Tabriz Journal of Electrical Engineering (TJEE), vol. 52, no. 4, Winter 2022 Serial no. 102
DOI: 10.22034/tjee.2022.15744

Topic Detection on COVID-19 Tweets: A Comparative Study on Clustering and Transfer Learning Models

Elnaz Zafarani-Moattar[1], Mohammad Reza Kangavari[*, 2], Amir Masoud Rahmani[1]

[1] Department of Computer Engineering, Science and Research Branch, Islamic Azad University, Tehran, Iran
[2] Department of Computer Engineering, Iran University of Science and Technology, Tehran, Iran

[*] Corresponding Author: kangavari@iust.ac.ir

Received: 08/09/2022, Revised: 11/16/2022, Accepted: 12/17/2022

Abstract
Automatic topic detection seems unavoidable in social media analysis due to big text data which their users generate. Clustering-based methods are one of the most important and up-to-date categories in topic detection. The goal of this research is to have a wide study on this category. Therefore, this paper aims to study the main components of clustering-based-topic-detection, which are embedding methods, distance metrics, and clustering algorithms. Transfer learning and consequently pretrained language models and word embeddings have been considered in recent years. Regarding the importance of embedding methods, the efficiency of five new embedding methods, from earlier to recent ones, are compared in this paper. To conduct our study, two commonly used distance metrics, in addition to five important clustering algorithms in the field of topic detection, are implemented by the authors. As COVID-19 has turned into a hot trending topic on social networks in recent years, a dataset including one-month tweets collected with COVID-19-related hashtags is used for this study. More than 7500 experiments are performed to determine tunable parameters. Then all combinations of embedding methods, distance metrics and clustering algorithms (50 combinations) are evaluated using Silhouette metric. Results show that T5 strongly outperforms other embedding methods, cosine distance is weakly better than other distance metrics, and DBSCAN is superior to other clustering algorithms.

Keywords:
Topic Detection, Transfer Learning, Embedding Methods, Distance Metrics, Clustering Methods, COVID-19.

1. Introduction

In mid-December 2019, a Corona-type virus called COVID-19 was reported in Wuhan, China, which quickly became the most important news headline worldwide and had wide reactions on social media. When this disease has been reported for the first time, no one could imagine that it would become a global pandemic and be continued in this way, and it would lead to severe damage to the economies of nations by widespread closures. The number of people infected by the coronavirus has reached more than 640 million in the world in November 2022, and the number of its victims has exceeded 6.6 million [1]. The United States is at the first rank having more than 99 million COVID-19 infections [2], followed by India, France and Germany.
The high importance of the COVID-19 pandemic and its worldwide prevalence has led to the daily production of large volumes of messages on the topic of COVID-19 on social media. The extent of the COVID-19 topic has also led to the production of these messages in many subtopics; for example, symptoms, the way to prevent, diagnose and treat, vaccine production process, as well as its impact on the economy, individual life, and people's mental health (e.g. mental health of people has been studied in [3], [4]). Each of these subtopics is a topic with a high volume of published messages. Therefore, COVID-19 dataset is suitable for topic detection as it contains many topics. Above mentioned reasons (high importance of COVID-19 and its suitability for topic detection) in addition to need to study different aspects of topic detection, conduct us to do this research.
As production of digital data increases everywhere, it is becoming difficult to categorize and retrieve such huge amount of data. Therefore, topic detection is considered as a way to mine such big data. Topic detection is the automatic process to extract and link similar documents about an event or subject from input text [5]. Topic detection methods on text are classified into different categories from different aspects [6]–[10]. Based on [8], topic detection methods are classified into five categories: clustering, frequent pattern mining, exemplar-based, matrix factorization, and probabilistic models. Exemplar-based technics are those in which a post (i.e. exemplar) is provided to user instead of keywords. Therefore, topics should be extracted somehow before finding best exemplar. Probabilistic models need sufficient frequency of words to be reliable but most of the time, length of posts are not big enough in social media. Matrix factorization methods are time consuming. Therefore, frequent pattern mining methods used to be leader in this field. But, as good embedding

DOI: 10.22034/tjee.2022.15744

methods are developing, researches are moving to clustering methods. Therefore, clustering based methods are considered in this paper.

