.3211	1.0	0.5053	0.4746	0.4339	0.3211	1.0	0.5053	0.4746
Cof4	0.4145	0.4076	0.5053	1.0	0.5859	0.4145	0.4076	0.5053	1.0	0.5859
Cof5	0.4773	0.4451	0.4746	0.5859	1.0	0.4773	0.4451	0.4746	0.5859	1.0
	Ⅲ. $\mathrm{Sim_{LSA}}$ (k=120)					Ⅳ. $\mathrm{Sim_{ONTO}}$				
Doc#	Cof1	Cof2	Cof3	Cof4	Cof5	Cof1	Cof2	Cof3	Cof4	Cof5
Cof1	1.0	0.9723	0.4840	0.4568	0.5646	1.0	0.5753	0.5982	0.4565	0.5238
Cof2	0.9723	1.0	0.3347	0.4740	0.5944	0.5753	1.0	0.5651	0.4117	0.4825
Cof3	0.4840	0.3347	1.0	0.6172	0.4787	0.5982	0.5651	1.0	0.5480	0.5826
Cof4	0.4568	0.4740	0.6172	1.0	0.6536	0.4565	0.4117	0.5480	1.0	0.6562
Cof5	0.5646	0.5944	0.4787	0.6536	1.0	0.5238	0.4825	0.5826	0.6562	1.0

Table 1 illustrates the similarities between pairs of homogenous documents in topic coffee. The results in Part Ⅰ are provided by cosine similarity in VSM. Part Ⅱ and Part Ⅲ provide the LSA-based document similarity results. On one hand, we can see the entire space of LSA in Part Ⅱ precisely simulates VSM and provides the same results. On the other hand, the space of the appropriate k dimensions in LSA captures the semantic relationships among homogenous documents and performs better than cosine measure in VSM. Part Ⅳ illustrates the results given by the ontology-based similarity measure. In comparing with the cosine similarity, the ontology-based measure provides more dramatic results, in that the former approach fails to give an appropriate assessment for the similarity between two semantically relevant documents represented by different terms, but the latter approach well handles such situation. However, the similarity between Cof1 and Cof2 is 0.5753, which is smaller than the values give by VSM-based measure and LSA-based measure (k=120), with values 0.7258 and 0.9723, respectively. The

reason for this phenomenon is that the special term "coffee" has high term frequencies in documents Cof1 and Cof2, so the two latter measurements provide relative high similarities. However, after stemming the term "coffee" is transformed to an incomplete form "coffe", which is not included in Wordnet lexicon and will not be considered as a concept for similarity calculation in Sim_{ONTO}. Thus, the value of Sim_{ONTO} between Cof1 and Cof2 is small.

Table 2. The partial results of the various similarity measurements for heterogeneous documents (coffee & trade).

	Ⅰ. Sim_{VSM}					Ⅱ. Sim_{LSA} (k =200)				
Doc#	Cof1	Cof2	Cof3	Cof4	Cof5	Cof1	Cof2	Cof3	Cof4	Cof5
Trade1	0.0434	0.0682	0.0506	0.0114	0.0479	0.0434	0.0682	0.0506	0.0114	0.0479
Trade2	0.0406	0.0959	0.0189	0.0129	0.0970	0.0406	0.0959	0.0189	0.0129	0.0970
Trade3	0.0882	0.0693	0.0514	0.0583	0.1071	0.0882	0.0693	0.0514	0.0583	0.1071
Trade4	0.0215	0.0339	0.0084	0.0457	0.1431	0.0215	0.0339	0.0084	0.0457	0.1431
Trade5	0.0211	0.0332	0.0082	0.2263	0.0872	0.0211	0.0332	0.0082	0.2263	0.0872
	Ⅲ. Sim_{LSA} (k=120)					Ⅳ. Sim_{ONTO}				
Doc#	Cof1	Cof2	Cof3	Cof4	Cof5	Cof1	Cof2	Cof3	Cof4	Cof5
Trade1	0.0484	0.0697	0.0585	0.0191	0.0500	0.0460	0.0559	0.0458	0.0512	0.0579
Trade2	0.0430	0.1090	0.0155	0.0104	0.1073	0.0496	0.0640	0.0434	0.0457	0.0746
Trade3	0.1274	0.0920	0.0724	0.1086	0.1324	0.0593	0.0437	0.0565	0.0613	0.0581
Trade4	0.0167	0.0372	0.0062	0.0435	0.1289	0.0414	0.0496	0.0462	0.0580	0.2433
Trade5	0.0188	0.0380	0.0060	0.0040	0.0541	0.5484	0.4545	0.4591	0.4669	0.3620

