International Journal of Early Childhood Special Education (INT-JECSE)
DOI:10.9756/INTJECSE/V14I5.582 ISSN: 1308-5581 Vol 14, Issue 05 2022

Finding the Short Text Knowledge Using Amalgamative Text Mining Classification

#1.KANNEGANTI MOUNIKA, M.TECH SCHOLAR, C.S.E DEPT, NRI INSTITUTE OF TECHNOLOGY

mounikakanneganti2@gmail.com

#2. Mr.NARENDRA BABU PAMULA, ASSOCIATE PROFESSOR, C.S.E DEPT, NRI INSTITUTE OF TECHNOLOGY

#3. Dr. D.SUNEETHA ,HEAD OF THE DEPARTMENT, DEPARTMENT OF C.S.E,NRI INSTITUTE OF TECHNOLOGY

ABSTRACT

As text semantics has an important role in text meaning, the term semantics has been seen in a vast sort of text mining studies. However, there is a lack of studies that integrate the different research branches and summarize the developed works. This paper reports a systematic mapping about semantics-concerned text mining studies. This systematic mapping study followed a well-defined protocol. Its results were based on 1693 studies, selected among 3984 studies identified in five digital libraries. The produced mapping gives a general summary of the subject, points some areas that lacks the development of primary or secondary studies, and can be a guide for researchers working with semantics-concerned text mining. It demonstrates that, although several studies have been developed, the processing of semantic aspects in text mining remains an open research problem.

INTRODUCTION

Text mining is concerned with the task of extracting relevant information from natural language text and to search for interesting relationships between the extracted entities. Text classification is one of the basic techniques in the area of text mining. It is one of the more difficult data-mining problems, since it deals with very high-dimensional data sets with arbitrary patterns of missing data.

The problem address in this project was to learn to classify (guess the category/group) unlabeled text documents. The problem would be solved by taking a large set of labeled (the category/group of the document) text documents and building a

data mining classifier from those documents. The data mining classifier would then be able to classify an unlabelled example based on the information learned from the labelled examples. The wide availability of web documents in electronic forms requires an automatic technique to label the documents with a predefined set of topics, what is known as automatic Short Text Categorization (TC). Over the past decades, it has been witnessed a large number of advanced machine learning algorithms to address this challenging task. By formulating the TC task as a classification problem, many existing learning approaches can be applied, but its limitations are exposed when the underlying text is very small.

A wide range of applications handle short texts. For example, news recommendation system need to process the news titles which may be not strictly syntactical; in web search, queries consist of a very small number of keywords. Short texts introduce new challenges to many text related tasks including information retrieval (IR), classification, and clustering's. Unlike long documents, two short texts that have similar meaning do not necessarily share many words. For example, the meanings of "upcoming apple products" and "new iphone and ipad" are closely related, but they share no common words. The lack of sufficient statistical information leads to difficulties in effectively measuring similarity, and as a result, many existing text analytics algorithms do not apply to short texts directly.

More importantly, the lack of statistical information also means problems that can be safely ignored when we handle long documents become critical for short texts. Take polysemy as an example. The word "apple" gives rise to different meanings in "apple product" and "apple tree". Due to the scarcity of contextual information, these ambiguous words make short texts hard to understand by machines. In this paper, we propose a novel approach for understanding short texts. Our approach has two components: i) a semantic network based approach for enriching a short text; and ii) a deep neural network (DNN) based approach for revealing the semantics of a short text based on its enriched representation.

1.2 PURPOSE OF THE PROJECT

Data analysis is a complex process that consists of finding a suitable data representation, a suitable machine learning method, and using a suitable evaluation metric (one that reflects what the user is really interested in). All these choices are crucial from the "planning to learn" perspective, and none are trivial.

We propose a mechanism to semantically enrich short texts using Probase. Given a short text, we first identify the terms that Probase can recognize, then for each term we perform conceptualization to get its appropriate concepts, and further infer the co-occurring terms. We denote this two-stage enrichment mechanism as Concepts-and- Co-occurring Terms (CACT). After enrichment, a short text is represented by a set of semantic features and is further denoted as a vector that can be fed to our DNN model to do semantic hashing. We give related definitions as below:

Semantic features—the elementary unit terms that compose (enriched) short texts. The semantic features include original terms, those inferred concepts and co-occurring terms.

Semantic feature vocabulary—a vocabulary that consists of top-k (e.g., 3,000) most frequent semantic features obtained from the whole training data set.

Semantic feature vector—a k-dimensional vector that represents a (enriched) short text, each element of the vector is the count of a semantic feature that occurs in the short text.