Clustering requires distance and calculating distance needs embedding. The aim of this research is to simultaneously study these three factors (embedding methods, distance metrics and clustering methods) and their interaction.

This paper is a comparison study in the context of clustering based topic detection on COVID-19 data. The tweets posted on the Twitter social network on the topic of COVID-19 is used as a dataset. Twitter is chosen because it is very popular and used by most of papers in this field. Words of dataset can't directly fed into clustering algorithms. They should be converted to vectors. Embedding methods are used for this reason. In this paper, five word embedding methods are selected and investigated, namely, Word2Vec, fastText, GloVe, BERT and T5. Word2Vec is one of the earliest embedding methods, and T5 is one of the latest and most up-to-date embedding methods. Distance of vectors should be measured before feeding them into clustering algorithms. Euclidian distance and Cosine distance are the most important and popular distance metrics which are used by most papers in this field. Both metrics are studied in this research. Finally, clustering algorithms should be used to cluster and link similar documents together. Five clustering algorithms are investigated in this paper which are: k-means, DBSCAN, OPTICS, spectral and Jarvis-Patrick. These algorithms are selected from different categories to have a wide study on clustering algorithms.

According to the above description, three major research questions are considered for this paper：

1) Which of the embedding methods has better performance in topic detection on COVID-19 tweets?

2) Which of the distance metrics has better performance in topic detection on COVID-19 tweets?

3) Which of the clustering methods has better performance in topic detection on COVID-19 tweets?

The remaining of the paper is organized as flowing: In section 2 some related works are reviewed and works on COVID-19 are organized. Basics of methods which are used in this study are presented in the section 3. Section 4 provides a framework in which the study is defined and done. Details of parameters used for implementation in addition to details of dataset, evaluation metrics and results of study are presented in section 5. Discussion and conclusion are provided in sections 6 and 7 respectively.

2. Literature Review

There are many articles in the field of COVID-19 disease, including three fields of image, dataset and text. This work is done in the field of text. Most of the previous works on COVID-19 are in the field of image, which have been performed on X-ray images of the chest[11]–[14]. In fact, they attempt to detect healthy people from COVID-19 patients by the lung image using deep learning methods. In the field of data, various datasets have been compiled, such as CORD-19 and CORD19STS datasets. There are several works in the field of text, such as sentiment analysis and topic detection (topic modeling). In the following, each field will be studied in detail.

Since all NLP based works require a dataset, providing a dataset is considered in many papers from different aspects. In CORD-19 [15], 128,000 articles with keywords such as COVID, Coronavirus and 2019-nCoV have been collected from publishers such as Elsevier, Springer, etc. Then, clustering and duplicates removing have been applied to the articles. In CORD19STS [16], a parameter called STS[1] is measured on CORD-19 articles and manually annotated by labels such as Related, Somewhat-related and Not-related by AMT[2] users. In [17], a corpus including 7500 tweets about the corona test has been provided and manually labeled in five classes of events with topics of "Tested Positive", "Tested Negative", "Can Not Test", "Death" and "Cure & Prevention". For each of these events, a number of questions are addressed. These questions are: who, when, and where has a positive or negative corona test? In [18], a multi-language dataset including 6 million tweets has been collected. Considering the distribution of collected tweets shows that English has the highest rate among the 66 languages in the dataset, and 63% of the tweets are in English. Also, in [19], a dataset containing 123 million tweets has been published, in which 60% of tweets are in English. In [20], a dataset called CovidQA has been presented, which contains 124 question-article pairs. In [21], a dataset called TweetsCOV19 has been introduced, which contains more than 11 million tweets. Metadata about tweets such as entities, hashtags, user mentions, sentiments and URLs are also extracted. There is also a dataset on Twitter [22] in the sentiment analysis area, which classifies tweets into five classes as follows: very positive, positive, neutral, negative and very negative. Also, 10 topics are specified, and the relation of each tweet with these 10 topics are determined. However, this dataset includes tweets that do not belong to any of these topics or belong to several topics simultaneously. In [23], the COV19Tweets dataset is introduced, which has more than 310 million tweets in English along with a sentiment score. The Geo version is presented as GeoCOV19Tweets dataset, which includes tweets originated from 204 different countries, and the United States has the largest rate (43%).

