RESEARCHERID
THOMSON REUTERS

ISSN: 2277-9655
[Umanandhini * *et al.*, 6(12): December, 2017] Impact Factor: 4.116
IC™ Value: 3.00 CODEN: IJESS7

IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

FUZZY SCORE BASED SHORT TEXT UNDERSTANDING FROM CORPUS DATA USING SEMANTIC DISCOVERY

D.Umanandhini*[1] & S.Manimegalai[2]

[*1]Assistant professor, Department of Computer Applications, Kovai Kalaimagal College of Arts and Science, Coimbatore

[2]M.Phil Scholar, Department of Computer Science, Kovai Kalaimagal College of Arts and Science, Coimbatore

DOI: 10.5281/zenodo.1116682

ABSTRACT

Short text understanding and short text are always more ambiguous. These short texts are produced including Search queries, Tags, Keywords, Conversation or Social posts and containing limited context. Generally short texts do not contain sufficient collection of data to support many state-of-the-art approaches for text mining such as topic modelling. It presents a comprehensive overview of short text understanding. Here we used a novel framework are Text Feature Extraction Algorithm and Fuzzy weighted Vote algorithm First, Text classification based on semantic feature extraction.

Its goal is that use semantic feature extraction to improve the performance of classifier. And second, Fuzzy weighted Vote algorithm is the combination of Fuzzy logic and weighted vote algorithm, which means it generates the fuzzy score and then based on this score the weight is calculated during shortening the text. In experimental results, the novel Feature Extraction and voter has higher safety performance than the previous classification algorithms. This proposed criterion can provide almost accurate safety and also a good range of accessibility. We have proved that in problems where the weighted voting distinguish some alternatives and finds the best alternative. Reduced Computation time comparing to other previous process and schemes.

KEYWORDS: Short text understanding, Text segmentation, Concept labelling, Tagger

I. INTRODUCTION

Understand Short Texts

Mapping a short text (i.e., a word or a phrase) to concepts is an important problem in natural language processing. For instance, the word apple can be mapped to the concepts fruit and food or company and firm, and a phrase apple orchard can be mapped to a piece of land with fruit trees. This mapping of words to their most appropriate concepts is what we seek to accomplish in conceptualization. It is an important component of semantic understanding tasks, for example, matching of a search query to an advertisement. The query-ad matching task depends heavily on conceptualization because more full-blown analysis such as syntactic parsing is often not applicable nor helpful for understanding queries or ad keywords.

It focuses on conceptualizing from texts or words. For example, given the word “India,” a person will form in his mind concepts such as country or region. Given two words, “India” and “China,” the top concepts may shift to Asian country or developing country, etc. Given yet another word, “Brazil,” the top concepts may change to BRIC or emerging market, etc. Besides generalizing from instances to concepts, humans also form concepts from descriptions. For example, given words “body,” “smell” and “colour,” the concept of wine comes into our mind. Certainly, instances and descriptions may mix, for example, we conceptualize {“apple,” “headquarter”} to company, but {“apple,” “smell,” “colour”} to fruit.

A latent topic is represented by a set of words. Machines are not able to grasp the concepts behind these words, nor do they know the properties and relationships associated with the concepts. In particular, using statistical

ISSN: 2277-9655
Impact Factor: 4.116
CODEN: IJESS7

approaches to find topics from short text (search queries, twitter messages, etc.) is often infeasible, as short text does not have enough content from which we can build a statistically meaningful model.

Information explosion highlights the need for machines to better understand natural language texts. It, focus on short texts which refer to texts with limited context. Many applications, such as web search and micro blogging services etc., need to handle a large amount of short texts. Obviously, a better understanding of short texts will bring tremendous value. One of the most important tasks of text understanding is to discover hidden semantics from texts. Many efforts have been devoted to this field. For instance, named entity recognition (NER) locates named entities in a text and classifies them into predefined categories such as persons, organizations, locations, etc. Topic models attempt to recognize "latent topics", which are represented as probabilistic distributions on words, from a text. Entity linking focuses on retrieving "explicit topics" expressed as probabilistic distributions on an entire knowledgebase. However, categories, "latent topics", as well as "explicit topics" still have a semantic gap with humans' mental world.

Semantic Discovery

Measuring semantic similarity between terms is a fundamental problem in lexical semantics and it finds many applications in web and document search, question and answer systems, and other text analytics and text understanding scenarios. By terms, we mean either single words or multi-word expressions (MWEs). The two terms are semantically similar, if their meanings are close or the concept or object that they represent share many common attributes. For example, "emerging markets" and "developing countries" are similar because their semantic contents (the subset of countries) are very similar. Another example, "Google" and "Microsoft" are similar because they are both software companies. However, "car" and "journey" are not semantically similar but related because "car" is a transport means for the activity "journey".