Next, we introduce how to produce these semantic features for short texts using our proposed enrichment mechanism CACT. Note that in this paper, we focus on conceptualization and inferring co-occurring terms (do semantic enrichment) for noun phrases. Verbs and adjectives are also important as they can be useful for disambiguation and other tasks. We still remain them as the semantic features for the short text.

MODEL DESIGN: Problem Definition:

Assume there are n training short texts in the dataset, denoted as: X ¼ fx1; x2; . . . ; xng 2 Rk_n, where k is the dimensionality of the feature vector, and the row is indexed by position of the semantic feature in the vocabulary. The value xi j represents the frequency of feature j occurs in the (enriched) short text i. The objective of our model is to learn optimal binary hashing codes Y ¼ fy1; y2; . . . ; yng 2 f0; 1gm_n for the training set X, and a hashing function f : Rm ! f0; 1gm, which maps each short text to its hashing

codes with m bits (i.e., yi ¼ fðxiÞ). The learning model we design, a four-layer deep neural network. The input layer is fed by a feature vector of one short text and the output represents the learned binary code for that text.

We further propose a two stage semi-supervised approach to train the DNN model to make it capture semantic features from the input and finally encode those features into the optimal binary hashing codes. Specifically, we first treat the DNN model as a stacked auto-encoders, where each layer (except the input layer) is both a hidden layer of current auto encoder and a input layer of next auto-encoder we then greedily train each auto-encoder with back propagation method. We call this stage pre-training procedure, and it is totally unsupervised. The philosophy of this design rests in the concept of stacking. The first auto-encoder tends to learn primary features of the raw input data and the following ones further capture more abstract features from its underlying input.

After pre-training, the model is further trained as a whole network through a supervised method which aims to predict the label information (e.g., the tag or category) of input data, and we call this stage fine-tuning process. Once the training process completes, the model is used to do semantic hashing for a given text s. As the raw output is a real valued vector, we further use a threshold to make the output to be binary such that 1) the codes preserve similarity which indicates that semantically similar short texts should be mapped to similar hashing codes within a short Hamming distance, and 2) each bit has an equal probability as positive (the bit is 1) or negative (the bit is 0). For clarity, in the following sections, we introduce the architecture details of our DNN model in terms of the training stages, that is, the pre-training and fine-tuning.

Before start training the model, we randomize all the parameters, including weights W and bias b, using Gaussian distribution, where the mean and standard deviation is (0, 0.01). In the optimization of network's parameters, instead of updating the parameters only after scanning all the training examples, we usually perform the update once an example is processed, and we call this method stochastic gradient descent (SGD) In our work, to further improve.

The distribution of hidden features on third auto-encoder. The fine-tuning architecture of DNN model. training efficiency, we divide the data into small mini-batches, which consists of a fixed number of training examples (we set the number to 200 and 1,000 for pre-training and fine-tuning respectively), and the parameters are updated by the average derivatives of loss J with respect to each mini-batch. We call this method Mini-batches gradient descent and it can significantly speed up the training process because it computes the derivatives for many training examples simultaneously using matrix-matrix multiplies in memory.

PRELIMINARIES:

In this section, we describe

Probase, a large-scale semantic network with millions of fine-grained, interconnected, and probabilistic concepts,

ii) back propagation, an effective method to train neural network, and

iii) auto-encoder, an unsupervised learning algorithm that learns features from unlabeled data

Probase is a large-scale probabilistic semantic network that contains millions of concepts of worldly facts. These concepts are harvested using syntactic patterns (such as the Hearst patterns) from billions of webpages. For each concept, it also finds its instances and attributes. For example, company is a concept, and it is connected to instances such as apple and microsoft. Moreover, Probase scores the concepts and instances, as well as their relationships. For example, for each concept c and each of its instance t, Probase gives pðtjcÞ, the typicality of instance t among all instances of concept c, and pðcjtÞ, the typicality of concept c among allconcepts that have instance t.

Furthermore, Probase gives pðt1jt2Þ, the co-occurrence probability between the two instances t1 and t2. In addition, there are many other probabilities and scores in Probase. These abundant information allows us to build inferencing mechanisms for text analysis and text understanding. Compared with other knowledge bases such as Word- Net, Wikipedia, and ODP, Probase has two advantages. First, the rich concept information enables interpretation at fine levels. For example, given “China, India”, the top concepts Probase returns are country, Asian country. Given “China, India, Brazil”, the top concepts become developing country, BRIC country, emerging market.

Other knowledge bases do not have a fine-grained concept space, nor an inferencing mechanism for the concept, therefore they can at most map them into the concept of country, which is often too general and coarse level for sophisticated text understanding.

