AIP Conference Proceedings

Assessing semantic similarity of texts – Methods and algorithms

Cite as: AIP Conference Proceedings **1910**, 060012 (2017); https://doi.org/10.1063/1.5014006
Published Online: 07 December 2017

Anna Rozeva, and Silvia Zerkova

View Online

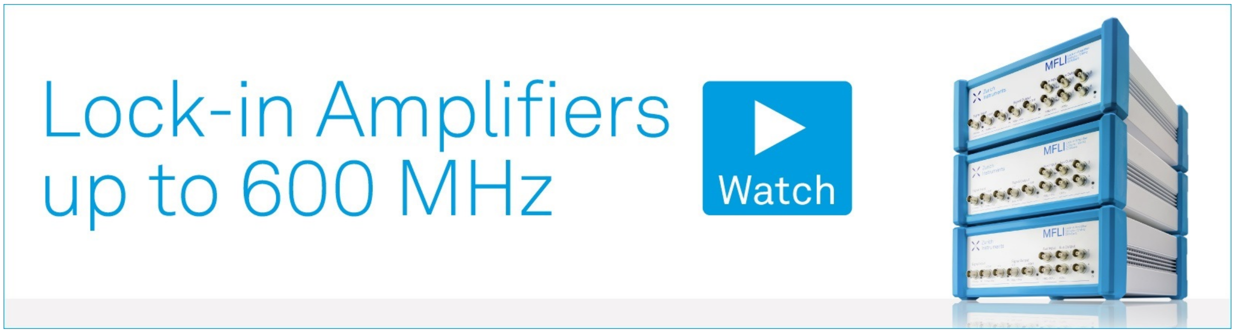
Export Citation

ARTICLES YOU MAY BE INTERESTED IN

Big data processing in the cloud - Challenges and platforms
AIP Conference Proceedings **1910**, 060013 (2017); https://doi.org/10.1063/1.5014007

Automatic short answer scoring using words overlapping methods
AIP Conference Proceedings **1818**, 020042 (2017); https://doi.org/10.1063/1.4976906

Big data analytics for early detection of breast cancer based on machine learning
AIP Conference Proceedings **1910**, 060016 (2017); https://doi.org/10.1063/1.5014010



AIP Publishing

AIP Conference Proceedings **1910**, 060012 (2017); https://doi.org/10.1063/1.5014006 **1910**, 060012

© 2017 Author(s).

Assessing Semantic Similarity of Texts – Methods and Algorithms

Anna Rozeva[1, a] and Silvia Zerkova[1, b]

[1] *Technical University of Sofia, 8 Kliment Ohridski blv., 1000 Sofia, Bulgaria*

[a)] Corresponding author: arozeva@tu-sofia.bg
[b)] siz@abv.bg

Abstract. Assessing the semantic similarity of texts is an important part of different text-related applications like educational systems, information retrieval, text summarization, etc. This task is performed by sophisticated analysis, which implements text-mining techniques. Text mining involves several pre-processing steps, which provide for obtaining structured representative model of the documents in a corpus by means of extracting and selecting the features, characterizing their content. Generally the model is vector-based and enables further analysis with knowledge discovery approaches. Algorithms and measures are used for assessing texts at syntactical and semantic level. An important text-mining method and similarity measure is latent semantic analysis (LSA). It provides for reducing the dimensionality of the document vector space and better capturing the text semantics. The mathematical background of LSA for deriving the meaning of the words in a given text by exploring their co-occurrence is examined. The algorithm for obtaining the vector representation of words and their corresponding latent concepts in a reduced multidimensional space as well as similarity calculation are presented.

1. INTRODUCTION

The problem of text analysis has been intensively researched in the recent years in different text-related application areas like: text classification, information retrieval, topic tracking, document clustering, questions generation, question answering, short answer scoring, machine translation, essay scoring, text summarization, topic detection [11]. It proves to be extremely challenging in educational systems as a "hot" research topic with high potential for facilitating online education and online assessment. Some educational tasks that require analysis of text are: free-text answers' assessment in tests by determining semantic similarity of sentences [2]; analysis of the difficulty of texts for selecting them according to specific pedagogical objectives [1], analysis of the similarity of a student-generated text fragment to one generated by an expert [4].