Most of the researches in the field of text are about topic detection/modelling and sentiment analysis. Some examples will be discussed in the following paragraphs. One of the works in the field of sentiment analysis is presented in [24]. Four classification algorithms are considered in this work: Linear Regression Model, Naïve Bayes Classifier, Logistic Regression and K-Nearest Neighbour. Here, the length of the tweet is also taken into account to evaluate the performance of each method in short tweets and longer tweets. Textual data visualization is also used to identify the critical trend of change in fear-sentiment. The results are presented as word clouds and top n-grams (1-4

[1] Semantic Textual Similarity

[2] Amazon Mechanical Turk

DOI: 10.22034/tjee.2022.15744

grams). In [25], topic detection and sentiment analysis are studied on the Reddit social network. The research framework consists of four steps. In step 1, COVID-19-related comments are collected from the Reddit social network. Step 2 includes preprocessing of data. In step 3, the LDA method is used for topic detection. In step 4, the combination of embedding and LSTM is used for sentiment classification on COVID-19 comment. The GloVe 50-dimensional is used for embedding. Top-10 topics and word clouds are used to show results. In [26], eleven salient topics with the highest score in the LDA method are selected. In addition to these topics, bigram[3] is also specified. Eight emotions, such as Anger, Fear, Joy, Sadness, etc., are considered. The experimental results show that the feeling of fear is prominent. Results are show by bigrams and distance map. In [27] and [28], LDA method is used for topic modeling, and VADER[4] is applied for sentiment analysis. VADER is a lexicon and rule-based sentiment analysis tool that is specifically attuned to the sentiment expressed in social media. Top-10 topics and word clouds are used to show results. In [29], a neural network for sentiment analysis using multilingual sentence embedding is presented. This network is trained on the Sentiment140 dataset. Also, it is tested on pre-trained word- and sentence-level embedding models: Word2Vec, BERT, ELMO and MUSE[5]. MUSE outperform other methods and its MSE is 0.11. In addition, tree map of Twitter activity in Europe during the time period from December 2019 to April 2020 is plotted. The United Kingdom, Spain, Germany, Italy, and France have the highest contribution of tweets.

A sample of text classification works is presented in [30], which uses KNN, LR[6] and SVM for classification. In classification, topics must be predefined as labels in advance. In this paper, 11 labels are defined as follows: “Donate”, “News & Press”, “Prevention”, “Reporting”, “Share”, “Speculation”, “Symptoms”, “Transmission”, “Travel”, “Treatment”, “What Is Corona?”. Experiments of this paper shows that SVM reached to the accuracy of 86.92% which outperformed other methods. Another sample of text classification is [31], which is done on Sina Weibo social network. Here, Weibo data are classified into seven types of situations, using supervised learning methods, namely SVM, Naïve Bayes and Random Forest. The average accuracy of the SVM, NB, and RF classifiers are 0.54, 0.45, and 0.65 respectively.

3. Materials and Methods

As mentioned in the introduction section, a comparative approach is selected to compare between three components: embedding methods, distance metrics and clustering methods. The proposed framework will be presented in section 4, and it will be mentioned that the main processing phase of the framework includes these three components. Therefore, these three components are described in this section.