Second, the probabilistic nature of Probase allows us to build inferencing mechanisms to map words in a context to appropriate fine-grained concepts. This allows us to perform text analytics in the concept space, which contains much richer information than the original short text, which is often sparse, noisy, and ambiguous.

Proposed Work

The challenging problem of inducing a taxonomy from a set of keyword phrases instead of large text corpus is the current project context. We propose a multiple inferencing mechanism called conceptualization to get the most appropriate sense for a term under different contexts. The concept space we employ is provided by Probase API which contains millions of fine-grained, interconnected, probabilistic concepts. The concept information is more powerful in capturing the meaning of a short text because it explicitly expresses the semantics. However, conceptualization alone is still not enough for tasks such as comparing two short texts or classifying short texts. Consider the same two short texts: “upcoming apple products” and “new iphone and ipad”.

After conceptualization, we get a set of concepts for each short text but there are still no common terms. To reveal the similarity of their semantics, we further built an inferencing mechanism on Probase to extract certain popular co-occurring terms for each original noun term, and these can be seen as new contexts for that short text. We first do clustering on the Probase terms based on their co-occurrence relationship, after that, we determine whether an entity belonged to same cluster. For instance, If we have a keyword "dogs", our Probase driven extracts other polysemy words like "mutt", "canines", "mongrel" etc which definitely forms a cluster group and we shall repeat the process for other nouns in the short text and from the obtained results we shall identify the most commonest matching entities using a 3-layer stacked auto-encoders for hashing terms to reduce processing complexity.

Cosine similarity coefficient, a measure that is commonly used in semantic text classifications which measures the similarity between two texts and determines the probable measure.

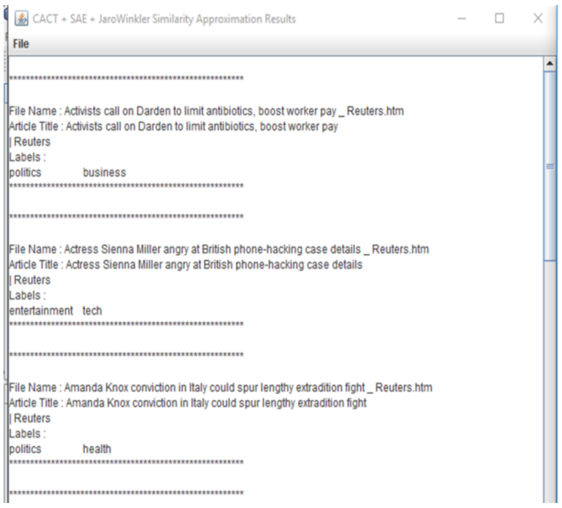
CACT's approach to use Cosine's similarity co-efficient increases time complexity exponentially.

So we propose to replace Cosine's similarity coefficient with Jaro Winkler similarity measure to obtain the similarity matching of text pairs(source text and destination text).

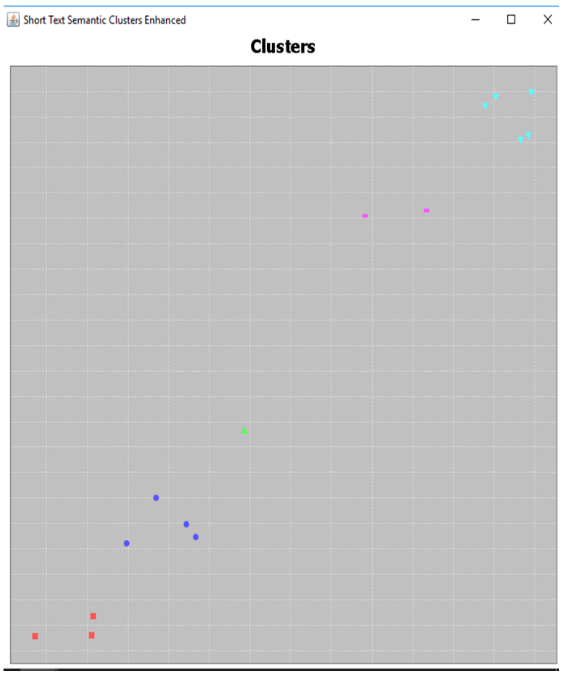
Jaro-Winkler does a much better job at determining the similarity of strings because it takes order into account using positional indexes to estimate relevancy.

It is presumed that Jaro-Wrinkler driven CACT's performance with respect to one-to-many data linkages offers an optimized performance compared Cosine driven CACT's workings.

