Grenze International Journal of Engineering and Technology, Jan Issue

Representation Of Documents using Minimal Dictionary of Embeddings

Remya R.K. Menon[1], Gayathri S[2] and Amina A[3]
[1-3]Department of Computer Science and Applications, Amrita Vishwa Vidyapeetham, Amritapuri, India
Email: ramya@am.amrita.edu, {amscp2csc20024,amscp2csc20005}@am.students.amrita.edu

***Abstract*—With the rapid growth of the internet the availability of online information has also increased. As a result, it has become increasingly important to provide improved techniques for effectively and efficiently finding and representing textual content. Obtaining a representation of a document which can contain the semantic aspect of the document is always a hot topic of discussion. There are many methods to represent a document in vector form. But in many of these methods, the minimal representation of the document is not discussed. Dictionary learning is a method which can be used to obtain a minimal representation of documents. This paper presents a model that can obtain the minimal representation of documents using a sparse model and dictionaries of embedding. Two suitable techniques that we employ are SVD (Singular Value Decomposition) for dictionary learning and OMP (Orthogonal Matching Pursuit) for sparse coding.**

***Index Terms*— Dictionary learning, Document representation, Document retrieval, Sparse representation, Embedding, Keywords.**

I. INTRODUCTION

The growth in information and communication technology has provided accessibility to large volumes of data nowadays. This has resulted in an exponential increase in the number of documents available online. Therefore, the effective retrieval and mining of these text information are becoming impossible without efficient document content organization, summarizing, and indexing. As a result, document representation is critical in a wide range of real-world applications, including document retrieval, web search, and spam filtering. Regarding Natural Language Processing, text or document representation has always been an important research area. Document representation aims to encode all of a document's semantic content into a real-valued representation vector that can be used in subsequent activities.

The basic linguistic piece of natural language is generally a document. The user can access these documents or any other text contents on the internet through proper searching. When searching for a document, the mapping of the search keywords and document collections takes place. These need to be done based on semantic text similarity, then only the user gets a better result. Also updating this document collection by adding a new document needs to be checked. Both these aspects can achieve their best version through a minimal representation of documents. The initial step in this representation is the embedding of documents. Documents may contain words or text that is described by the ordering of its characters. They cannot be directly processed by machine for any NLP task, which should be in the form of numbers. To conduct further analysis, it must be converted from text to a real-valued vector.

Grenze ID: 01.GIJET.9.1.49
© Grenze Scientific Society, 2023

There are numerous embedding methods, including Word2vec, the TF-IDF approach, the method of Doc2Vec, BERT, etc. The representation of the document is made using it. But one of the important points that have to be taken care of is that the embedding which is used to represent the document has to hold the semantic aspect of the document[8]. Semantics is the study of word meanings that enables computers to comprehend and interpret phrases, paragraphs, or entire documents. Semantically similar documents can be found by using semantically similar words[5]. Semantic reconstruction of the text data is found yet an open issue. Comparative studies on different vectorization techniques have been done in the past, but not more focused on their semantic representation. Representation of vectors based on the semantic aspects is one of the challenging problems faced by NLP (Natural Language Processing). The model of Traditional text representation aims to represent unorganized text documents numerically to compute them mathematically. The issue is how to describe text documents using explicit or implicit semantics rather than word usage or their occurrence.

Modelling the underlying semantic similarity of sentences or words can significantly improve operations like understanding text and retrieving information. A good model must be immune to changes in the language or syntax used to represent the same idea. The discovery of such a semantic textual similarity metric has sparked a lot of study attention. In this project, we are trying to obtain a minimal representation of documents with the help of a dictionary of embeddings. This model can be used to decompose the input sentence to its constituent relevant words.

Dictionary learning is the process of training a set of words to approximate a given document with a minimal combination of these atoms. It is called a 'dictionary' using which training data can be represented. This technique can be used in many applications such as Document summarizing, document retrieval and so on. In our study, we have particularly considered the use of a dictionary for the retrieval of constituent words in the input sentence. One of the applications of this model is document retrieval. Document retrieval is the process of acquiring the essential information from documents, when we are searching for a document from a collection of documents, Similarity checking, or Semantic matching should take place between the input query and the document collection. Therefore, the semantic significance of the document should be maintained in its representation.

We propose our document representation approach using the Dictionary of embeddings by considering the order of words and sentences.