Specifically, semantic similarity is defined by some measure of distance between two terms on is a taxonomy. It is clear that "car" and "journey" are quite far away from each other in an is a taxonomy from WordNet. Semantic similarity is a more specific relationship and is much harder to model than relatedness (which can be modelled by term co-occurrence). Recent work on term similarity can be roughly classified into two main categories: knowledge based and corpus based.

Knowledge based approaches rely on handcrafted resources such as thesauri, taxonomies or encyclopaedia's, as the context of comparison. Most work in this space depends on the semantic is a relations in WordNet which is a manually curated lexicon and taxonomy. Corpus based approaches work by extracting the contexts of the terms from large corpus and then inducing the distributional properties of wordsorn-grams. Corpus can be anything from web pages, web search snippets to other text repositories.

II. METHODOLOGY

Ambiguous Segmentation

"April in Paris lyrics" vs. "vacation April in Paris"

Both a term and its sub-terms can be contained in the vocabulary, leading to multiple possible segmentations for a short text. However, a valid segmentation should maintain semantic coherence. For example, two segmentations can be derived from "April in Paris lyrics", namely {April in pairs lyrics} and {April pairs lyrics}. However, the former is a better segmentation according to the knowledge that "lyrics" is more semantically related with songs ("April in pairs") than months ("April") or cities ("pairs").

Noisy Short Text

"new York City" vs. "nyc" vs. "big apple"

In order to find the best segmentation for a given text by considering semantic coherence, we first need to extract all candidate terms. It can be easily and efficiently done by building a hash index on the entire vocabulary. However, short texts are usually informal and error-prone, full of abbreviations, nicknames, misspellings, etc. For example, "new York city" is usually abbreviated to "nyc" and known as "big apple". This calls for the vocabulary to incorporate as much information about abbreviations and nicknames as possible. Meanwhile, approximate term extraction is also required to handle misspellings in short texts.

Ambiguous Type

"Pink [e](singer) songs" vs. "pink[ad j] shoes"

The tag terms with lexical types (i.e., POS tags) and semantic types (i.e., attribute, concept, and instance). It will explain why we consider these types and how they contribute to short text understanding. A term can belong to several types, and its best type in a short text depends on context semantics. For example, "pink" in "pink songs" refers to a famous singer and thus should be labelled as an instance, whereas it is an adjective in "pink shoes" describing the colour of shoes. Traditional POS taggers determine lexical types based on linguistic rules or lexical and sequential probabilities learned from labelled corpora.

Ambiguous Instance

"Read harry potter [e](book) " vs. "watch harry potter [e](movie) " vs. "age harry potter [e](character) "

An instance (e.g., "harry potter") can belong to multiple concepts (e.g., book, movie, character, etc.). It can retrieve such one-to-many mappings between instances and concepts directly from existing knowledgebase. However, instances might refer to different concepts when context varies. Some methods attempt to eliminate instance ambiguity based on similar or related instances, but the number of instances that can be retrieved from a short text is usually limited, making these methods inapplicable to instance disambiguation in short texts. An observe that other terms, such as verbs, adjectives, and attributes, can also help with instance disambiguation.

III. IMPLEMENTATION

Vertex Cover approximation algorithm is used to do faster than the existing algorithm; also it will produce and maintain better accuracy and efficiency. This algorithm handles the data in parallel to reduce the computation time. Text Feature Extraction Algorithm is used to extract or find out the features in the text input using semantic analysis with source WordNet. This algorithm provides best accuracy than the Chain model and Pairwise Model. Fuzzy weighted Vote algorithm is the combination of Fuzzy logic and weighted vote algorithm, which means it generates the fuzzy score and then based on this score the weight is calculated during shortening the text.

A properly performing majority voting with a given consensus-threshold chooses at unique, from the agreed voter information, where a majority exists. The outcome of in exact majority and fuzzy weighted-average voters for all contract voting periods are identical. This implication leads us to introduce a novel voter that is a combination of majority and weighted-average voters. Majority voting used in agreement cases and Score-based fuzzy weighted-average voting used in disagreement cases.

The voter is less complex and quicker than the weighted-average voter, since in the majority of the cases it does not perform the relatively time Consuming weighted averaging procedure. Moreover, the use of the majority algorithm in agreeing voting cycles of the novel voter improves its whole safety level compared to the standard weighted-average voter. This effect brings us to present a novel voter that is a combination of majority and fuzzy weighted-average voters.

