TELKOMNIKA, Vol.12, No.4, December 2014, pp. 1045~1052
ISSN: 1693-6930, accredited A by DIKTI, Decree No: 58/DIKTI/Kep/2013
DOI: 10.12928/TELKOMNIKA.v12i4.811

Process Improvement of LSA for Semantic Relatedness Computing

Wujian Yang*[1], Lianyue Lin[2]
Department of compute and science, Zhejiang University City College
No. 48 Huzhou Road, Hangzhou, Zhejiang, China 310000
*Corresponding author, e-mail: yangwj@zucc.edu.cn[1], linlydtc@gmail.com[2]

Abstract

Tang poetry semantic correlation computing is critical in many applications, such as searching, clustering, automatic generation of poetry and so on. Aiming to increase computing efficiency and accuracy of semantic relatedness, we improved the process of latent semantic analysis (LSA). In this paper, we adopted "representation of words semantic" instead of "words-by-poems" to represent the words semantic, which based on the finding that words having similar distribution in poetry categories are almost always semantically related. Meanwhile, we designed experiment which obtained segmentation words from more than 40000 poems, and computed relatedness by cosine value which calculated from decomposed co-occurrence matrix with Singular Value Decomposition (SVD) method. The experimental result shows that this method is good to analyze semantic and emotional relatedness of words in Tang poetry. We can find associated words and the relevance of poetry categories by matrix manipulation of the decomposing matrices as well.

***Keyword**: semantic relatedness, Latent Semantic Analysis, poetry category, singular value decomposition*

1. Introduction

Tang poetry, as a kind of Chinese classical literature, has a profound impact on China and even the world. Compared with the modern Chinese, ancient poetry, which conveyed all sorts of emotions by refined words, rhythmic syllables and various figures of speech, has special syntax. However, automatic analysis of ancient poetry, a part of Natural Language Processing (NLP) [1], has already been a hot issue, which involves various essential tasks, e.g., establishment of corpus [2], word segmentation [3], semantic analysis [4], vector space model [5], identification of poetry style [6], etc.

The ability to quantify semantic relatedness of words in poems should be an integral part of semantic analysis, and underlies many fundamental tasks in NLP, including information retrieval, word sense disambiguation, and text clustering, etc. In contrast to semantic similarity, which is the special case of relatedness, the notion of relatedness is more general than that of similarity like Budanitsky et al [7] argued, as the latter subsumes many different kind of specific relations, including metonymy, antonym, functional association, and others. In this paper we deal with semantic relatedness.

Semantic relatedness computing of natural language texts requires encoding vast amount of world knowledge. Until recently, prior work of linguistic resources using pursued two main directions. One is lexical databases such as WordNet [8], Wikipedia [9], encodes relations between words such as synonymy, hypernymy, and the other is large-scale text corpora, provide statistical corpus for computer learning like Latent Semantic Analysis (LSA) [10].

But in general computing of modern language semantic relatedness, the least resources used are knowledge-free approaches that rely exclusively on the corpus data themselves. Under the corpus-based approach, word relationships are often derived from their co-occurrence distribution in a corpus [11]. With the introduction of machine readable dictionaries, lexicons, thesauri, and taxonomies, these manually built pseudo-knowledge bases provide a natural framework for organizing words or concepts into a semantic space. Kozima and Furugori [12] measured word distance by adaptive scaling of a vector space generated from LDOCE (Longman Dictionary of Contemporary English). Morris and Hirst [13] used Roget's thesaurus to detect word semantic relationships. With the recently developed lexical taxonomy WordNet [14], many researches have taken the advantage of this broad-coverage taxonomy to study word/concept relationships [15].