II. Literature Review

Word embedding is the technique by which words are mapped to real-valued vectors. Concept of word embedding was introduced in 2003 and used it in combination with the model's parameters to train a neural language model [2]. Word embedding approaches based on co-occurrence statistics have shown to be quite good at obtaining the semantic and grammatical representation of words using a low-dimensional continuous vector [14]. There are many different types of embedding techniques such as Word2vec, Glove, BERT etc

TF-IDF was a relevant technique which was used commonly for vector representation of text. In the study conducted in [9], they analysed the precision of using various word representations for depression detection. As per their analysis, it is found that TF-IDF will give importance to the frequency of word occurrence in a document. Considering word ordering will improve tasks which are related to documents [6]. Word2vec considers the order in which the words appear in the documents [4] is an empirical evaluation of doc2vec with practical insights into Document Embedding Generation, which gives an evaluation of doc2vec by comparing the document embedding generated by doc2vec with another embedding which is word2vec. In 2014, doc2vec was suggested as an extension to word2vec to learn document-level embeddings. The study present an evaluation of doc2vec over two tasks. The experiment was carried out with both variants of doc2vec such as dbow and dmpv. In Word2vec it consists of two methods to learn the representation of Continuous Bag of Word (CBOW) and Skip-gram models. In CBOW, it predicts the centre word using the context words, while in Skip-gram, it predicts the context words using the centre word. But Glove uses the ratio of co-occurrence probabilities between words to estimate target words. [3]. BERT uses masked language modelling. BERT keeps some words masked and then forces the model to identify those words by understanding the context. When we use dictionary learning along with the vectorization method it will help us to represent these vectors are simply the linear combination of simpler word components. SentenceBERT performs well in capturing the semantics of even long sentences [11].

There are different ways of Document Representation discussed in the various paper. The concept of deep dictionary learning is one of them. In the paper called Deep Dictionary Learning [13], the study suggests deep dictionary learning as a means to combine the ideas of two paradigms, such as dictionary learning and deep

learning, and it demonstrates how using dictionary learning layers can result in deeper structures. Noisy Deep Dictionary Learning, another work on the same subject [10]. Instead of clean data, using augmented noisy data for training stacked autoencoders were included in the study. The training data in this study were corrupted with varying degrees of noise to increase resilience. One method for learning the representation layer is the Restricted Boltzmann Machine (RBM). For the recognition of images with a minimal amount of data, a coding network and deep dictionary learning was used [12]. This research intends to improve dictionary learning's capacity for deep representation. To achieve these goals, a new method of DDLCN (Deep Dictionary Learning and Coding Network), a deep dictionary learning technique for learning multi-layer deep dictionaries, brings together the benefits of dictionary learning with deep learning. DDLCN has shown early signs of promise. It was once used in place of conventional convolutional layers. Adopted the Classifier Algorithm of Support Vector Machine (SVM) and conducted numerous experiments (recognition of object hand-written digits and face) to assess the effectiveness of the proposed DDLCN. According to the studies conducted in the area of Deep Dictionary Learning, the majority of them are in the area of image classification and only a small number of studies have been conducted in this area.

Dictionary learning is a helpful method for extracting basic variables from several modalities [14]. A finite training set is used in classic dictionary learning. Initial research on learning dictionaries focused on building a foundation for representation. The dictionary atoms and loading coefficients were not constrained. To learn the basis optimal direction was used:

$$\min_{D,Z} \|X - DZ\|_F^2 \qquad (1)$$

where D stands for the dictionary, X for the input document, and sparse representation of the input document denoted by Z. Z must be sparse since the objective of problems requiring sparse representation is to learn a basis that can represent the samples sparsely. This is done using the widely utilised approach known as SVD. [13]. It is solved by the equation:

$$\min_{D,Z} \|X - DZ\|_F^2 \qquad (2)$$

such that

$$\|Z\|_0 \leq r \qquad (3)$$

The SVD process contains two main steps. It first learns the dictionary and then employs this knowledge in the second stage to generate a sparse representation of the input document [13]. Each input in the dictionary is called an atom. SVD is an effective method but the major problem with SVD is that it takes more time as the SVD (Singular Value Decomposition) has to be calculated in every iteration [13].

Document retrieval is one of the major applications of Dictionary Learning. The growth of technologies has resulted in the availability of information at our fingertips. Nowadays everything can be found on the Internet. No matter what the topic is, the information regarding that topic can be fetched through a web search. All we have to do now is enter a query for the topic to be searched. A method that compares two batches of text and assigns a score was traditionally used for search retrieval. The result will show how similar the
two texts are [7]. One of the most significant advantages of improved retrieval is that it increases the likelihood of finding relevant material on the internet. Sometimes the web search may not yield relevant results. This can happen because of three main reasons. First, the user's keywords could be related to multiple topics, thus the search results might not be focused on the topic of interest. Second, the query can be too short to identify what the user is exactly looking for. Third, until the user sees the results, he may not be sure what he is searching for. Or sometimes the situation may occur as even though the user knows exactly what he is looking for, he may not be able to give an appropriate query [1].