Text mining [6] is the basic approach and technique for performing sophisticated analysis of natural language text. One of its important methods concerns determining semantic similarity of texts. This analysis is performed on defined structured model of the natural language text, which is obtained after performing certain text processing and transformations. Usually it is vector-based, where a text document is represented by vector of the features that describe its content. The model of a collection of documents is a space of the document vectors, which is high dimensional. The basic research tasks there refer to reducing the dimensionality of the document vector space and the more efficient extraction of the features characterizing it.

The problem of determining semantic similarity of texts has been addressed by defining algorithms and similarity measures. Classification of similarity measures at string, corpus and knowledge-based levels is presented in [10] and [11]. Classification of similarity metrics based on the model used for text representation as well as on the structure units of the text content is shown in [3]. Semantic similarity algorithm, based on the clustering approach and the concept of higher-order co-occurrences in distributional semantics is proposed in [12]. The task of predicting textual coherence by implementing the random indexing method for the location of documents in a semantic space after its mathematical translation is discussed in [13]. Approach for determining semantic similarity of documents is

Proceedings of the 43rd International Conference Applications of Mathematics in Engineering and Economics
AIP Conf. Proc. 1910, 060012-1–060012-8; https://doi.org/10.1063/1.5014006
Published by AIP Publishing. 978-0-7354-1602-4/$30.00

shown in [15]. It combines ontology based similarity model and information content knowledge-based similarity measure. Ontology based semantic document model is presented in [17].

The most generally applied method of text mining and corpus-based similarity measure is latent semantic analysis (LSA) [5]. It addresses the problem of the high dimensionality of the vector space model and the more adequate capturing of the text document meaning by implementing mathematical transformation with singular value decomposition of the document vector matrix. Assessment of different LSA algorithms for determining the mapping of patents and scientific publications at a large scale is performed in [9].

The paper is organized as follows: text mining techniques for obtaining a representative structured document model are presented in Section 2; semantic similarity algorithms and measures are shown in Section 3; LSA mathematical background and algorithm are discussed in Section 4, followed by conclusion and guidelines for future work.

2. TEXT MINING TECHNIQUES FOR DOCUMENT MODELING

Text mining [6] performs processing and analysis of collections of text documents, which exposes similarity and relationships among them and more sophisticated knowledge discovery tasks in the form of semantic related patterns. General architecture of text mining system is shown in Fig.1.

