의미 벡터 확장을 통한 유전자 클러스터링

Genetic Clustering with Semantic Vector Expansion

쑹웨이, 박순철
전북대학교 컴퓨터공학과

Wei Song(songwei@chonbuk.ac.kr), Soon-Cheol Park(scpark@chonuk.ac.kr)

요약

본 논문에서는 퍼지 논리 기반의 유전자 알고리즘(GA)과 의미 벡터 확장 기술을 이용한 문서 클러스터링 시스템을 제안한다. GA에 관련된 여러 논문에서 이미 알려졌듯이 GA알고리즘의 성공 여부는 군체의 다양성과 수렴하는 능력에 따라 결정된다. 이러한 두 인자 사이의 영향력을 조절하기 위하여 우리는 퍼지 논리 기반의 연산자를 사용한다. 전통적인 문서 클러스터링 알고리즘에서 문서를 나타내기 위한 가장 일반적이고 직선적인 방법은 벡터 공간 모델이다. 그러나 이 방법은 다차원 특징 공간의 원인이 될 뿐만 아니라, 클러스터링의 정확성에 영향을 미칠 수 있는, 단어 간의 의미상 관계성을 무시한다. 본 논문에서는 LSA를 사용하여 문서를 관련되는 의미상의 벡터 개념으로 확장시킨다. 또한 이것은 벡터의 크기를 크게 줄일 수 있다. 본 논문에서 제안한 클러스터링 알고리즘을 테스트하기 위하여 20개의 뉴스 그룹과 로이터 데이터를 사용했다. 제안된 방법은 문서를 표현하는 다양한 환경에서 일반적인 GA보다 더 나은 결과를 보여준다.

■ 중심어 : | 클러스터링 | 유전자 알고리즘 | 의미 벡터 확장 | LSA |

Abstract

This paper proposes a new document clustering system using fuzzy logic-based genetic algorithm (GA) and semantic vector expansion technology. It has been known in many GA papers that the success depends on two factors, the diversity of the population and the capability to convergence. We use the fuzzy logic-based operators to adaptively adjust the influence between these two factors. In traditional document clustering, the most popular and straightforward approach to represent the document is vector space model (VSM). However, this approach not only leads to a high dimensional feature space, but also ignores the semantic relationships between some important words, which would affect the accuracy of clustering. In this paper we use latent semantic analysis (LSA)to expand the documents to corresponding semantic vectors conceptually, rather than the individual terms. Meanwhile, the sizes of the vectors can be reduced drastically. We test our clustering algorithm on 20 news groups and Reuter collection data sets. The results show that our method outperforms the conventional GA in various document representation environments.

■ keyword : | Clustering | Genetic Algorithm | Latent Semantic Analysis(LSA) | Semantic Vector Expansion |

접수번호 : #090121-002
접수일자 : 2009년 01월 21일
심사완료일 : 2009년 03월 03일
교신저자 : 박순철, e-mail : scpark@chonuk.ac.kr

I. Introduction

The rapid prevalence of web applications and the explosive growth of knowledge require more efficient technologies for the design, implementation and management of web-based information systems. As one of the hotspots and key techniques in web applications, there have been extensive studies and obvious progress in automatic document clustering. Clustering is a popular unsupervised classification technique which partitions a set of data points into groups such that similar points belong to the same group, and dissimilar points belong to different groups. Some clustering techniques that are available in literature are K-means algorithm [1], OPTICS [2], Single-Link method [3] and graph theoretic approach [4]. K-means algorithm, one of the most widely used, attempts to create K partitions of the points. However, it suffers from the limitation of the local optimal solution which depends on the choice of initial centers distribution [1]. OPTICS or Single-Link method compute a representation of the possible hierarchical clustering structure of the database in the form of a reach ability plot from which clusters at various resolutions can be extracted. Koontz et al. propose a graph theoretic approach [4]. The clustering is realized by finding the valley of the density function. The quality of the result relies on the quality of the estimation technique for the density gradient. Genetic algorithm (GA) [5] is a robust probabilistic search and optimization technique directed by the guidelines of natural genetics and evolution principles. In this paper we propose a novel fuzzy logic-based genetic algorithm (FLGA) which exerts several control parameters to manipulate the operators of GA, i.e. selection, crossover, and mutation. FLGA estimates the direction of the evolution and can effectively avoid convergence to a local optimal solution.

