Semantically Enriched Weighted Word Embedding for Short Text Representation

Sathya Priya C Abarna E Nandini V A¹ and Vijayarani J²

¹Corresponding author email id: vjji@cs.annauniv.edu

Date of publication (dd/mm/yyyy): 06/07/2017

Abstract – Short texts are very noisy and sparse in the use of vocabulary. Short text clustering is a challenging task. The scarcity of data and the semantic sensitivity to context affect the classification of short text. To cluster the short texts by their meaning, more semantics must be added to short text. Concepts and co-occurring terms for each term in the short text obtained from a probabilistic knowledge base like Probase can enrich semantics. A method is constructed based on semantic and frequency information to arrive at low dimensional representation for short texts. For this purpose, an aggregated weighted word embedding representation called Semantically Enriched Weighted Word Embedding (SEWWE) for Short Text Representation is designed. The proposed method outperforms other tf-idf based representations.

Keywords – Knowledge Base, Probase, Semantic Similarity, Semantic Enrichment, Word Embedding.

I. INTRODUCTION

Short pieces of text reach us every day through the use of social media such as twitter, newspaper headlines and texting. They are very noisy and sparse. Short texts introduce new challenges to many text related tasks including information retrieval, classification and clustering. Two different short texts with same context may share no common words which makes machine hard to understand. Many approaches have been proposed for short text understanding. Additional semantic information is needed to effectively understand the short texts. A short text can be enriched with external knowledge resources like WordNet, Wikipedia etc. These have their own limits. For example, WordNet does not contain information for proper nouns and does not weigh senses based on their usage which often gives rise to misinterpretation of short texts.

The representation of short texts should grasp most of the semantic information in that fragment which is achieved through effective vector representation of that short text. Usual tf-idf is not suitable for representing short texts because it is based on exact word overlap. So every word in the short text is projected into a semantic vector.

Machines are not able to grasp the concepts behind the words in a short text, nor do they know the properties and relationships associated with the concepts. In particular, using statistical approaches to find exact context behind the short text is often infeasible, as short text does not have enough content from which we can build a meaningful model. Hence there is a need to provide a probabilistic framework for understanding the short text, which includes a knowledge base and certain inference techniques to enable machines to perform human-like conceptualization.

A model is developed which enables us to derive the most likely concepts from a set of words or a short piece of text. The concepts in Probase, a universal probabilistic ontology with 2.7 million concepts harnessed automatically from a corpus of 1.68 billion web pages. They are more consistent with humans common knowledge. Word embeddings, a vector representation of terms are used to represent terms in a semantic space in which proximity of vectors can be interpreted as semantic similarity (i.e). The words nearer to each other in the semantic space are more similar compared to the words which are far apart.

Semantics in the short text is largely ignored in the mining process and mining results often have low interpretability. One particular challenge faced by such approaches lies in short text understanding, as short texts lack enough content from which statistical conclusions can be drawn easily. Short text is semi structured. These short text have completely different textual format, in which the used language is full of slang, hash-tags and spelling errors. Unlike normal text, the mining algorithms cannot be applied directly. The exact context behind the short text is harder to predict. Hence we improve text understanding by using a probabilistic knowledgebase that is rich in terms of the concepts it contains.

The main objective is to find out semantic similarity between two texts i.e., if two pieces of text mean the same thing. To add more semantics to short texts, additional knowledge sources are used. For getting semantically enriched short text, an effective vector representation is found through weighted word embedding.

II. RELATED WORK

De Boom, et al [2], designed a novel technique to aggregate both tf-idf and word embedding for short text representation and investigated the effectiveness of naive techniques, as well as traditional tf-idf similarity for short text fragments of different lengths. They represented both the fixed and variable length short texts by weighing word embed-dings. Weight determined from the idf value is assigned to each word in the short text. Every embedding vector for each sorted word is multiplied with the weight and averaged to get a single text representation. They evaluated by finding the Split error and JS Divergence for Twitter data using Wikipedia embeddings. The proposed method (SEWWE) performs conceptualization using Probase to get related concepts and co-occurring terms of the concepts. Hence the weighted vectors of words, concepts (from Probase) and co-occurring terms (from
De Boom, et al [3], published a work that combines word embedding and Tf-idf information for semantic contents within very short text fragments. To experiment their work they used the Wikipedia dump of March 4th, 2015, after cleaning the articles by removing markup and punctuation. This method considered only nonstop words, evaluated their distance measure using k-NN classification and learned weights for entire word vectors instead of separate dimension. The proposed method also combines word embedding with knowledge base but the difference is in the knowledge based used ie. Probase.