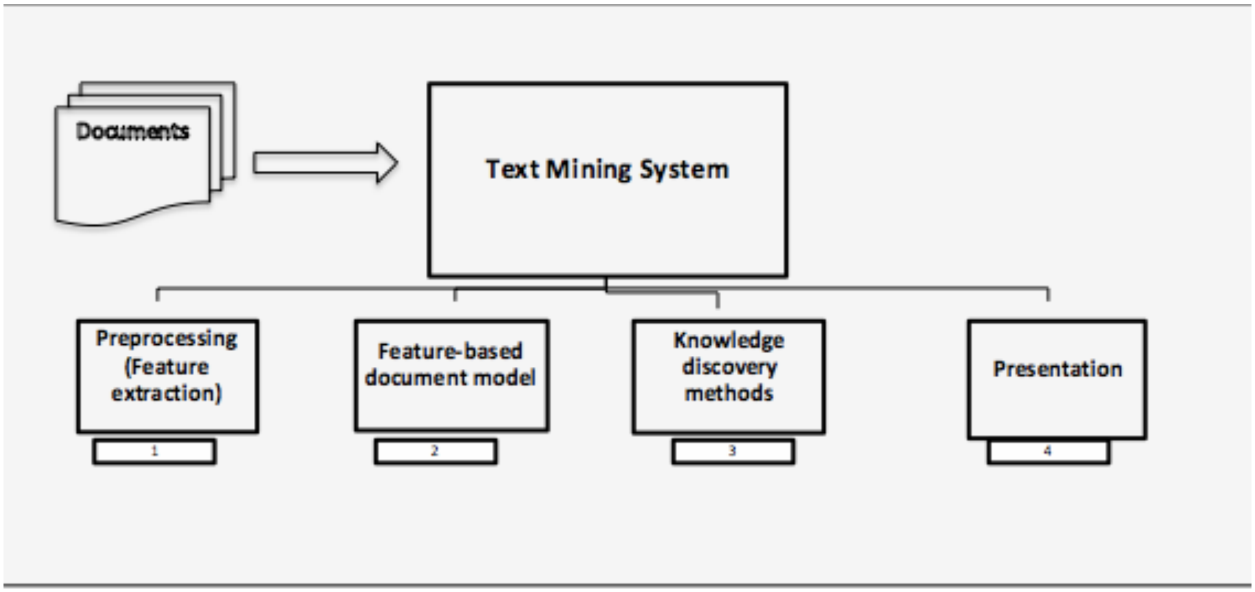

Documents
Text Mining System
Preprocessing (Feature extraction)
1
Feature-based document model
2
Knowledge discovery methods
3
Presentation
4

FIGURE 1. Text mining system architecture

The input to the text mining system is unstructured collection of texts in free language, referred to as documents. Processing and analysis of a document collection demands breaking the text into meaningful elements, words, sentences, paragraphs, etc. This is performed at the preprocessing stage (1) of the text mining system. At this phase understandable elements from the character stream are extracted. The text understanding consists of two phases, i.e. text linguistics and context (domain knowledge). The linguistics task concerns processing of natural human language, which involves the following steps:

- Tokenization – the stream of characters is divided into meaningful items, which for the text mining systems most often are words and sentences;
- Part-of-Speech Tagging – words are annotated according to the context they have in a sentence, i.e. noun (object), relationship (preposition), etc.
- Syntactical parsing – is performed after the rules of an adopted grammar. As a result phrases of items, which are syntactically grouted together are produced, or dependencies between words are built. Due to restrictions concerned with volume, time, robustness and efficacy the complete parsing of sentences is often replaced by shallow parsing. In the later case simpler and unambiguous phrases are obtained.

As a result of the linguistic task words and phrases tagged with their role in a sentence are produced. The obtained linguistic categories serve as input for the next preprocessing task, the contextual one, which is domain or problem dependent and semantically richer. The basic methods of the domain dependent preprocessing are:

- Categorization – each document is attached a set of concepts (keywords);
- Information extraction – the information providing the meaning of a document as entities and relations with co-reference resolved is extracted and represented in structured form. In that way it becomes available for further more sophisticated processing.

The preprocessing stage of the text mining system workflow provides for obtaining structure from the unstructured text collection stream. The entities (items) with their relationships extracted are referred to as features or terms. They represent the meaning of a document. They are the basis for designing a structured model of the initial document collection. The most general document model is vector-based. The document is represented as a vector of the features obtained by preprocessing and their weights as shown in Fig.2