An evaluation of our proposed concept suffices as validation.



```
******************************************************

File Name : Activists call on Darden to limit antibiotics, boost worker pay _ Reuters.htm
Article Title : Activists call on Darden to limit antibiotics, boost worker pay
| Reuters
Labels :
politics          business
******************************************************

******************************************************

File Name : Actress Sienna Miller angry at British phone-hacking case details _ Reuters.htm
Article Title : Actress Sienna Miller angry at British phone-hacking case details
| Reuters
Labels :
entertainment   tech
******************************************************

******************************************************

File Name : Amanda Knox conviction in Italy could spur lengthy extradition fight _ Reuters.htm
Article Title : Amanda Knox conviction in Italy could spur lengthy extradition fight
| Reuters
Labels :
politics          health
******************************************************

******************************************************
```



CONCLUSION

In this paper, we propose a novel approach for understanding short texts. First, we introduce a mechanism to enrich short texts with concepts and co-occurring terms that are extracted from a probabilistic semantic network, known as Probase. After that, each short text is represented as a 3,000- dimensional semantic feature vector. We then design a more efficient deep learning model, which is stacked by three auto-encoders with specific and effective learning functions, to do semantic hashing on these semantic feature vectors for short texts. A two-stage semi-supervised training strategy is proposed to optimize the model such that it can capture the correlation ships and abstract features from short texts. When training is done, the output is thresholded to be a 128-dimensional binary code, which is regarded as a semantic hashing code for that input text. We carry out comprehensive experiments on short text centered tasks including information retrieval and classification. The significant improvements on both tasks show that our enrichment mechanism could effectively enrich short text representations and the proposed auto-encoder based deep learning model is able to encode complex features from input into the compact binary codes.

REFERENCES

1. P. Li, H. Wang, K. Q. Zhu, Z. Wang, and X. Wu, "Computing term similarity by large probabilistic is a knowledge," in Proc. 22nd ACM Int. Conf. Conf. Inf. Knowl. Manage., 2013,
2. D. Kim, H. Wang, and A. H. Oh, "Context-dependent conceptualization," in Proc. 23rd Int. Joint Conf. Artif. Intell., 2013, pp. 2654–2661.
3. W. Wu, H. Li, H. Wang, and K. Q. Zhu, "Probase: A probabilistic taxonomy for text understanding," in Proc. Int. Conf. Manage. Data, 2012, pp. 481–492.
4. Y. Song, H. Wang, Z. Wang, H. Li, and W. Chen, "Short text conceptualization using a

probabilistic knowledge base,” in Proc. 22nd Int. Joint Conf. Artif. Intell., 2011, pp. 2330–2336.

5. Y. Bengio, “Learning deep architectures for ai,” Found. Trends Mach. Learn., vol. 2, no. 1, pp. 1–127, 2009.
6. R. Salakhutdinov and G. E. Hinton, “Semantic hashing,” Int. J. Approx. Reasoning, vol. 50, no. 7, pp. 969–978, 2009.
7. W. Wang, C. Xiao, X. Lin, and C. Zhang, “E_cient approximate entity extraction with edit distance constraints,” in Proceedings of the 2009 ACM SIGMOD International Conference on Management of Data, ser. SIGMOD ’09, New York, NY, USA, 2009, pp. 759–770.
8. 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.
9. https://dataaspirant.com/wp-content/cache/page_enhanced/dataaspirant.com/2015/04/11/five-most-popular-similarity-measures-implementation-in-python/_index.html_gzip
10. https://docs.oracle.com/cd/B28359_01/datamine.111/b28129/process.htm
11. http://prize2630.aourprize1.loan/?utm_medium=NQ3aDvyuBCtafRQJPeFC66tm%2bMNW8T%2baflxP0d0AJGo%3d&t=main5
12. https://www.google.co.in/search?q=data+mining+tutorial&sa=X&ved=0ahUKEwjx_4bis9_bAhUSWX0KHdDZBPAQ1QII5wEoBw&biw=1366&bih=662#
13. http://program.niit.com/niitpgpjava?utm_siteid=Google_NIITJavaIntentHyderabadBMMRx&utm_adunit=text&utm_banner={banner}&utm_keyword=&utm_device=c&gclid=EAIaIQobChMI8___u7Xf2wIVkQ4rCh2ynARPEAAYASAAEgKC8vD_BwE
14. http://www.java67.com/2017/01/12-advanced-java-programmingbooksforexperienced programmers.html
15. https://www.quora.com/Which-is-the-best-online-course-and-book-to-learn-Java-programming-at-home-Which-tutorials-can-help-me
16. http://iiti.ac.in/people/~tanimad/JavaTheCompleteReference.pdf
17. http://www.oracle.com/technetwork/java/effectivejava-136174.html