In most of the existing work on dictionary learning, more consideration was given to image data. Using the dictionary learning for sentence representation is discussed less.

III. Methodology

In this section, we will discuss different methods that we adopted for our model for the representation of documents. While searching for a particular document from a collection of documents, it is critical to gain a minimal representation that is semantically equivalent to the input. That is our primary objective. This is fulfilled through a variety of techniques.

The model mainly contains 2 kinds of Documents, one is the input Document/Query, and another is the collection or set of words, where the algorithm searches for words which match with the input documents.

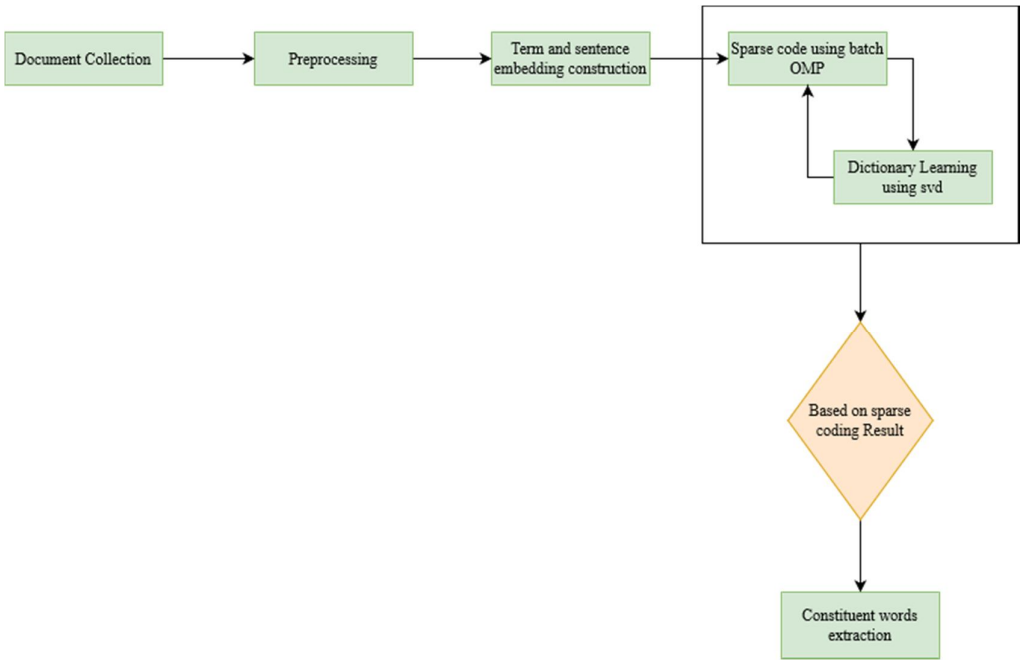

Figure 1. Architecture Diagram

A. Preprocessing of Documents

Before commencing the real processing of the documents preprocessing is performed on them for document preparation. It aids with the removal of data that is not useful, which is unnecessary data that is not needed for further processing. This will improve the quality of the document. In our project, we used Preprocessing methods such as Tagging, Lemmatization, removal of Punctuation, removal of Stop words and Numerical values, Tokenization etc. Because this is primarily a decision on which information from a collection should be extracted to meet a user's needs.

B. Dictionary Creation of Document Collection by Using Word Embedding

Dictionary-based learning is utilized for the representation of a Document set. It is a part of machine learning and is built on a synthesis framework, they seek to discover a frame; it is called a 'dictionary' in which training data can be represented sparsely. Dictionary learning is the process of learning a group of atoms to approximate an input document with a sparse linear combination of these atoms. The dictionary contains the mapping of all words, which denotes the mapping of each token into a unique id or vector after tokenization. In general, dictionaries are produced using the term vectors. TF-IDF is a common method used for creating vectors in Dictionary. But it cannot always identify the contexts of the words or comprehend their meaning. They give more importance to the frequency or weight of the term in a document. Instead of employing such methods, we used Word2vec's Word Embedding method to create a term-matrix for Dictionary in our model.
Word2vec is a popular Word embedding technique that uses a two-layer neural network to interpret a text by vectorizing words that deep neural networks can comprehend. It has the concept of learning the word embedding and connecting the target word to its context word. Word embeddings ultimately aid in developing the relationship of a word with another word with a similar meaning through the built vectors. The skip-gram approach of word2vec is employed here.