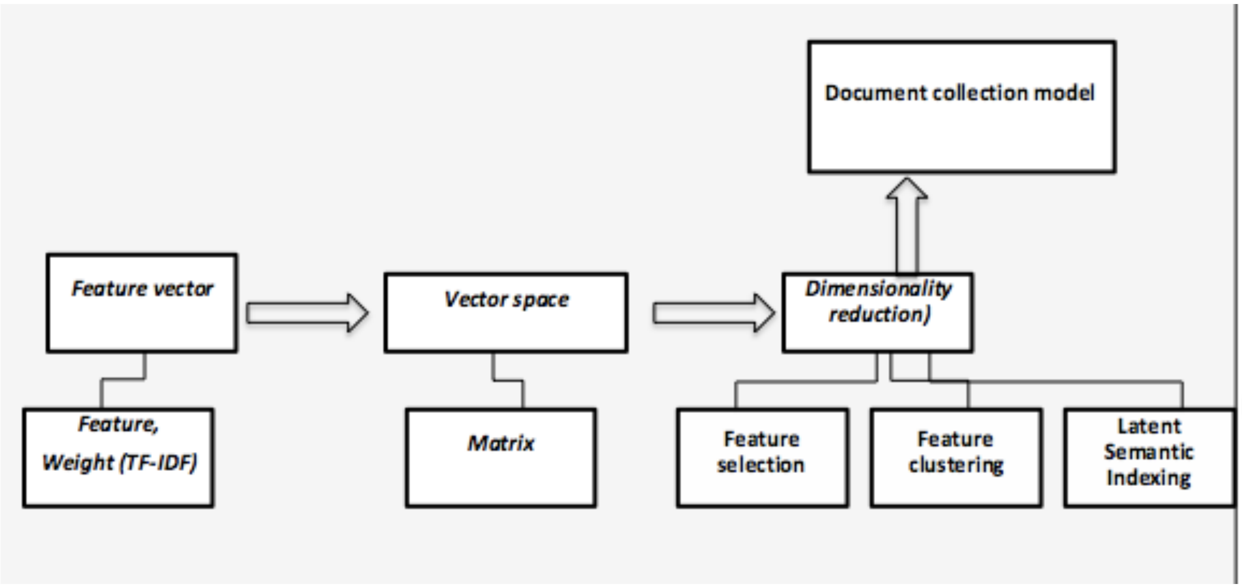


FIGURE 2. Document collection model

The weighting scheme that is most often implemented is the term frequency – inverted document frequency (TF-IDF). It is calculated from the frequency of a feature *f* in the document *d*, the number of documents in the collection *N* and the number of documents, containing this feature *DocFreq(f)* as shown in equation (1).

$$TF - IDF = TermFreq(f, d). \log\left(\frac{N}{DocFreq(f)}\right) \qquad (1)$$

The representation model of the document collection, stage (2) in Fig.1, is a vector space with the features as dimensions and the documents as vectors in the space. It is represented by *Nxn* matrix with N documents and n features. The number of extracted features at the document preprocessing stage is usually very high and the matrix of the document vector space is extremely sparse. Since most of the features do not contribute to the document meaning reduction of the vector space dimensionality will facilitate the matrix processing without any loss of the document meaning. The most frequently implemented methods for reduction, as shown in Fig.2 are:

- Feature selection – removal of common words without contribution to the document meaning;
- Feature extraction – implementation of clustering techniques for grouping features that are synonyms and the obtained feature groups are further on used as synthetic features;
- Latent semantic indexing – performs singular value decomposition of the document matrix.

The most prominent knowledge discovery methods in the text mining architecture, stage (3) in Fig.1, are: distribution analysis, clustering, trend analysis, and association rules [6]. In order to perform these methods in many practical applications, domain or background knowledge has vital role. This information is encoded in ontologies, lexicons, thesauri and taxonomies. Approach for text mining with implementation of ontologies is presented in [16].

3. SEMANTIC SIMILARITY ALGORITHMS

Semantic similarity measures and algorithms are discussed in [10] and [11]. They are classified as corpus-based and knowledge based. The corpus-based algorithms process text document collection, extract information from it and use the information obtained for determining similarity between text elements (words, phrases). The knowledge-based algorithms derive semantic similarity by using information from semantic repositories (ontologies, semantic networks).

3.1 Corpus-based similarity algorithms

Hyperspace analogue to language inspects co-occurrences of words and puts them in semantic space as matrix with elements, which represent the degree of association between the corresponding row and column words. The algorithm measures the co-occurrence of a focus word to neighboring words and weights it inversely proportional to the distance between the word and the focus word. The weight is inserted as element in the matrix.

Explicit semantic analysis measures the degree to which two texts are related. It implements vector-based representation of the texts and the cosine measure to assess how are they related.