In order to perform document clustering, an important step is text representation. Vector space model (VSM) one of the widely used, computes a measure of similarity by defining a vector that represents each document. However, VSM has many drawbacks, because the inherent high dimensions with a large number of terms are prone to cause an over fitting problem. Moreover, if we represent all texts by this way the ambiguity meaning of terms may prohibit identifying the semantic closeness between vectors. Latent semantic analysis (LSA) [6][7] is an automatic method that projects this large space into a space with semantic dimensions and filters out the noise found in the documents. So in this semantic structure two documents which have the same semantics are located close to one another because the similar contexts in the documents will have similar vectors in semantic vector space. We use LSA to conceptually expand documents to relative semantic vectors which can provide robust semantic relationships between them. What is more, it drastically reduces the dimensions which are very suitable for clustering computing.

II. Fuzzy logic-based genetic clustering

In genetic algorithm a random distributed population is created first. Each individual in the population is encoded in the form of strings, called chromosomes. A fitness function, corresponding to each chromosome, represents the degree of fitness. Biologically inspired operators, such as selection, crossover and mutation, are exerted to yield new offspring. These operators continue several generations until the termination criterion is satisfied.

In early versions of GAs, the parameters p_c , p_m

are constant or they are simply adjusted by some analytical functions. So a large population is used to prevent a premature convergence. However, this would cause a high computation cost. Meanwhile, according to the analysis of evolving process, these two parameters affect largely the behaviors of GA, i.e. the capability to converge to an optimum and the capability to explore new regions of the solution space. Thus, they need to be properly defined by users or they are automatically adjusted in accordance with the nature of evolution. In this paper we propose several control parameters which can adaptively manipulate the crossover probability p_c and the mutation probability p_m. We firstly define the evolution effect E_i from the previous generation and the current generation to depict the evolution trend for individual X_i, that is,

$$E_i = e^{fit_i(g)-fit_i(g-1)}, \quad (1)$$

where g is the number of generations. When $fit_i(g)-fit_i(g-1)>0$, the individual X_i makes a progress in the current generation. Otherwise, the individual X_i is retrogressive or remains the same. We use $e^{fit_i(g)-fit_i(g-1)}$ to restrict the effect of the case ($fit_i(g)-fit_i(g-1)\le 0$) in the interval [0, 1]. That is, we limit the effect of the later case as a small positive number and stress the effect of the former case which can do better in denoting the development level. The mean evolution effect for current generation is given by:

$$E_m = \frac{1}{p}\sum_{i=1}^{p} e^{fit_i(g)-fit_i(g-1)}, \quad (2)$$

where p is the number of the individuals in population. The mean evolution effect is used to denote the direction of evolution. If it is a big progress, we know that the current generation contains more numbers of excellent individuals, and we can moderately decrease the diversity of the population to maintain these excellent individuals. If it is a small progress or a medium state, we can slightly adjust the diversity of population, and if it is retrogressive, we need to greatly expand the diversity of population by enhancing the probability of crossover and mutation. The parameter α is given by $\max\{e^{\overline{fit}(g)-\overline{fit}(g-1)}, \varepsilon\}$, where ε is empirically set based on the fitness function itself. The value of the evolution rate δ is defined in [Table 1].