Received August 26, 2014; Revised October 14, 2014; Accepted November 2, 2014

However, Chinese language used in the ancient times was quite different from modern Chinese language and the nature of poetry. So, for tang poetry's special syntax and word limitation, present semantic relatedness computing of Tang poetry based on large-scale poetry corpora and word co-occurrence. HU, J. and YU, S. [16] defined a statistic model to extract contextual similarity words from the corpus based on 614 million chars of Chinese ancient poetry. Zhou, C. L. [17] computes words semantic relatedness by combining methods of latent semantic analysis (LSA) and mutual information (MI), which is general method in words relatedness of Chinese poetry.

Latent Semantic Analysis (LSA), a new algebraic model of information retrieval, is proposed by S.T. Dumais in 1988. Thus, it's a purely statistical technique, leverages word co-occurrence information from a large unlabeled corpus of text. Based on an assumption that each word's meaning in tang poems can be represented by its co-occurrence words for their regular co-occurrence in large-scale texts corpora, LSA does not rely on any human-organized knowledge; rather, it "learns" its representation by applying SVD to the words-by-poems co-occurrence matrix. So, we can imagine that how tremendous the size of this matrix is. In order to reduce the matrix for efficient and rapid computing, we proposed a method of process improvement, which aim to build a "words-by-poetry categories" co-occurrence matrix. In this paper, we mainly discuss about LSA, propose our process improvement based on the previous method as well.

The contributions of this paper are threefold. First, we propose to classify the poems by emotions, and then build a "words–by-poetry categories" co-occurrence matrix. Specifically, we introduce an improved method of LSA using in semantic relatedness computing of words in Tang poems. Second, we structure a matrix, which represent poems in corpus with much smaller size than previous method, and present specific methods for applying Single Value Decomposition (SVD) efficiently and rapidly. Finally, we propose the applications of result computed by this improved method.

2. The Process Improvement of Relatedness Computing Method

Study the previous method to compute relatedness with LSA, We can find that it use information between words and poems. Thought association of words and poems can be measured, but complicated statistics shows up in the following aspects.

Firstly, it needs large-scale non-repetitive text corpora and the workloads of the collection and entry is heavy. Secondly, for the row vector is word vector, the preparatory work of segmentation and statistic are complicated, which should split and count words for each poem in corpus individually, and weights the frequency that each word appears in each poem as element of vector.Thirdly there are a mass of words, and the number of poems is vast as well, which makes matrix to be huge and have a strong impact on efficiency of operation. Finally, it can't prevent the zero angles of vectors caused by sparseness problem, and the number of sparse word is still a lot.

These first three factors above make the matrix large which cause lower computational efficiency. And the final sparseness problem influences the accuracy of calculation. In response to this semantic relatedness computing, we come up with a method of process improvement (as Figure 1), which is composed of two novel components: a new pattern for representing semantic of words in poems, and a new method for computing the semantic relatedness between words, even association between poetry categories.

The mainly improvement in this paper is that we use representation of words' semantic by "words-by-poetry categories" instead of "words- by-poems". Our hypothesis is that words which behave almost similarly in poetry categories are semantically related. For simple syntax of Tang poetry, the mainly feeling of poet is expressed by unified emotional words in poem.

Thus, our improved method consists of three main steps. First, construct matrix, the representation of words' semantic. Second, decompose matrix into three significant matrices by SVD. Finally, analyze the matrices and compute relatedness.

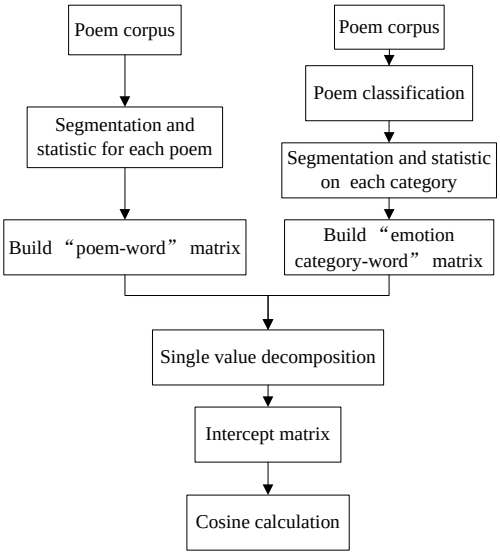

Figure 1. Previous method and process improvement method

2.1 Construction of the Matrix

In improved representation, each word is mapped into a vector of corresponding row in matrix, and each column vector of matrix represents a poetry category. So, prior work fell into three steps. First, we can divide the set of Tang poetry into several categories, which is clarified by emotion. Second, split words and compute the time that each word appears in category for each category individually. Finally, we can build the matrix.