Point-wise mutual information – information retrieval - semantic similarity between words is calculated by means of probabilities. High frequency of co-occurrence of words determines high score of the point-wise mutual information similarity measure. Its extension second-order co-occurrence point-wise mutual information sorts lists of neighbor words of the two input words. This provides for obtaining similarity between words, which do not co-occur frequently, but have the same co-occurring neighboring words.

Normalized Google distance is obtained from the matches to a set of keywords returned by the Google search engine. When two words have similar meaning their Google distances are close to one another. Two words, appearing separately have infinite Google distance, while words appearing together have Google distance equal to zero.

Extraction of words with similar distribution using co-occurrences assumes that similar meaning appear in similar contexts. The method uses three words frame for determining co-occurrences. Similarity is determined by applying measures to the corresponding word vectors. The basic measures used are similarity based on words' colocation sets and similarity based on sets of similar by distribution word sets.

Latent semantic analysis is based on the assumption that similar words appear in similar text fragments. It will be discussed in more detail in the next section.

3.2 Knowledge-based similarity algorithms

There are two types of similarity measures – semantic similarity and semantic relatedness. They are based either on information content or on path length. They implement relations between words and concepts that are available in semantic repositories. Scoring function for calculating semantic similarity between texts T_1 and T_2 from [11] is described by equation (2).

$$sim(T_1, T_2) = \frac{1}{2}\left(\frac{\sum_{w \in T_1} maxsim(w, T_2) * idf(w)}{\sum_{w \in T_1} idf(w)} + \frac{\sum_{w \in T_2} maxsim(w, T_1) * idf(w)}{\sum_{w \in T_2} idf(w)}\right) \qquad (2)$$

The following metrics identify words with high semantic similarity:

- Based on the shortest *path length* between concepts obtained by the count of nodes and the maximum taxonomy depth D – equation (3).

$$sim = -log\frac{length}{2D} \qquad (3)$$

- Based on function of the overlap of definitions, which is provided by dictionary and performs word disambiguation;

- Based on a combination of the estimated depth of the two analyzed concepts c_1 and c_2 in taxonomy (Word Net) and the least common subsumer's (LCS) depth – equation (4).

$$sim = \frac{2depth_{LCS}}{depth_{c1}+depth_{c2}} \tag{4}$$

- Based on a function of the information content IC of the least common subsumer of two concepts, which is calculated by the probability $(-\log(P(c))$ of the presence of the concept c in the text collection - equation (5).

$$sim = \frac{2IC_{LCS}}{IC_{c1}+IC_{c2}} \tag{5}$$

Besides the corpus - and knowledge based similarity measures hybrid measures combining several ones are implemented as well. As shown in [18] semantic similarity of sentences is determined by implementing semantic relatedness measure over the sentence followed by knowledge-based semantic similarity scores calculated for the words that appear in the same roles in both sentences.

4. LATENT SEMANTIC ANALYSIS

Latent semantic analysis / indexing (LSA / LSI) [5], [7], [8], [14] is a major approach in the text mining processing. It has proven to be the most widely applied corpus-based similarity measure. It facilitates the capture of the most descriptive features of the document meaning and thus the elaboration of a document vector space with reduced dimensionality. The basic assumptions of LSI are [8]:

- The meaning of a word is obtained as an average of its presence in all documents;
- The meaning of multi-word constructs is determined by the way the words are configured within it;
- The latent (hidden) associations among words are discovered by the inspection of word co-occurrence with each single word.

Associations between points in a reduced vector space are induced on the basis of determining correlations in their distribution. This semantic knowledge LSI determines by an inductive process on the basis of the way words and phrases are distributed within the documents.

4.1 Algorithm

The processing steps of the LSI algorithm [7] are presented in Fig. 3.