Table 1. The definition of evolution rate δ

Development	Evolution effect	Evolution rate
Progressive	$E_m \ge \alpha$	δ =0.80
Medium	$1 \le E_m < \alpha$	δ =1.15
Retrogressive	$0 < E_m < 1$	δ =1.30

In order to vary p_c and p_m adaptively, for preventing premature convergence of GA to a local optimum, it is essential to be able to identify whether the GA is converging to an optimum. We use Var to depict the distribution of the population in the current generation, that is,

$$Var = (fit_{\max} - \overline{fit})/(fit_{\max} - fit_{\min}), \quad (3)$$

where $fit_{\max}$, $\overline{fit}$ and $fit_{\min}$ represent the maximum, the average and the minimum value of the fitness, respectively. Var depicts the closeness of the distribution in the current generation. Var is likely to be less for the population that has converged to an optimal solution than that for a population scattered in the solution space. We here empirically divide the value of Var into three intervals according to its

mathematic definition:

Definition 1: if $0 < Var \le \frac{1}{4}$, the average distribution of all individuals are close to the best one and distant from the worst one. This case may cause a premature convergence.

Definition 2: if $\frac{1}{4} < Var \le \frac{3}{4}$, an extensive diversity of distribution exists in the current population.

Definition 3: if $\frac{3}{4} < Var < 1$, the average distribution of all individuals is very distant from the best individual although it seldom occurs.

The relative distance between the fitness of the individual X_i and the best fitness is defined by:

$$G = (fit_{\max} - fit(X_i)) / (fit_{\max} - fit_{\min}) . \quad (4)$$

The parameters Var and G are defined in the interval [0, 1]. Var is determined by the distribution of all individuals in the current population and G is given by a specific individual.

In order to optimize the behaviors of GA, we adopt several rules to guide the crossover and mutation appropriately. The rules include:

Rule 1: To maintain the diversity of each population, the $p_c(X_i)$ of the remote individual X_i needs to be enhanced in next generation.

Rule 2: To maintain the excellent individuals in the population, if the fitness of the individual X_i is close to the best fitness, the $p_c(X_i)$ will decrease in next generation.

Rule 3: If the evolution process tends to a local optimum, we will enhance $p_m(X_i)$ to prevent convergence to a local optimal solution.

Rule 4: If the current generation has extensive diversity, we need to decrease $p_m(X_i)$.

Rule 5: At the beginning of the evolution an extensive crossover is used to make a wide-ranging exploration and then the crossover is decreased to make a high quality exploration.

From the rules above, we use a series of integers to depict the approximate trend of p_c and p_m in the next generation, where positive sign means enhancement, negative sign means reduction, and the their absolute value represents the magnitude. In [Table 2], x and y represent the intensity of p_c and p_m respectively.

From [Table 2] we can see that if $G < Var$ ($G < Var$ and $0 < Var \le \frac{1}{4}$), that is, the fitness of the individual X_i is close to the best fitness and the excellent individual X_i has high probability to be maintained in the next generation. So $p_c(X_i)$ is decreased although Var is small. However, this case may cause a local optimum ($0 < Var \le \frac{1}{4}$). Thus, the value of $p_m(X_i)$ is slightly increased. If $\frac{1}{4} < Var \le \frac{3}{4}$, that is, an extensive diversity exists in the current population, we need to decrease the $p_m(X_i)$ although $G \ge Var$. However, the $p_c(X_i)$ of the remote individual X_i ($G \ge Var$) needs to be slightly increased. The case $\frac{3}{4} < Var < 1$ means the average distribution of all individuals is very distant from the best individual. So we need to enhance the probability of selection p_s although this case seldom occurs.