In Table 2 we compare the similarities between heterogeneous topics coffee and trade. In part Ⅰ the similarities between the most pairs of documents are very small. Also, in part Ⅱ the whole dimension of our transformed LSA model precisely imitates VSM. In part Ⅲ some similarities become bigger and some similarities become smaller when comparing with the values between the corresponding pairs in part Ⅰ and part Ⅱ. That's because LSA is to reveal the latent semantics based on the context, but not solely from individual terms. In part Ⅳ the ontology-based similarity is illustrated. In contrast, the similarities between a few pairs of heterogeneous documents are not significantly smaller than that between the pairs in homogeneous documents although this method on the average performs well to distinguish heterogeneous documents. As shown in part Ⅳ, the similarities between Trade5 and Cof1, Cof2, Cof3, Cof4, Cof5 are apparently distinct from the similarities between other pairs of heterogeneous documents and close to the similarity between homogeneous documents. The possible reasons for this case are twofold. Trade5 and the coffee documents may have strong semantic relations although being manually classified to different topics, because as we know Reuter collection is economics-related dataset. Furthermore, Wordnet is a general-purposed lexical database and may have limitation to describe the sophisticated semantics between documents in the specialized domains. In summary, although the ontology-based method also has limitation to distinguish the dissimilar documents from the similar ones, it on the average performs well as shown in the two tables. What's more, in this study we propose two hybrid strategies which combine the different similarity measures and can partially overcome the limitation of each method. So the coefficients λ in (6) and δ in (13) need to be properly defined. In the next part the proposed genetic algorithm based on the two hybrid strategies is implemented for clustering.

5.2. The experiments results and analysis of genetic algorithm for text clustering

In this part we use precision P and recall R [19] to evaluate the performance of our clustering algorithm. The performance of the genetic algorithm proposed is then illustrated with the different similarity measures in Table 3. We also compare the self-adaptive genetic algorithm (SAGA) with the standard genetic algorithm [19] and k-means method. The comparison results of precision and recall are shown in Table 4.

Table 3. The performance of SAGA with the various similarity measures

	coffee		trade		crude		sugar	
	P	R	P	R	P	R	P	R
Sim_{VSM}	0.6207	0.7200	0.5600	0.5600	0.5417	0.5200	0.7727	0.6800
Sim_{LSA}	0.6970	0.9200	0.5926	0.6400	0.8667	0.5200	0.8400	0.8400
Strategy 1	0.7240	0.8400	0.7273	0.6400	0.7083	0.6800	0.8400	0.8400
Strategy 2	0.7931	0.9200	0.7917	0.7600	0.7391	0.6800	0.8750	0.8400

Table 4. The performance of SAGA in comparison with the standard GA and k-means

	coffee		trade		crude		sugar	
	P	R	P	R	P	R	P	R
SAGA in Strategy 1	0.724	0.840	0.727	0.640	0.708	0.680	0.840	0.840
SAGA in Strategy 2	0.793	0.920	0.791	0.760	0.739	0.680	0.875	0.840
GA in Strategy 1	0.724	0.840	0.653	0.680	0.750	0.600	0.840	0.840
GA in Strategy 2	0.766	0.920	0.782	0.720	0.727	0.640	0.840	0.840
k-means in Strategy 1	0.450	0.720	0.428	0.360	0.523	0.440	0.833	0.600
k-means in Strategy 2	0.500	0.760	0.500	0.400	0.578	0.440	0.782	0.720

Table 3 illustrates that in strategy 2, SAGA almost obtains the best clustering performance in terms of the results of precision and recall. For topic crude, the precision of the SAGA is only inferior to that of Sim_{LSA} method and exceed that in strategy 1, but the recall in strategy 2 gets the best. We can notice that although in strategy 1 we combine the ontology-based and VSM-based similarity, the results of such method are not apparently better than that of Sim_{LSA} method. The average precision and recall for Sim_{VSM} are 0.6237 and 0.6200, for Sim_{LSA} are 0.7490 and 0.7300, for strategy 1 are 0.7499 and 0.7500, for strategy 2 are 0.7997 and 0.8000, respectively.

SAGA is also compared with the standard GA [19] and k-means approach with the same similarity measurements. We can see from Table 4 that the performances of SAGA with the both strategies are significantly superior to that of k-means for precision and recall. Because it has known that k-means usually suffers from the limitation of the sub optimal. However, in comparison with the standard GA, the effect of our method on the precision and recall is not so significant, because the both genetic algorithms can provide near-optimal solutions. The main purpose of our algorithm is to speed up convergence by our self-adaptive evolution operators.

To sum up, from the experiments results given in Table 3, we can expect that the enhancements for the performance of SAGA are basically by using the strategy 2. What' more, the number of the dimensions used in such strategy for chromosome center encoding is much less than the number of the terms which is used for chromosome center encoding in VSM and strategy 1. In our experiments we set k in (11) as 120 empirically. Thus, strategy 2 decreases the computational complexity. In Table 4, with the same semantic measure strategies, our genetic algorithm is a success method which outperforms the clustering results given by k-means approach. Moreover, the convergence speed of our genetic algorithm is much faster than that of the standard genetic algorithm.