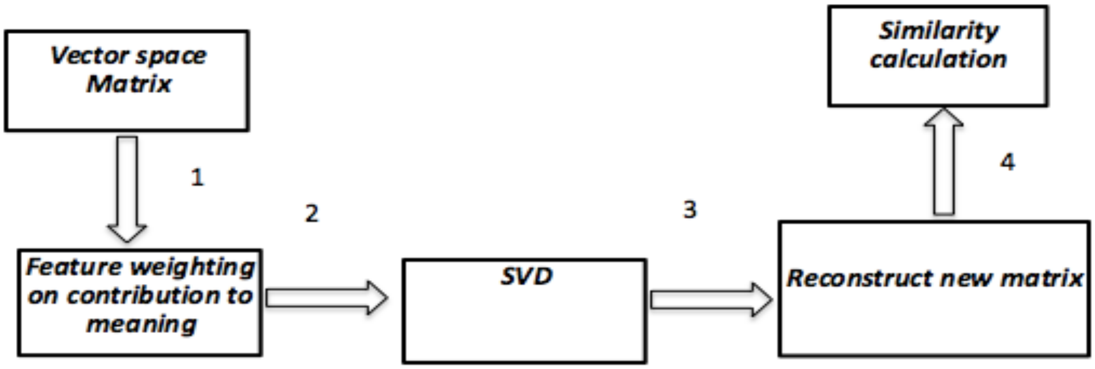


FIGURE 3. LSI algorithm

Step1: Implements frequency-to weight transformation of features within a document. The transformation addresses the growth rate of the understanding of the text meaning. Equation (6) implements the transformation.

$$weight_{ij}^{loc} = \log\,(freq_{ij}^{loc} + 1) \tag{6}$$

Further on the global feature's frequency within the document collection is calculated after equation (7).

$$weight_{i}^{glob} = \frac{1+\sum_{j=1}^{n} p_{ij}.\log\,(p_{ij})}{\log\,(n)}, \quad p_{ij} = \frac{frec_{ij}^{loc}}{\sum_{j=1}^{n} frec_{ij}^{loc}} \tag{7}$$

The sum of local the feature's local frequencies within the documents represents its global frequency. The weighted value of a feature is calculated by equation (8).

$$weight_{ij}^{feature} = \frac{weight_{ij}^{loc}}{weight_{ij}^{glob}} \tag{8}$$

The frequency-to-weight transformation assesses the contribution of a feature to the meaning of a text, as the co-occurrence, expressed by local and global frequencies, does not imply high informative value. The implemented weighting method attaches higher weight to features that have high local frequency and low global frequency.

Step2: Singular value decomposition of the matrix, obtained from Step1 to three matrices after equation (9).

$$M_{mxn} = T\Sigma D^{T} \tag{9}$$

Matrices $T_{mxm}\ and\ D_{nxn}^{T}$ are orthogonal, where *T* is the matrix representation of features and *D* is the matrix representation of documents. Matrix Σ is "cellular diagonal" matrix, i.e. it has the diagonal matrix D_{rxr} , *r < min(m, n)*, on the main diagonal and all other elements on the main diagonal are 0. After the SVD transformation of *D* the singular values of the matrix *M* are obtained on the main diagonal of *D* in a descending order. The singular values of *M*, contained in *D*, represent the dimensions of the meaning of the elements (words and contcepts) of the analyzed text.

Step3: Construction of new lower dimensionality matrix M^s can be performed by the substitution of some of the singular values with 0. The rule for the substitution is from the lowest to the greatest singular value. The remaining non-zero singular values are *s*, *s<r* (the number of values in *D*). The construction of M^s is achieved by calculation of equation (5) by replacing Σ with Σ^s. The value of *s*, which represents the number of dimensions of the newly constructed matrix of the semantic representation of co-occurrence of words within the documents, is provided as parameter. The matrices of the feature T^s and document D^T representations in the lower dimensionality space are obtained by multiplication with the singular value matrix Σ^s.