Table 2. The intensity of p_c and p_m

Var	G	x	y
$0 < Var \le \frac{1}{4}$	$G \ge Var$	+2	+2
	$G < Var$	−1	+1
$\frac{1}{4} < Var \le \frac{3}{4}$	$G \ge Var$	+1	−1
	$G < Var$	−2	−2
$\frac{3}{4} < Var < 1$	Enhance the probability of selection		

Two monotonically increased functions are used to generate u and v which affects p_c and p_m in the next generation. Parameters u and v are defined as

$$u = 1 + k_1 x \quad x \in \{2,1,-1,-2\}; k_1 = 0.2; \tag{5}$$

$$v = 1 + k_2 y \quad y \in \{2,1,-1,-2\}; k_2 = 0.1; \tag{6}$$

The crossover probability p_c determines the rate at which individuals are subjected to crossover. The higher the value of p_c, the quicker are the new individuals introduced into the population. However, the individuals may be disrupted faster than selection exploits them if a high p_c is applied. Mutation is just a secondary operator to restore evolving material. So here we define a moderately small k_1 and a small k_2 to plot out the magnitude for p_c and p_m in the different situations. Moreover, we have defined the evolution rate δ which is able to adjust p_c and p_m in general. Hence, in the next generation the crossover probability $p_c'(X_i)$ and the mutation probability $p_m'(X_i)$ for individual X_i are given by:

$$p_c'(X_i) = \delta \mu \, p_c(X_i), \tag{7}$$

$$p_m'(X_i) = \delta \nu \, p_m(X_i), \tag{8}$$

where δ is the evolution rate given in [Table 1]. We use double-point crossover in the primary 200 generations to make a strong exploration and then the classical single-point crossover is used in the subsequent generations.

Ⅲ. LSA for semantic vector expansion

Latent semantic analysis (LSA) is a well developed method which projects the high dimensional document vectors into a space with latent semantic dimensions. Singular value decomposition (SVD) is a mathematical concept, which is commonly used in most of the latent semantic analysis methods.

The original corpus can be initially represented as a document-by-term matrix $D(n \times m)$, where n is the number of documents and m is the number of terms. The transpose of matrix D is the term-by-document matrix $A(m \times n)$, and the singular value decomposition of A is defined as:

$$A = U \Sigma V^T, \tag{9}$$

where U is an $m \times m$ orthogonal matrix whose columns define the left singular vectors of A, V^T is an $n \times n$ orthogonal matrix whose rows define the right singular vectors of A, and Σ is an $m \times n$ diagonal matrix containing the singular values $\lambda_1 \ge \lambda_2 \ge \mathrm{L} \ge \lambda_n$. It has known that the number of terms is much more than the number of documents, that is m is much greater than n. Thus we only use matrix U_1 for the new corpus matrix C construction. The size of the U_1 is the economy size of the matrix U, depending on the number of

the nonzero singular values in matrix $\Sigma(m \times n)$.

Thus, the new corpus matrix C is defined by:

$$\underset{n \times n}{C} = \underset{n \times m}{D}\ \underset{m \times n}{U_1}. \quad (10)$$

We can use a row of elements in C to represent the relative document. That is, the number of dimensions is decreased from m for the original document to n where n is much smaller than m.

In order to further reduce the dimensions with latent semantic analysis, we can simply choose the k ($k < n$) largest singular values and the corresponding left and right singular vectors. The approximation matrix A_k is given by:

$$A_k = U_k \Sigma_k V_k^T, \quad (11)$$

where U_k is comprised of the first k columns of the matrix U and V_k^T is comprised the first k rows of the matrix V^T. $\Sigma_k = diag(\lambda_1, \lambda_2, \mathrm{L}\ , \lambda_k)$ is the diagonal matrix comprised of the first k factors. LSA is firstly proposed in query-based information retrieval system [6]. Nowadays, it is also widely adopted in clustering [7] and classification [8]. We propose a modified LSA technique in our systems. Our LSA-based corpus C_k is given by:

$$\underset{n \times k}{C_k} = \underset{n \times m}{D}\ \underset{m \times k}{U_{1k}}, \quad (12)$$

where U_{1k} is the first k columns of the matrix U_1. The semantic vector for each document d is newly formed by:

$$\underset{1 \times k}{\hat{d}} = \underset{1 \times m}{d^T}\ \underset{m \times k}{U_{1k}}, \quad (13)$$

where the number of dimensions for vector $\hat{d}$ is k ($k < n$), which is much smaller than that of the original document in VSM with dimensions m. Once the relative semantic vector is defined by this way, the similarity between them can be computed using cosine measure.