Specifically, the set of poetry can be divided into N categories, which have M words from segmentation. Thereby, we can build "words-by-poetry categories" matrix with size of M × N. The frequency that word i in poetry category of j can be expressed as fij, and the weight of word can be expressed as aij, so the matrix can be expressed as $A = [a_{ij}]$.

2.2 Singular Value Decomposition

In the construction of the matrix, we constructed the matrix to represent semantic of words in tang poetry, where, is local weight of the word i in the category j, g is global weight of the word i in the full texts (Tang poetry set). Because of a practical theorem that there must exist singular value decomposition (SVD) [18,19] for any nonzero matrix, so in the realization of LSA, we use a typical construction method of LSA / SVD, which based on building matrix space model to do the calculation of singular value decomposition.

2.2.1 Theorem

Let $A \in R_r^{m \times n}[C_r^{m \times n}]$. Then there exist orthogonal unitary matrices $U \in R^{m \times n}[C^{m \times n}]$ and $V \in R^{m \times n}[C^{m \times n}]$ such as that

$$A = U \Sigma V^T [U \Sigma V^H]$$

where

$$\Sigma = \begin{pmatrix} S & 0 \\ 0 & 0 \end{pmatrix}$$

and $S = diag[\sigma_1, \sigma_2, ..., \sigma_r]$ with $\sigma_1 \geq ... \geq \sigma_r > 0$.

2.2.2 Main algorithms of singular value decomposition

Algorithms I

- Find the eigenvalues and eigenvectors of *A'A* as
  $A'A = V\mathrm{diag}(\Sigma_r^2, O)\, V'$, $V = (V_1, V_2)$
  V_1 is an orthogonal matrix with size of m × r;
- Compute $U_1 = AV_1\Sigma^{-1}$, U_1 is an orthogonal matrix with size of n × r.
- Extended U_1, then get an orthogonal matrix with size of n × n: $U_{m\times m} = (U_1, U_2)$
- Finally, we can find $A_{m\times n} = U_{m\times m}\Sigma_{m\times n}V'_{n\times n}$

Algorithms II

- Transfer A into a pair of diagonal with Householder method, which means that there exist n × n orthogonal matrix P and m × m orthogonal matrix Q , which can realize:

$$PAQ = \begin{pmatrix} E \\ O \end{pmatrix}$$

  Where E is Doubly Diagonally Matrix (other than the elements in diagonal and secondary-diagonal, the remaining are zero).
- Using deformation of QR method and iterative computation reduce the elements of secondary-diagonal of E to zero gradually, so realize the diagonalization of matrix, and finally get result of SVD. Deformation of QR method is based on the approach of QR decomposition to find general matrix eigenvalues and eigenvectors.

2.3 Analysis of intercept matrix

LSA is essentially a dimensionality reduction technique that allow to obtain the largest singular value k (k ≤ r ≤ n <m) to reduce the size of *U*, *V*. The matrix A_k, product of matrices $U_{m\times k}$, $D_{k\times k}$ *and* $V_k'_{\times n}$, is generate from the interception, can represent the original matrix approximately as shown in follow.

From that we can get three smaller matrices with clear meaning, $U_{m\times k}$ shows the relative feature between the words, $V'_{k\times n}$ shows the poetry classes as well, and the middle matrix shows the importance of different columns in $U_{m\times k}$.