Wu Wentao, et al [4], presented a novel iterative learning framework that aims at acquiring knowledge and highlights the need to enable machines to better understand electronic text in human language. To overcome this, they devised a universal, probabilistic taxonomy (Probase) that is more comprehensive than existing ones with the highest precision of 92.8%. Each fact or relation is associated with some probabilities to measure its plausibility and typicality. It is the largest general-purpose taxonomy fully automatically constructed from HTML text on the web. It is a novel semantic search prototype to handle semantic queries and their evaluation shows that 80% of the returned result are considered as relevant by users, compared with less than 50% of those result from searching the original queries on Google or Bing. Also they enabled machines to conceptualize from a set of words by performing Bayesian analysis and performs K-means clustering.

Li Peipei, et al [5], proposed an effective approach for semantic similarity between terms with any multi-word expression (MWE) and introduced a clustering approach to orthogonalize the concept space in order to improve the accuracy of the similarity measures. This paper gives detailed description about the features of Probase. WordNet does not cover all possible meanings of multi-word expression. Corpus based approaches too are not suitable for finding the similarity of MWE. They explained the use of Probase for finding more meaningful similarity with better coverage. They used M & C datasets for evaluating word similarity and created their own datasets for evaluating MWEs. The main objective of proposed method is to use Probase for finding semantically enriched short text representation.

Kim, Dongwoo, Haixun Wang, and Alice H. Oh [6], proposed a corpus-based framework for context-dependent conceptualization by combining Latent Dirichlet Allocation, a widely used topic model with Probase, a large-scale probabilistic knowledge base, which directly defines to learn the context sensitive concept distribution over the given sentences and phrases. LDA was employed to discover the topics by computing and maximizing the posterior distribution of topics from the documents. They combined concept vector of words to get the sentence vector either by assigning equal weight or by assigning tf-idf weight. The evaluation of conceptualization was done with URL query similarity of click through data and random pairs. The proposed work performs conceptualization similar to this, but it also includes co-occurring terms obtained from Probase and assigns tf-idf weight.

Tom Kenter and Maarten de Rijke [7], proposed to move from word-level to text-level semantics by combining insights from methods based on external sources of semantic knowledge with word embeddings. They derived multiple types of meta-features from the comparison of the word vectors for short text pairs and from the vector means of their respective word embeddings. The proposed method differs from this with the use of knowledge base Probase. Song, Yangqiu, et al [8], introduced a method of conceptualizing short text using a probabilistic knowledgebase. They detected and mapped terms in short text to instances and attributes in the knowledgebase and also derived the most likely concepts using Bayesian inference. But they didn't combine the short text representation as done in the proposed work.

Wang, Zhongyuan, et al [9], demonstrated the advantage of using a knowledge base (Probase) for search. Unlike traditional knowledge base that treats knowledge as black and white, it supports probabilistic interpretations of the information it contains and also enables it to incorporate heterogeneous information. The well known Freebase contains no more than 2000 concepts, but Probase has millions of concepts obtained by the study of 2 years worth of Microsoft's Bing search log and found 85% of the searches contains concepts and instances are exists in Probase.

Wang, Fang, et al [10], proposed a framework that measures the semantic similarities between short texts from the angle of concepts, so as to avoid surface mismatch and they require only few training data to learn concept model. They evaluated their result using Precision, Recall and F-value (harmonic mean of precision and recall). The proposed method also finds semantic similarity between short texts but only after enrichment.

III. PROPOSED SYSTEM ARCHITECTURE

The short text in social platform contains spelling errors, hashtags, urls, mentions and the used language is full of slang. Hence there is a need to enrich the short text using any one of the knowledge bases like Freebase, WordNet, ConceptNet and Probase. By assigning weights using idf values to the enriched short text, a vector representation is devised through word embedding (Figure 1).
Nearly 2000 short texts collected are preprocessed from which the terms available in Probase are identified. If it is a single term, (related concept term, probability, inverse document frequency) and (co-occurring term, probability, inverse document frequency) are extracted. In the case of multiple terms, conceptualization is done by three mechanisms Naive Bayes, Intersection and Collapsed Gibbs sampling.

(i) Naive Bayesian Mechanism is given by,
\[ P(c|E) = \frac{P(E|c)P(c)}{P(E)} \propto P(c) \prod_i P(e_i|c) \] (1)
where c-concept and entity E = e_i, i = 1, 2, ... M

(ii) Intersection mechanism is devised to avoid over generalization for input text that contain multiple concepts. To achieve this different terms are grouped for conceptualization and the results of different groups are combined using union operation.