C. Generate Sentence Embedding for Input Document/Query

The input Y in our model consists of the embedding of the sentences. We have used sentence embedding with sentence transformers to generate the embedding for the sentences. Sentence Transformers is a Python framework using Siamese BERT-Networks for generating embedding for Text, Images and so on. We gain large speed benefits despite employing the same final outer layers because of the dense embeddings, which is the dense vector representation provided by transformer models is so much richer in information. It converts

sentences and paragraphs into a dense vector space with 384 dimensions, which can be utilized for applications like clustering and semantic search.

D. Sparse Representation of Documents

The sparse coding technique is used to introduce the sparse representation of input documents. To represent an input vector as a linear combination of these basis vectors, sparse coding seeks to identify a collection of dictionary basis vectors from which it can be derived. It should be in a representation such that "SE≈DX", where 'SE' denotes the embedded vectors of Input query, 'D' denotes the Dictionary of term vectors of the Document set and 'X' denotes the Sparse Representation and declared the Sparsity constraint 'S', they decide how many words to extract. The dictionary, D, and the sentence, Y, are initialized in sparse coding to calculate the representation coefficients, X. A pursuit algorithm executes this process and establishes an approximation of the solution. Before SVD, sparse coding is the necessary step. The matching pursuit (MP) and orthogonal matching pursuit (OMP) algorithms are the most fundamental of the various pursuit algorithms that exist. For each row in the sparse code corresponding to the dictionary column, a unique dot product is generated for each sparse order. The relevant dictionary columns are then extracted for getting the words based on that value.
This process continues until the sparsity constraints are met and the input document or query is represented sparsely.

E. Error Checking

Error checking is a procedure for ensuring and enhancing the accuracy of the data retrieval system. The sparse representation is acquired using OMP, and SVD is used to update the dictionary to include the error term Y DX. When the error is greater than the threshold value T, the system invokes the decomposition algorithm SVD (Singular Value Decomposition) to improve the fit of the data by updating the atoms in the dictionary. The steps are repeated until the error is less than the threshold value.

F. Finding the words by using the sparse representation with fewer errors

Based on the sparsity constraints and error checking, we retrieved the words with fewer errors. Our objective is to extract relevant words related to the input statement as well as statements that are related to it. So, the extracted words can be implemented for the ranking techniques in the Information retrieval system, which is the process of arranging the best matches to the input statement, resulting in a deeper level of result.

IV. Result and Analysis

The data collection we utilised in our experiments consists of 30 documents each consisting of one or two sentences. These 30 documents were used to construct the dictionary among which 20 are based on "global warming" which is collected from different sites available on the Internet and 10 are from different domains. For testing purposes, we used 4 sentences which are about global warming. The dictionary consists of the term vectors created by using Word2Vec. There were 173 words in the vocabulary and the dimension of the vector was 384. So, the dictionary is of dimension 173 x 384. The sentence embedding was given as the input which is Y. There were 4 sentences, and each sentence embedding was of size 384. So, the dimension of Y was 4 x 384. In the phase of sparse coding, the resulting sentence embedding values are provided. The approximation error (Y-DX) was found when sparse code X was acquired. When the dictionary learning phase begins, this is delivered, and new dictionary atoms are created. Iterative updates are made to the dictionary columns.
The output is evaluated using a range of precision and accuracy. By dividing the entire number of recovered words, including False Positives, by the number of applicable words (True Positive: TP), Precision(P) was calculated (FP).

$$P = TP/(TP + FP) \qquad (4)$$

The absolute number of right classifications divided by the total number of classifications is how accuracy, also known as the truth of results, is calculated.

$$A = TP + TN/(TP + FN + TN + FP) \qquad (5)$$

The number of words that we need to obtain in the output can be determined by setting the sparse count. When the sparse count is 6, we will get 6 words as output. When it is 13, we will get 13 words as output.
The 173 words in the dictionary are obtained from the 30 documents that we used to create the term vectors. Among these 173 words, 109 words are related to global warming and the remaining 64 words are related to other domains. In our experiment, we have given different sparsity counts as 13, 80 and 100. So when the

sparsity count is 6, out of the 6 words retrieved, how many relevant words there are compared to how many irrelevant words there are reveals how many cases are true positives (TP) and how many are false positives (FP). The number of words that are applicable and inapplicable out of the 167 words that were not recovered also reflects the number of false negative (FN) cases and the number of true negative (TN) cases, respectively.
The experiment was conducted with different input sentences. The output that we obtained for 4 test sentences is as follows:

TABLE I. CONFUSION MATRIX WHEN THE SPARSE COUNT = 13

	Relevant Words	Non-Relevant Words
Retrieved Words	True Positive = 11	False Positive = 2
Non-retrieved words	False Negative = 98	True Negative = 62