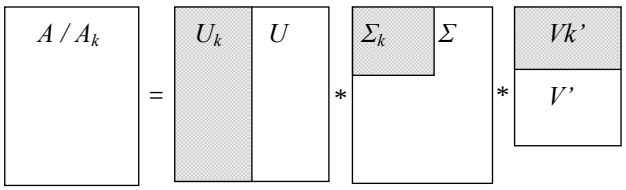

Figure 2. Matrix after Intercept

So, we can calculate the relatedness of two words by corresponding rows in product of $U_{m\times k}$ *and* $\Sigma_{k\times k}$, which can be expressed as $U_{m\times k}\times\Sigma_{k\times k}$, and the value is represented by cosine of corresponding vectors. The more cosine close to 1, the higher correlation is. On the contrary, the more cosine close to 0, the less correlation is. Similarly, we can calculate relatedness between two classes by every two columns in $V'_{k\times n}$. The more cosine close to 0, the less correlation between categories and the poetry division is more reasonable.

According to this law, we can do some judgment of relatedness as follow. One is about poetry classification, which can give reference to find out whether category division is scientific or not. If the relatedness between two certain categories is higher, so for these two categories probably is being classified basing on the different facts, which will break the rationality of classification. Such as “separation” and “sadness”, they are different kinds which are classified on the different facts, because almost every separation poem is sad. The other is relatedness of words, which compute by the cosine calculating of row vectors in $U_{m\times k}\times\Sigma_{k\times k}$ matrix, can

determine the strength of semantic relatedness between words, provide the basis for establishment of words related libraries of emotions, and emotions unified judgment of automatically generated Tang poetry.

3. Experimental Procedure and Analysis of Results

In this section we implement our improved method in our experiments, provides the experimental result, and compare empirical evidence that contribute to measuring semantic relatedness of words to previous method. Finally, we analyse matrixes come from singular value decomposition of original matrix.

We implemented our LSA approach using more than 40,000 poems as local poetry corpus at the beginning. Then divide poems into 11 categories based on knowledge of literature, includes patriotic, war, farewell, seasons, weather, love, plant, nostalgia, festivals, rural reclusion, landscape. Each poem can be divided into several different categories for its emotion diversity.

3.1 Segmentation and statistics

Classification, we can implement segmentation, statistics, and frequency weighting for each category, and get word weights at last. The Table 1 shows a part of weighting result of farewell category.

Table 1. Segmantation and Statistis Result of Categories

No	Word	frequency	weight
1	a thousand li	16	127.04
2	white cloud	10	82.90
3	Qingshan	8	76.56
4	spring breeze	12	66.48
5	old friend	10	66.30
6	Miles	7	62.30
7	Where	11	57.64
8	Willows	8	53.92

We compare our representations and statistics of segmentation to previous approach, which reduce the time of implementation like this work, find it has been shown to be significantly superior to other approaches. For hypothesis that words which behave almost similarly in poetry categories are semantically related, we just do this work for each category instead of each poem individually. So, the time of segmentation and statistics can be reduced from the number of poems to the number of category. The weight of each word corresponds to the word's importance to each emotional category.

Table 2. Construction of the Frequency Matrix

	patriotic	War	Farewells	season	weather	love	plant	nostalgia	festival	Hermit	landscape
Homeless	28.9200	0.0000	14.4600	0.0000	14.4600	0.0000	0.0000	14.4600	14.4600	0.0000	0.0000
Han Dynasty	43.5000	130.5000	14.5000	14.5000	0.0000	0.0000	14.5000	14.5000	14.5000	0.0000	0.0000
Yinshan	42.7800	71.3000	0.0000	0.0000	14.2600	0.0000	0.0000	14.2600	0.0000	0.0000	0.0000
Miles	34.3700	4.9100	62.3000	24.5500	4.9100	24.5500	14.7300	0.0000	9.8200	9.8200	9.8200
Chang an	34.0000	17.0000	34.0000	8.5000	25.5000	25.5000	17.0000	42.5000	8.5000	0.0000	8.5000
White jade	33.6900	0.0000	11.2300	0.0000	0.0000	0.0000	33.6900	11.2300	0.0000	0.0000	0.0000
Xianyang	32.5800	0.0000	10.8600	0.0000	10.8600	0.0000	0.0000	21.7200	0.0000	0.0000	0.0000
Loyalty	31.5200	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	7.8800	0.0000	0.0000
Peking	29.9400	0.0000	9.9800	0.0000	9.9800	0.0000	0.0000	0.0000	19.9600	9.9800	0.0000
Inch	29.5600	0.0000	0.0000	0.0000	0.0000	7.3900	0.0000	0.0000	0.0000	0.0000	0.0000
gray hair	26.9000	0.0000	0.0000	10.7600	5.3800	16.1399	0.0000	5.3800	16.1399	0.0000	10.7600
Gold	19.2000	9.6000	24.0000	14.4000	0.0000	4.8000	9.6000	19.2000	9.6000	4.8000	0.0000