Ⅳ. Experiments results

In this section we implement our method of FLGA for text clustering on 20-newsgroup corpus (18828 version) and Reuter-21578 collection, which are the most-widely adopted benchmark data sets in text mining field. Data set 1 containing 200 documents from four 20-newsgroup topics and data set 2 containing 600 documents from six Reuter topics are selected. After preprocessing, there are 7117 terms and 5870 terms for data set1 and data set 2, respectively, in the vocabulary. It has been well established in GA literature that moderately large values of p_c and small values of p_m are commonly employed in GA practice. In our experiment, the parameters p_c and p_m for each individual are initially set as 0.3 and 0.05 respectively. The initial p_s is 0.3 and we restrict the maximum value of p_s to be 0.6.

The main difficulty in the application of genetic algorithm to document clustering is the high dimensional feature space. In our experiment we decrease the number of terms from 7117 to 1500 for data set 1 and from 5870 to 1000 for data set 2, respectively, by choosing the highest term weights to construct the original document-by-term matrix D. The formula to calculate the term weight is given by:

$$W_{ij} = f_{ij} / (tf_{ij} + 0.5 + 1.5 \times (dl / avgdl)) \times idf_j, \quad (14)$$

where $idf_j = \log(N / n)$, N is the total number of documents in the data sets, and n is the

number of documents in which the i^{th} term comprised. tf_{ij} is the term frequency of i^{th} indexing term in document j, dl is the length of document. This formula normalizes the length of documents rather than the simple $tf \times idf$. Then we use the LSA method proposed to further decrease the dimensions in the semantic vector space. We compare the performances by varying the number of the dimensions k in C_k (12) from 200, 180, 160, 140, 120, 100, 90, 80, 70, 60, 50 to 40 for data set 1 and from 600, 550, 500, 450, 400, 350, 300, 250, 200, 180, 160, 140, 120, 100,90, 80,70, 60, 50 to 40 for data set 2 respectively.

We use F-measure to evaluate the performance of our clustering algorithm. [Figure 1] and [Figure 2] show the performances of FLGA with the different ranks k. We also compare FLGA with the conventional GA.

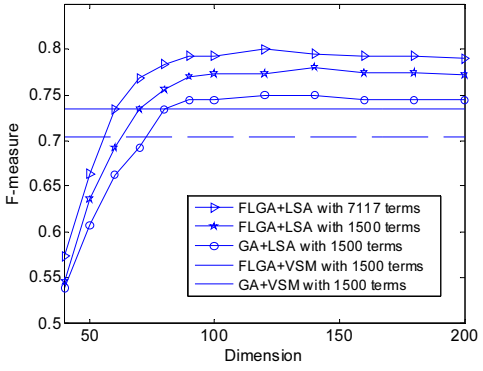

Figure 1. The clustering performance for Dataset 1

Form [Figure 1] we can see that the F-measure of GA with 1500 terms in VSM is 0.704. The F-measure of FLGA with 1500 terms in VSM is 0.735 which is better than that of GA in VSM. The performances of FLGA(s) are further enhanced in the semantic space. From 70 dimensions the F-measures of FLGA(s), with 7117 and 1500 terms in LSA model, outperform that of FLGA in VSM. For 7117 terms in LSA, FLGA obtain its best performance on 120 dimensions with value 0.800. For 1500 terms in LSA, FLGA obtains its best performance on 140 dimensions with value 0.783. When the dimension is close to the 200, the performances of FLGA(s) in LSA model are slightly decreased in comparison with their best performances.