TABLE II. CONFUSION MATRIX WHEN THE SPARSE COUNT = 80

	Relevant Words	Non-Relevant Words
Retrieved Words	True Positive = 71	False Positive = 9
Non-retrieved words	False Negative = 38	True Negative = 55

TABLE III. CONFUSION MATRIX WHEN THE SPARSE COUNT = 100

	Relevant Words	Non-Relevant Words
Retrieved Words	True Positive = 89	False Positive = 11
Non-retrieved words	False Negative = 20	True Negative = 53

TABLE IV. EVALUATION RESULT

Sparse Count	Precision	Accuracy
13	85%	42%
80	89%	73%
100	89%	82%

TABLE V. RESULT OF ONE INPUT SENTENCE

Sentence	Key Words (Sparse count =13)
Global warming is an aspect of climate change that refers to the long-term rise of the planet's temperature because of the increased concentration of greenhouse gas in the atmosphere mainly from human activity such as burning fossil fuels.	fossil, film, encourage, worry, earth, significant, global warming, often, current, weather, collect, preproduction, serious

V. CONCLUSIONS

Document representation is always a hot topic of discussion. As the number of documents increases day by day, an effective way to represent these documents is inevitable. Also, while representing these documents the meaning of the documents must not be lost. In our model, we have proposed an effective way of using a dictionary of term vectors for obtaining the minimal representation of documents. There are many applications in which document representation is essential. We can use the retrieved words in applications like document retrieval. Our proposed model for document representation has been verified, opening the path for successfully improving the retrieval procedure. We have obtained 89% precision in our experiment.

REFERENCES

[1] Hiteshwar Kumar Azad and Akshay Deepak. "Query expansion techniques for information retrieval: a survey". In: Information Processing & Management 56.5 (2019), pp. 1698–1735

[2] Yoshua Bengio, R´ejean Ducharme, and Pascal Vincent. "A neural probabilistic language model". In: Advances in neural information processing systems 13 (2000).

[3] Snehal Bhoir, Tushar Ghorpade, and Vanita Mane. "Comparative analysis of different word embedding models". In: 2017 International conference on advances in computing, communication and Control (ICAC3). IEEE. 2017, pp. 1–4.

[4] Jey Han Lau and Timothy Baldwin. "An empirical evaluation of doc2vec with practical insights into document embedding generation". In: arXiv preprint arXiv:1607.05368 (2016).

[5] Remya RK Menon, Astha Ashok, and S Arya. "Document Cluster Analysis Based on Parameter Tuning of Spectral Graphs". In: Innovative Data Communication Technologies and Application. Springer, 2022, pp. 401–413.

[6] Remya RK Menon, R. Akhil dev and S. G. Bhattathiri, "An Insight into the Relevance of Word Ordering for Text Data Analysis," 2020 Fourth International Conference on Computing Methodologies and Communication (ICCMC), 2020, pp. 207-213, doi: 10.1109/ICCMC48092.2020.ICCMC-00040.

[7] Remya RK Menon, J. Kaartik, E. T. Karthik Nambiar, A. K. T.K. and A. K. S., "Improving Ranking in Document based Search Systems," 2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184), 2020, pp. 914-921, doi: 10.1109/ICOEI48184.2020.9143047.
[8] VT Priyanga et al. “Exploring fake news identification using word and sentence embeddings”. In: Journal of Intelligent & Fuzzy Systems 41.5 (2021), pp. 5441-5448.
[9] Niveditha Sekar, S Chandrakala, and G Prakash. “Analysis of Global Word Representations for Depression Detection.” In: ISIC. 2021, pp. 136–148.
[10] Vanika Singhal and Angshul Majumdar. “Noisy deep dictionary learning”. In: Proceedings of the Fourth ACM IKDD Conferences on Data Sciences. 2017, pp. 1–10.
[11] V Sreevidhya and Jayasree Narayanan. “Short descriptive answer evaluation using word-embedding techniques”. In: 2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT). IEEE. 2021, pp. 1–4.
[12] Tang, H., Liu, H., Xiao, W., & Sebe, N. (2020). When dictionary learning meets deep learning: Deep dictionary learning and coding network for image recognition with limited data. IEEE transactions on neural networks and learning systems, 32(5), 2129-2141.
[13] Tariyal, S., Majumdar, A., Singh, R., & Vatsa, M. (2016). Deep dictionary learning. IEEE Access, 4, 10096-10109.
[14] Zhang, J., Chen, Y., Cheung, B., & Olshausen, B. A. (2019). Word embedding visualization via dictionary learning. arXiv preprint arXiv:1910.03833.