6. Conclusions

In this article we propose an ontology-based genetic algorithm for document clustering. The common problem existing in the fields of text mining and natural language processing is that the documents are simply represented as a string of identical terms, while the conceptual relationship between each pairs of documents is ignored. We take advantage of thesaurus-based and corpus-based semantic similarity measure to overcome this problem. Whereas, in general, the corpus-based method is rather difficult to tackle, a transformed LSA-based corpus model which can appropriately reveal the associated semantic relationship between documents is proposed. Then two hybrid strategies are put forward as the document semantic similarity measures in this study. In our experiments 200 Reuter documents from four topics are selected for test. The results show that our genetic algorithm in conjunction with the strategy 2, the combination of the transformed LSA-based similarity with the ontology-based similarity measure, gets the best clustering performance in terms of the precision and recall. Although the performance of the strategy 1 which combines the VSM-based similarity with the ontology-based similarity measure isn't apparently better than that

of the sole LSA-based method, it also outperforms the traditional cosine similarity in VSM. Further more, the proposed self-adaptive genetic algorithm, considering the influence between the diversity of the population and the selective pressure, efficiently evolve the clustering of the documents in comparison with the standard genetic algorithm and k-means algorithm with the same similarity strategies.

Acknowledgements

This research was partially supported by China Postdoctoral Science Foundation Funded Project (No.20080430463), Youth Research Support Fund funded by Nanjing University of Science & Technology. (No.JGQN0701), Scientific Research Foundation funded by Nanjing University of Science & Technology. (No.AB41123), Korea Research Foundation Grant (KRF-2006-321-A00012) and partially supported by Second Stage of Brain Korea 21.

References

[1] Koontz. W.L.G, Narendra. P.M, and Fukunaga. K, "A Branch and Bound Clustering Algorithm", IEEE Trans on Computers, pp. 908-915, 1975.John. B. Author, and A. Friend, "Journal paper's name", Journal;s name, Vol 39, No. 1, pp. 222-226, Feb. 2001.

[2] Frigui. H, and Krishnapuram. R, "A Robust Competitive Clustering Algorithm with Application in Computer Vision", IEEE Trans, Pattern Analysis and Machine Intelligence, vol. 21, no. 1, pp. 450-465, 1999.

[3] Koontz. W.L.G, Narendra. P.M, and Fucunaga. K, "A Graph Theoretic Approach to Nonparametric Cluster Analysis", IEEE Trans, Comput, C-25, pp. 936-944, 1975.

[4] Selim. S.Z, and Ismail. M.A, "K-means-type Algorithm: Generalized Convergence Theorem and Characterization of Local Optimality", IEEE Trans on Pattern Anal, Intell. 6, pp. 81-87, 1984.

[5] Maulik. U, and Bandyopadhyay. S, "Genetic Algorithm-based Clustering Technique", Pattern Recognition, vol. 33, no.9, pp. 1455-1465, 2000.

[6] Bandyopadhyay. S, Pal. S.K, and Aruna. B, "Multi-objective GAs, Quantitative Indices and Pattern Classification", IEEE Trans on Systems, Man and Cybernetics-B, vol. 34, no. 5, 2004.

[7] Miller. G.A, "WordNet: A lexical Database for English", Comn. ACM, vol. 38, no. 11, pp. 39-41, 1995.

[8] Hotho. A, Staab. S, and Stumme. G, "Wordnet Improves Text Document Clustering", Proc of the Semantic Web Workshop of the 26th Annual International ACM SIGIR Conference, 2003.

[9] Li. Y.H, Bandar. Z.A, and Mclean. D, "An Approach for Measuring Semantic Similarity between Words Using Multiple Information Sources", IEEE Trans on Knowledge and Data Engineering, vol. 15, no. 4, 2003.

[10] Shepard. R.N, "Towards a Universal Law of Generalization for Psychological Science", Science, vol. 237, pp. 1317-1323, 1987.

[11] Resnik. P, "Using Information Content to Evaluate Semantic Similarity in a Taxonomy", Proc. 14th Int'l Joint Conf. Artificial Intelligence, 1995.

[12] Hotho. A, and Stumme. G, "Conceptual Clustering of Text Clusters", proc of FGML Workshop, 2002.

[13] Francis. W.N, and Kucera. H, "Brown Corpus Manual-Revised and Amplified", Dept. of Linguistics, Brown Univ, Providence, R. I, 1997.

[14] Bellegarda. J.R, Butzberger. J.W, and Chow. Y.L, "A Novel Word Clustering Algorithm Based on Latent Semantic Analysis", Proc. ICASSP, pp. 172-175, 1996.

[15] Yao. X, Liu. Y, and Lin. G.M, "Evolutionary Programming Made Faster", IEEE Trans on Evolutionary Computation, vol. 3, no. 2, 1999.

[16] Lee. C.Y, and Yao. X, "Evolutionary Programming Using Mutations Based on the Levy Probability Distribution", IEEE Trans on Evolutionary Computation, vol. 8, no. 1, 2004.

[17] Davies. D.L, and Bouldin. D.W, "A Cluster Separation Measure", IEEE Trans. Patt. Anal. Mach. Intell. 1, pp. 224-227, 1979.

[18] Bandyopadhyay. S, and Mauilk. U, "Nonparametric Genetic Clustering: Comparison of Validity Indices", IEEE Transactions on System, Man and Cybernetics-Part C Applications and Reviews, vol. 31, no. 1, 2001.

[19] Song. W, and Park. S.C, Genetic algorithm-based text clustering technique, LNCS 4221, pp. 779-782, 2006.