Step4: Similarity of vectors in the reduced dimensionality space is determined by the cosine of the angle between them. Cosine = 1 (parallel vectors) implies that the features are synonyms, while cosine = 0 means that they are dissimilar. Calculation of the cosine of vectors v_1 and v_2 is performed by equation (10).

$$\cos(\Theta) = \frac{v1.v2}{\|v1\|.\|v2\|} \tag{10}$$

4.2 Similarity calculation

1. Word-to-word similarity with word vectors w_1 and w_2 in reduced space is calculated from a matrix, obtained by equation (11).

$$M_s M_s^T = T_s T_s^T \tag{11}$$

In (11) $M_s^T\ T_s^T$ denote the transposed matrices. Similarity between word vectors w_1 and w_2 is calculated by the cosine of the vectors in the ith row and jth column of the matrix.

2. Document-to-document similarity with document vectors d_1 and d_2 in reduced space is calculated from a matrix, obtained by equation (12).

$$M_s^T M_s = D_s D_s^T \qquad (12)$$

Similarity between document vectors d_1 and d_2 is calculated by the cosine of the vectors in the ith row and jth column of the matrix, obtained in (12).

3. Word-to-document similarity with vectors w_1 and d_1 is calculated by the cosine of the ith row and jth column of M_s, divided by $\|w_1'\|\ and\ \|d_1'\|$, where (13).

$$w_1' = w_1\sqrt{\Sigma} \quad , \qquad d_1' = \sqrt{\Sigma}d_1 \qquad (13)$$

Similarity between document vectors d_1 and d_2 is calculated by the cosine of the vectors in the ith row and jth column of the matrix, obtained in (13).

4.3 LSA applications

Practical testing of LSA for determining similarity between documents and parts of documents can be performed in http://lsa.colorado.edu with methods shown in Fig.4.

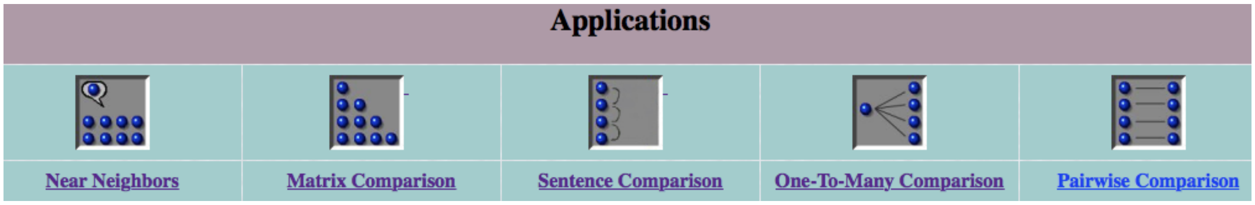


FIGURE 4. LSI applications

Near neighbors selects a set of n near neighbor terms based on a submitted term or piece of text. The terms returned are those in the LSA space, which are nearest the submitted term or piece of text.

Matrix comparison performs analysis of the similarity of multiple texts or terms within a particular LSA space. Each text is compared to all other texts.

Sentence comparison analyses similarity of sequential sentences where each one is compared to the next.

One-to-many comparison determines similarity of multiple texts, where one designated text is compared to all other texts.

Pairwise comparison performs similarity analysis of multiple texts with each pair of texts being compared to one another.

CONCLUSION

The paper presented basic aspects of the theoretic and application background of the assessment of semantic similarity of texts. The general technology for sophisticated analysis of text – text mining with methodology, architecture and challenges was discussed. The focus was the design of structured model of natural language text. The generally adopted vector space model serving the task was shown. Algorithms for assessing semantic similarity and similarity measures were discussed. The method of latent semantic analysis as an important text mining method and algorithm for assessing semantic similarity of text was explained with the mathematical background, algorithm for implementation and similarity calculation. The research will support further establishment of conceptual model, framework and architecture for implementation of sophisticated knowledge discovery methods in educational systems for facilitating online assessment. Further on we will evaluate and enhance similarity measures and LSA algorithms for solving typical e-assessment tasks in online educational environments.