3.2 Matrix construction Analysis

After segmentation and weighting, we merge duplicate of words and build the matrix as Table 2 showing. In order to avoid the influence of sparse words on the calculation accuracy, we excluded some word with both lower weight and frequency.

In order to improve the accuracy of relatedness computing of words show in several categories, we excluded words only appear in one category. Finally, we build a 2889 * 11 matrix of high-frequency words, which is much less than previous matrix.

We compare the result of matrix construction to previous approach, this improved method reduces the number of column sharply, and the size of matrix decrease from "words × poems" to "word × categories". However, whenever we add poems to local corpus in previous approach, the size of this matrix will increase in the meantime, but it won't in improved method.

3.3 Analysis of Matrices after SVD

Matrix construction is the base to the next work. As we introduced in the second section, after matrix construction, we can decompose matrix by single value decomposition into 3 matrixes:

$$\underset{t\times d}{X} = \underset{t\times r}{T_0}\ \Sigma_0\ \underset{r\times d}{V_0'}$$

With the algorithms introduced in third section we decompose matrix into T, Σ, V matrices as Table 3 shows (It's only a part of matrix).

Table 3. Matrices of U, Σ, V

Matrix of *U*				
0.0442	-0.0451	0.1833	-0.0158	-0.0445
0.0792	-0.0838	0.2695	-0.0278	-0.1279
0.0428	-0.0726	0.1698	-0.0056	-0.0574
0.0576	0.0340	0.0062	-0.0100	0.0009
0.0842	-0.0523	0.0053	-0.0153	0.0144
Matrix of *Σ*				
819.82	0.00	0.00	0.00	0.00
0.00	434.17	0.00	0.00	0.00
0.00	0.00	387.36	0.00	0.00
0.00	0.00	0.00	369.27	0.00
0.00	0.00	0.00	0.00	353.02
Matrix of *V'*				
0.1780	-0.1856	0.2090	0.0149	-0.0886
0.2368	-0.2177	0.8174	-0.0919	-0.2895
0.4637	0.1740	-0.1633	-0.8231	0.0979
0.4114	0.3867	0.1215	0.4053	0.2429

Table 4. Cosine Calculation of $U_{M\times K}\times\Sigma_{K\times K}$

Wanderer	Farewell	0.973729	Han Dynasty	run amuck	0.9673017
	a few words	0.973729		Ryongson	0.9657026
	Separate	0.973729		War	0.9635461
wealthy	Nothingness	0.9622504		Huanglong	0.9635461
	Competed	0.9622504		Border-fortress	0.9635461
	gentle and simple	0.9622504		grape	0.9635461
golden armor	Loulan	0.9847319		battlefield	0.958317
	Qinghai	0.9689628		desert	0.9571064
hero	Resurgence	0.9831921	country	how can	0.9843091
	Luxury	0.9831921		much difficulties	0.9707253
	Xi Shi	0.9708392		Karlaua	0.9683641
	Weeds	0.9697423		hero	0.9558059

As we described in last section, the value of cosine calculate by two lines of $U_{m\times k}\times\Sigma_{k\times k}$ matrix, represent the correlation of words. The greater the value is, the more closely relatedness is between words. So, we can find some associated words shown in Table 4. **Error! Reference source not found.**The similar distribution in categories of two words can show the similar emotion they have.