This score based fuzzy voting algorithm is useful for multi-state safety-critical systems. It improves the both safety and availability. In all case this algorithm gives good results both small errors and large errors. These findings cause for the need of a voting criteria which can merge all the benefits of the interviewed voting methods, but can prevent the disadvantages of all of them. Inspiration of the perform here is to apply that desired voting criteria, which can provide excellent accessibility and safety performances in all kinds of mistake circumstances and enhances the overall reliability and stability of the program.
Modules scores are computed to avoid the classical fuzzy rules used for inference. These scores can be used as weighted vectors directly instead of computing several rule outputs.

This voting algorithm uses the concept of the dynamic threshold. It gives all advantages of Majority voting and fuzzy weighted- average voting strategies in the case of safety and availability performances. This voting algorithm is useful for multi-state safety-critical system. This voting technique is useful for both permanent and intermittent errors. This algorithm is used to provide an error masking capability in safety-critical systems and to hide the occurrence of errors from the system output. This voter can be used in any safety critical system without having much information about the system, data and range of data.

The Majority voter with high level of safety has usually a low stage of accessibility and the Fuzzy weighted-average gives advanced stage of availability in the cost of low safety and higher safety than the

standard weighted-average voting algorithm. In this combine these properties, take the advantages of both voters. The experimental results revealed that the novel voter has higher safety performance than the majority and fuzzy weighted-average voting algorithms.

IV. EXPERIMENTAL RESULT AND ANALYSIS

It compare existing and proposed algorithms are Randomized approximation algorithm, Weighted Vote algorithm, Fuzzy weighted Vote algorithm. Here i take five different classes are as Accuracy, F-measure, Precision. I collect number of related different classes. To make the raw text valuable, that is to prepare the text, considered only the keywords. That is unnecessary words and symbols are removed. For this keyword extraction process to drop the common unnecessary words like am, is, are, to, from .etc. and also dropped all kinds of punctuations and stop words. Singular and plural form of a word is considered same.

Graph Result

Precision

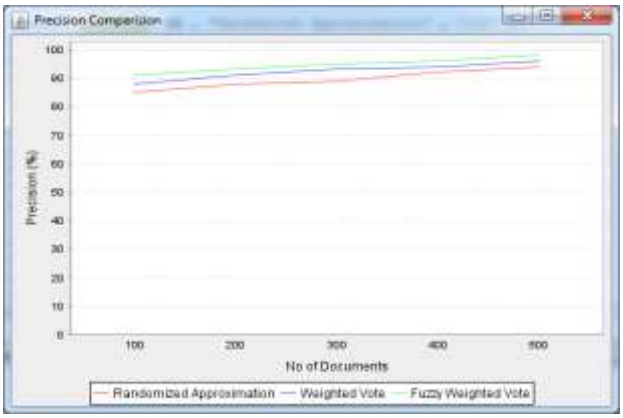

Fig1.Precision Comparison

F-Measure

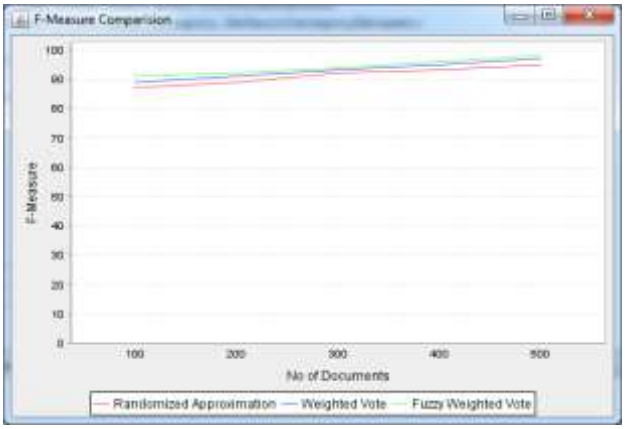

Fig2.F-measure Comaparison

ISSN: 2277-9655
Impact Factor: 4.116
CODEN: IJESS7

Accuracy

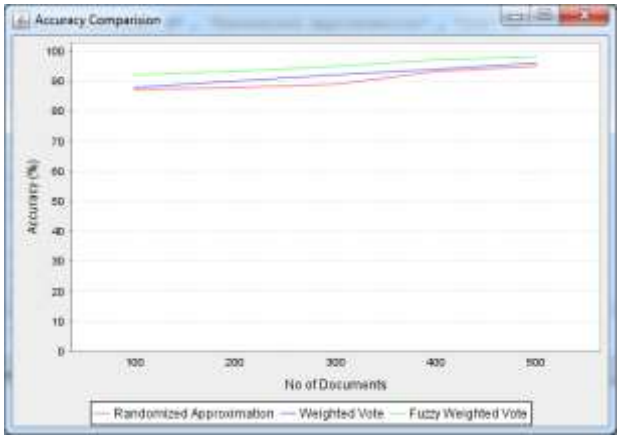

Fig3. Accuracy

This technique presented an efficient technique for web page classification. This technique will be more effective is the training set is set in such a way that it generates more sets. Though the experimental results are quite encouraging, it would better if the work with larger data sets with more classes. The existing technique requires more or less data for training as well as less computational time of these techniques.