REFERENCES

1. C. Graesser, D. S. McNamara, and J. M. Kulikowich, "Coh-Metrix: Providing Multilevel Analyses of Text Characteristics", *Educational Researcher*, 40(5), (2011), pp. 223-234.
2. P. Sravanthi and B. Srinivasu, "Semantic Similarity between Sentences", *International Research Journal of Engineering and Technology (IRJET)*, vol.4, issue 1, Jan.2017, pp. 156-161.
3. M. Ivanov and M. Raykova, "Metrics for Assessing the Similarity in Text Documents ", Computer Science and Education in Computer Science*,* vol.12, No1, 2016*,* pp. 63-77.
4. R. Azevedo, A. Johnson, A. Chauncey and C. Burkett, "Self-regulated Learning with MetaTutor: Advancing the Science of Learning with MetaCognitive Tools", In *New Science of Learning: Computers, Cognition, and Collaboration in Education* edited by M. S. Khine and I.M. Saleh (Springer Science+Business Media, LLC 2010), pp. 225-247.
5. T. Landauer, D. McNamara, S. Dennis and W. Kintsch (Eds.), *The handbook of latent semantic analysis* (Routledge, New York, 2011) pp. 3-10.
6. R. Feldman and J. Sanger, *The Text Mining Handbook. Advanced Approaches in Analyzing Unstructured Data* (Cambridge University Press, New York, 2007), pp. 13-19.
7. P. Reidy, An Introduction to Latent semantic Analysis, Retrieved 10.06.2017 from http://www.ling.ohio-state.edu/~reidy.16/LSAtutorial.pdf
8. A. Kao, S. Poteet, J. Wu, W. Ferng, R. Tjoelker and L. Quach, Latent Semantic Analysis for Text Mining and Beyond, in *Intelligent Multimedia Databases and Information Retrieval: Advancing Applications and Technologies,* edited by L. Yan and Z. Ma (IGI Global, 2012), pp. 253-280.
9. T. Magerman, B. Van Looy, B. Baesens and K. Debackere, Assessment of Latent Semantic Analysis (LSA) text mining algorithms for large scale mapping of patent and scientific publication documents (Katholieke Universiteit Leuven Department of Mangerial Economics Strategy and Innovation, 2011, Working Paper 1114), pp. 1-75.
10. W. H. Gomaa and A. A. Fahmy, A Survey of Text Similarity Approaches, International Journal of Computer Applications 68(13), 0975 – 8887 (2013), pp. 13-18.
11. M.K.Vijaymeena1 and K.Kavitha, A Survey on Similarity Measures in Text Mining, Machine Learning and Applications, Machine Learning and Applications: An International Journal (MLAIJ) 3(1), (2016) pp. 19-28.
12. S. F. Hussain, "A New Co-similarity Measure: Application to Text Mining and Bioinformatics", Ph.D. thesis, Institut National Polytechnique de Grenoble, 2010.
13. D. Higgins and J. Burstein, Sentence Similarity Measures for Essay Coherence, In Proceedings of the 7th International Workshop on Computational Semantics (IWCS), Tilburg, The Netherlands, January 2007, pp. 1-12.
14. C. Boling, Semantic Similarity of Documents Using Latent Semantic Analysis, Proceedings of the National Conference On Undergraduate Research (NCUR) 2014 University of Kentucky, Lexington, KY April 3-5, 2014, pp. 1083-1092.
15. K. L. Sumathy and D. Chidambaram, A Hybrid Approach for Measuring Semantic Similarity between Documents and its Application in Mining the Knowledge Repositories, (IJACSA) International Journal of Advanced Computer Science and Applications, 7(8), 2016, pp. 231-237
16. A. Rozeva, Classification of text documents supervised by domain ontologies, Journal of Applied Technologies and Innovations, 8 (3), 2012, pp. 1-12.
17. A. Rozeva, Enhancing domain knowledge with semantic models of web documents, Journal of Mathematics and System Science, 3(7), 2013, pp. 319-326.
18. A. Nitish, A. Kartik and B. Paul, DERI&UPM: Pushing Corpus Based Relatedness to Similarity: Shared Task System Description, in *First Joint Conference on Lexical and Computational Semantics (*SEM), Montreal, Canada,* (Association for Computational Linguistics, 2012), pp. 643–647.