3.4 Additional Information

Besides computing the semantic relatedness of words from matrix decomposed by SVD, Table 5 show the cosine calculation result of V matrix, in which element (i, j) show the correlation between i and j categories. As we can see in the table, farewell categories have higher association with others, but it's limited to 10-16 magnitude, which is approximately equal to 0, almost irrelevant.

It can be seen from experiment data above that through the semantic relatedness computing of words by process improved method, we can find out associated word in emotion. Meanwhile, we can judge whether the classification is reasonable from analyzing the *V* matrix.

Table 5. Cosine Calculation of V Matrix

	war	Farewells	season	weather	love	plant	nostalgia	festival	Hermit	landscape
Patriotic	0.38	0.27	0.24	0.26	0.21	0.20	0.37	0.22	0.15	0.25
War	0.00	0.27	0.27	0.30	0.15	0.13	0.28	0.18	0.08	0.20
Farewells	0.00	0.00	0.44	0.44	0.29	0.33	0.38	0.36	0.25	0.38
Season	0.00	0.00	0.00	0.54	0.35	0.43	0.31	0.34	0.26	0.41
Weather	0.00	0.00	0.00	0.00	0.29	0.33	0.30	0.32	0.21	0.34
Love	0.00	0.00	0.00	0.00	0.00	0.31	0.27	0.25	0.12	0.21
Plant	0.00	0.00	0.00	0.00	0.00	0.00	0.25	0.30	0.19	0.28
Nostalgia	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.31	0.16	0.29
Festiva	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.20	0.24
Hermit	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.32

3.5 The Limitation of Improved Method

Although we have seen many results in which process improved method of LSA performs better than that before, we also present in this work some examples in which it performs weak.

One of the strength of the method sometimes also serves as its weakness. Although it's outstanding in finding the relatedness of words by their similar behavior in different categories, a prominent problem is that it is weak on figuring out the relatedness of words appear in one category merely with lower weight. So it's a problem caused by word sparse as well.

In my opinion, this lower frequency words is hard to use in poem creating, and have lower semantic relatedness with most words. Once we use it in the calculation, accuracy will be affected. In this case, a possible solution to improve accuracy can combine the other method such as MI, we can get overlap data of two methods as final result. And it can be seen that semantic relatedness computing of tang poem still have a long way to go. In the future, we can do more research in finding effective solution of word sparse.

4. Conclusion

The main innovative points of this paper is that we come up with a process improved method, which classify the Tang corpus by emotion at first, and then use matrix representation of words' semantic by "words-by-poetry categories" instead of "words-by-poems" ,finally decomposed it by single value decomposition as well.

So, what makes it different from others is that classified the poetry of corpus and implemented segmentation and statistics on each classification instead of each poem. With this process improvement, we can reduce the size of original matrix, so as to improve computational efficiency.

Experimental result of cosine of vectors in $U_{m\times k}\times\Sigma_{k\times k}$ matrix shows that the similar distribution in categories of two words can show the similar emotion they have, so we can find associated words by this way. Meanwhile, cosine calculation of *V* matrix shows the relevance of poetry categories.

This method based on the simple syntax of Tang poetry which create by words with similar emotion, and design for semantic relatedness computing of Tang poetry not any modern Chinese article. So all of this provides the basis for establishment of words emotions related libraries and emotions unified judgment of automatically generated Tang poetry.