From [Figure 2] we can see that from about 70 dimensions the performance of FLGA with 1000 terms in LSA outperforms that of GA in VSM, and from 80 dimensions it outperforms that of FLGA in VSM. For 5870 terms in LSA model, FLGA obtain its best performance on 300 dimensions. For 1000 terms in LSA, FLGA obtains its best performance on 350 dimensions. The best performances of FLGA(s) with 5870 and 1000 terms in LSA model are 0.792 and 0.765 respectively.

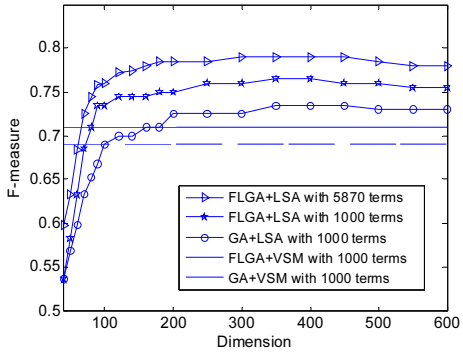

Figure 2. The clustering performance for Dataset 2

V. Conclusions

In this paper we develop a text clustering method using the fuzzy logic-based genetic algorithm (GA)

and the semantic vector expansion technology. The introduced semantic vector expansion takes advantage of latent semantic analysis method which not only reduces the dimension drastically, but also overcomes the problem existing in commonly used vector space model for text clustering. The fuzzy logic-based GA solves the problem of slow convergence existing in conventional GA and is able to escape from the trap of the local optimal solution. The experiment results show that the fuzzy logic-based GA enhances the performance of clustering, and the semantic vector expansion technique further improves its accuracy and efficiency. In the future we will refine our clustering program and speed up its running.

참 고 문 헌

[1] S. Selim and M. Ismail, "K-means-type algorithm: generalized convergence theorem and characterization of local optimality," IEEE Trans. Pattern Anal. Mach Intell. 6, pp.81-87, 1994.

[2] M. Ankerst, M. Breuing, and H. P. Kriegel, "OPTICS: Ordeing points to identify the clustering structure," In Proceedings of SIGMOD'99, pp.49-60, 1999.

[3] R. Sibson, "SLINK: An optimally efficient algorithm for the single-link cluster method," The Computer Journal, Vol.16, No.1, pp.30-34, 1973.

[4] W. Koontz, P. Narendra, and K. Fucunaga, "A graph theoretic approach to nonparametric cluster analysis," IEEE Trans. Comput, C-25, pp.936-944, 1975.

[5] S. Bandyopadhyay and S. K. Pal, "Multi-objective GAs, quantitative indices and pattern classification," IEEE Trans. Systems, Man and Cybernetics-B, Vol.34, No.5, pp.2088-2099, 2004.

[6] M. W. Berry, S. T. Dumais, and G. W. Brien, "Using linear algebra for intelligent information retrieval," SIAM Rev, Vol.37, No.4, pp.573-595, 1995.

[7] J. T. Sun, Z. Chen, and H. J. Zeng, "Supervised latent semantic indexing for document categorization," In Proceedings of ICDM'04, pp.535 - 538, 2004.

[8] M. G. Vozalis and K. G. Margaritis, "Using SVD and demographic data for the enhancement of generalized collaborative filtering," Information Sciences, 177, pp.3017-3037, 2007.

저 자 소 개

쑹 웨 이(Wei Song) 정회원

- 2004년 6월 : South-Central University for Nationalities, Computer Science and Technology (공학사)
- 2006년 8월 : 전북대학교 컴퓨터공학 (공학석사)

<관심분야> : 정보검색, 내용기반 문서 클러스터링, Neural Network

박 순 철(Soon-Cheol Park) 정회원

- 1979년 6월 : 인하대학교 학사
- 1991년 : Louisiana State University
- 1993년 : 한국전자통신연구원 (ETRI)
- 현재 : 전북대학교 전자정보공학부 교수

<관심분야> : 정보검색, 내용기반 문서 클러스터링, Neural Network