V. CONCLUSION

It presented an effective approach for semantic similarity between terms with any multi-word expression. Using a large scale semantic network automatically acquired from billions of web documents.

.And used Fuzzy weighted Vote algorithm is the combination of Fuzzy logic and weighted vote algorithm, which means it generates the fuzzy score and then based on this score the weight is calculated during shortening the text. This method can collectively infer the referent entities of all name mentions in the same document. By modelling and exploiting the global interdependence between different EL decisions, the proposed method can achieve competitive performance over the traditional methods. More importantly, the model enables probabilistic inference between concepts and instances which will benefit a wide range of applications that require text understanding.

The results revealed that the novel voter has higher safety performance than the Majority and fuzzy weighted voting algorithms. This voting criterion can provide almost accurate safety and also a good range of accessibility. It have proved that in problems where the weighted voting distinguish some alternatives and finds the best alternative. Reduced Computation time comparing to other previous process and schemes.

VI. REFERENCES

[1] A. McCallum and W. Li, “Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons,” in Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003 - Volume 4, ser. CONLL ’03, Stroudsburg, PA, USA, 2003, pp. 188–191.
[2] G. Zhou and J. Su, “Named entity recognition using an hmm-based chunk tagger,” in Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, ser. ACL ’02, Stroudsburg, PA, USA, 2002, pp. 473–480.
[3] D. M. Blei, A. Y. Ng, and M. I. Jordan, “Latent dirichlet allocation,” J. Mach. Learn. Res., vol. 3, pp. 993–1022, 2003.
[4] R. Mihalcea and A. Csomai, “Wikify! linking documents to encyclopedic knowledge,” in Proceedings of the sixteenth ACM conference on Conference on information and knowledge management, ser. CIKM ’07, New York, NY, USA, 2007, pp. 233–242.
[5] S. Kulkarni, A. Singh, G. Ramakrishnan, and S. Chakrabarti, “Collective annotation of wikipedia entities in web text,” in Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, ser. KDD ’09, New York, NY, USA, 2009, pp. 457–466.

[6] X. Han and J. Zhao, "Named entity disambiguation by leveraging wikipedia semantic knowledge," in Proceedings of the 18th ACM conference on Information and knowledge management, ser. CIKM '09, New York, NY, USA, 2009, pp. 215–224.
[7] "Structural semantic relatedness: A knowledge-based method to named entity disambiguation," in Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, ser. ACL '10, Stroudsburg, PA, USA, 2010, pp. 50–59.
[8] X. Han, L. Sun, and J. Zhao, "Collective entity linking in web text: A graph-based method," in Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, ser. SIGIR '11, New York, NY, USA, 2011, pp. 765–774.
[9] W. Shen, J. Wang, P. Luo, and M. Wang, "Linden: Linking named entities with knowledge base via semantic knowledge," in Proceedings of the 21st International Conference on World Wide Web, ser. WWW '12, New York, NY, USA, 2012, pp. 449–458.
[10] C. Li, J. Weng, Q. He, Y. Yao, A. Datta, A. Sun, and B.-S. Lee, "Twiner: Named entity recognition in targeted twitter stream," in Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval, ser. SIGIR '12, New York, NY, USA, 2012, pp. 721–730.
[11] D. M. de Oliveira, A. H. Laender, A. Veloso, and A. S. da Silva, "Fsner: A lightweight filter-stream approach to named entity recognition on twitter data," in Proceedings of the 22nd International Conference on World Wide Web, ser. WWW '13 Companion, Republic and Canton of Geneva, Switzerland, 2013, pp. 597–604.
[12] P. Ferragina and U. Scaiella, "Tagme: On-the-fly annotation of short text fragments (by wikipedia entities)," in Proceedings of the 19th ACM International Conference on Information and Knowledge Management, ser. CIKM '10, New York, NY, USA, 2010, pp. 1625–162.

CITE AN ARTICLE

Umanandhini, D., & Manimegalai, S. (n.d.). FUZZY SCORE BASED SHORT TEXT UNDERSTANDING FROM CORPUS DATA USING SEMANTIC DISCOVERY. *INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY, 6*(12), 268-273.