References

[1] Manning, Christopher D. *Foundations of statistical natural language processing*. Ed. Hinrich Schütze. MIT press. 1999.
[2] Su, JS., CL. Zhou, YH. Li. The establishment of the annotated corpus of Song dynasty poetry based on the statistical word extraction and rules and forms. *Journal of Chinese Information Processing.* 2007; 21(2): 52-57.
[3] Xue N. Chinese word segmentation as character tagging. *Computational Linguistics and Chinese Language Processing*. 2003; 8(1): 29-48
[4] Hu, JF. The lexicon meaning analysis-based computer aided research work of Chinese ancient poems. *Ph. D. Thesis*. *Beijing: Peking Universit.* 2001.
[5] Yang, Yu-Zhen, Pei-Yu Liu, Pei-Pei Jiang. Research on Text Representation with Combination of Syntactic in Vector Space Model. *Jisuanji Gongcheng/ Computer Engineering.* 2011; 37(3).
[6] Li, Liang-Yan, Zhong-Shi He, Yong Yi. *Poetry stylistic analysis technique based on term connections*. Machine Learning and Cybernetics, 2004. Proceedings of 2004 International Conference on. IEEE, 2004; 5: 2713-2718.
[7] Budanitsky, Alexander, Graeme Hirst. Evaluating wordnet-based measures of lexical semantic relatedness. *Computational Linguistics.* 2006; 32(1): 13-47.
[8] Agirre, Eneko, et al. *A study on similarity and relatedness using distributional and WordNet-based approaches*. Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics. Association for Computational Linguistics. 2009.
[9] Gabrilovich, Evgeniy, Shaul Markovitch. Computing Semantic Relatedness Using Wikipedia-based Explicit Semantic Analysis. *IJCAI*. 2007; 7: 1606-1611.
[10] Deerwester, Scott C., et al. Indexing by latent semantic analysis. *JASIS.* 1990; 41(6): 391-407.
[11] Church, Kenneth Ward, Patrick Hanks. Word association norms, mutual information, and lexicography. *Computational linguistics.* 1990; 16(1): 22-29.
[12] Kozima, Hideki, Teiji Furugori. *Similarity between words computed by spreading activation on an English dictionary*. Proceedings of the sixth conference on European chapter of the Association for Computational Linguistics. Association for Computational Linguistics. 1993.
[13] Morris, Jane, Graeme Hirst. Lexical cohesion computed by thesaural relations as an indicator of the structure of text. *Computational linguistics.* 1991; 17(1): 21-48.
[14] Miller, George A., et al. Introduction to wordnet: An on-line lexical database. *International journal of lexicography*. 1990; 3(4): 235-244.
[15] Resnik, Philip. *Using information content to evaluate semantic similarity in a taxonomy*. Proceedings of the 14th international joint conference on Artificial Intelligence. 1995; 1: 448–453.
[16] HU, Junfeng, Shiwen YU. The Computer Aided Research Work of Chinese Ancient Poems. *Acta Scicentiarum Naturalum Universitis Pekinesis*. 2001; 5: 022.
[17] Zhou, Cheng-Le, Wei You, Xiaojun Ding. Genetic algorithm and its implementation of automatic generation of chinese songci. *Journal of Software.* 2010; 21(3): 427-437.
[18] Ramos, Juan. *Using tf-idf to determine word relevance in document queries*. Proceedings of the First Instructional Conference on Machine Learning. 2003.
[19] Zhang X, Wang M. Sparse representation for detection of microcalcification clusters. *TELKOMNIKA Indonesian Journal of Electrical Engineering.* 2012; 10(3): 545-550.
[20] TIAN, Dong-feng, Fei OU, Wei SHEN. On the Application of Matrix Singular Value Decomposition Theory in Chinese Text Classification. *Mathematics in Practice and Theory.* 2008; 24: 021.
[21] Zhang PY. A HowNet-Based Semantic Relatedness Kernel for Text Classification. *TELKOMNIKA Indonesian Journal of Electrical Engineering*. 2013; 11(4): 1909-1915.
[22] Radinsky K, Agichtein E, Gabrilovich E, et al. *A word at a time: computing word relatedness using temporal semantic analysis*. Proceedings of the 20th international conference on World Wide Web. ACM. 2011: 337-346.