(iii) To avoid disambiguity between conceptualized terms, Latent Dirichlet Allocation (LDA) with collapsed Gibbs sampling is used. Conditional probability of each term in short text is calculated by,
\[ P(Z_i = k|\vec{z}, C) \propto (\sum_w n_{wk} + \alpha) (C_w + n_{wk} + \beta) \cdot \frac{C_k + n_{wk} + \beta}{\sum_k C_k + N_w + N_{\beta}} \] (2)
where
- \( \vec{z} \) -Sequence of term indices
- \( C \) - Topic assignment vector
- \( C_{wk} \) - No of times term ‘\( w \)’ assigned to topic ‘\( k \)’.
- \( n_{wk} \) - No of times term ‘\( w \)’ assigned to topic of sentence S
- \( N \) - Size of vocabulary

Using conditional probability of terms, the probability of concept can be calculated by,
\[ P(c|W,\vec{z}) \propto P(c|W) \sum_k P_{wk}P_{ck} \] (3)
where
\[ P_{wk} = P(Z_i = k|\vec{z}, C), P_{ck} = \frac{C_{ck}^k + \beta}{\sum_k C_{wk}^k + N_{\beta}} \]

Weight is assigned to each term according to their idf value and sorted from high to low idf values. The sorted terms are multiplied with a weight (mean of all idf values). Weighted aggregation of vectors of terms (Equation 4) gives the enriched embedding of short text.

\[ l \in \alpha, \beta: t(c^i) = \frac{1}{n} \sum_{j=1}^{n} w_j c_j^i \] (4)

IV. EXPERIMENTAL RESULTS

The dataset containing 2000 short texts are tokenized and stemmed with porter stemmer algorithm. Using proba-se semantic network, the Probase terms are identified. The result obtained after preprocessing (stop word removal and stemming) of short texts are shown below.
conceptualization maps each term of short text mapped to set of concepts.

For multiple terms the conceptualization is done as below.

4.2 Co-Occurring Terms
As a part of semantic enrichment, co-occurring terms semantically related to context are obtained based on their co-occurrence probability.

4.3 Tf-idf
Tf-idf values of words, concepts and their co-occurring terms of short texts are given below.
4.4. Vectors-Terms
Word vectors computed are listed below

Fig. 7. Vector Value based on idf weight

4.5. Vectors-Short Text
For each short text, weighted word embedding is obtained.

Fig. 8. Weighted Word Embedding of short text

4.5. Test Cases
Test cases defined in table have been tested for different short texts and the results proved to be consistent. The Test Case TC1 have been tested to prove that the cosine similarity between two short texts increases after enrichment. The test case TC2 having similar set of short text does not need to perform enrichment as the cosine similarity values remains the same before and after enrichment. The Test Case TC3 is tested against dissimilar short texts.

<table>
<thead>
<tr>
<th>TC Id</th>
<th>Test Case Scenario</th>
<th>Before Enrichment</th>
<th>After Enrichment</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>a) Wall Street set to open lower as oil prices dip</td>
<td>-0.037</td>
<td>0.455</td>
</tr>
<tr>
<td></td>
<td>b) Oil prices fall on profit taking after hitting 2016 highs</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.</td>
<td>a) Investing on Arvind and UFO Moviez: AK Prabhakar, IDBI Capital Markets</td>
<td>0.97</td>
<td>0.99</td>
</tr>
</tbody>
</table>
(ii) Cosine Similarity: Cosine similarity metric finds the normalized dot product of the two attributes. By determining the cosine similarity, we would effectively try to find the cosine of the angle between the two objects. Cosine similarity is particularly used in positive space, where the outcome is neatly bounded in $[0, 1]$. One of the reasons for the popularity of cosine similarity is that it is very efficient to evaluate, especially for sparse vectors.

$$\text{sim}(x, y') = \cos(\theta) = \frac{x \cdot y'}{|x| |y'|}$$  \hfill (5)

(iii) Manhattan Distance: Manhattan distance is a metric in which the distance between two points is the sum of the absolute differences of their Cartesian co-ordinates i.e., it is the absolute sum of the difference between the x-coordinates and y-coordinates.

\[
\sum_{i=1}^{k} |x_i - y_i| \tag{7}
\]

In a plane with p1 at $(x_1, y_1)$ and p2 at $(x_2, y_2)$ Manhattan Distance

\[
|x_1 - x_2| + |y_1 - y_2| \tag{8}
\]

Fig. 11. Manhattan distance

VI. CONCLUSION

In this paper, we proposed a novel approach for understanding short texts. First, we introduced 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, an effective method is devised to derive vector representations for very short fragments of text. For this purpose weighted word embeddings are learnt based on their idf value. Finally, the weighted vectors are averaged to arrive at a single vector representation for each short text. Standard cosine distance is used to calculate the similarity between a pair of vectors. In future, the work can be enhanced by designing a more efficient deep learning model, which is stacked by auto-encoders with specific and effective learning functions, to perform semantic hashing on these semantic feature vectors for short texts. Comprehensive experiments can be carried out on short text centered tasks including information retrieval and classification.

REFERENCES


conference on Conference on information and knowledge management, 2013, pp. 1401-1410. ACM.


AUTHORS' PROFILES

First [Sathya Priya C Abarna E Nandini V A]\(^1\)
B.E. (CSE), CEG Campus, Anna University, Chennai, Tamil Nadu, India.

Second [J.Vijayarani]\(^2\)
Teaching Fellow, Department of CSE, CEG Campus, Anna University, Chennai, Tamil Nadu, India. Area of specialization: Natural Language Processing, Text Mining, Semantic Computing.