3.1. Embedding Methods

The main concept of word embedding is that any words in a language can be expressed by a vector of numbers. Usually, the existing word embedding models have vectors with the length between 200 and 750. Word embedding consists of n-dimensional vectors which try to record the meaning of words and their content by numerical values. Each set of numbers can be considered as a "word vector", but it is not necessarily useful. A useful word embedding vector represents the meaning of that word, and the similar words reach similar embedding values. In other words, semantic similarity leads in vector similarity. Therefore, using these vectors for clustering results in semantic clustering.

In this study, five methods, from earlier to new methods, are selected among the embedding methods. Based on [32], word embedding methods are classified as representative methods and recent developed methods. Representative methods are classified into global matrix factorization methods and sliding window based methods. Word2Vec, which is baseline in word embedding, is chosen from the sliding window category. Global matrix factorization methods (such as LSA) are well on carrying statistical information of a corpus text but are weak in carrying word analogy. Meanwhile, those are not scalable to the large-scale corpus [10]. Therefore, none of the methods in this category is selected. Instead, GloVe is selected which is a hybrid of global matrix factorization methods and sliding window based methods. Recent developed methods, such as ELMo, GPT, Bert and T5, try to tackle the issue of word polysemy by presenting a contextualized representation of words. Bert is better than ELMo and GPT. Therefore, Bert and T5 are selected for this study. We will briefly review each method in the following.

3.1.1 Word2vec (2013)

Word2Vec has been developed in 2013 by a Mikolov et al. at Google [33]. Word2Vec is a baseline in the field of word embedding [34]. Although the one-hot encoding existed before Word2Vec, Word2Vec is the first embedding method in practice; because in one-hot encoding, we cannot make any significant comparison between two vectors and can only check whether the vectors are equal. In other words, semantic similarity does not lead to vector similarity in one-hot encoding. However, in Word2Vec, the vectors are calculated in a way that the semantic similarity leads to vector similarity. In Word2Vec, a distributed representation is used for each word. It means that each word is represented by a distribution of numerical values (weights) on different elements of the vector. Therefore, the representation of a word is distributed across all the elements of a vector instead of one-to-one mapping between an element in a vector and a word in a dictionary and each element contribute to the meaning of a large number of words. As a result, different words are described by different values of each element, and all elements contribute to recording the meaning of words.

[3] The most popular pairs of words within each topic

[4] Valence Aware Dictionary and sEntiment Reasoner

[5] Multilingual Universal Sentence Encoder

[6] logistic regression

DOI: 10.22034/tjee.2022.15744

3.1.2 fastText (2016)

fastText is a library created in 2016 by Facebook's AI Research (FAIR) lab [35], which is applied for learning word embedding and text classification [36]. The fastText uses the vector representation combination of word sections to represent a word vector. The models which assign a distinct vector to each word do not consider the word's morphology. This representation is considered as a constraint for languages that have a large number of words where there are so many rare words. The applied method in fastText is based on skip-gram model, where each word is represented as a bag of character n-grams, and a vector is assigned to each character n-grams. The sum of these vectors forms the word representation. In other words, the representation of each word is obtained from the sum of the character n-gram vectors of that word. This method has a high speed and is trained quickly on large corpora. This model is also able to create a word representation vector for words not appeared in the training dataset.

3.1.3 GloVe (2014)

GloVe is the abbreviation for Global Vector. This model was developed in 2014 at Stanford University as an open-source project [37]. This model is an unsupervised learning algorithm to obtain vector representation for words. This is achieved by mapping the words within a meaningful space where the distance of the words' vector is related to their semantic similarity. The log-bilinear regression model is used for unsupervised learning of word representation. This model combines the advantages and features of global matrix factorization and local context window models. This model acts based on statistical information and trains only non-zero elements in the word-by-word co-occurrence matrix instead of the entire sparse matrix. Although the methods such as LSA, as a member of the global matrix factorization family, are efficient in statistical information, they have weak performance in word analogy. In contrast, methods such as skip-gram, from the local context window family, perform better in terms of word analogy, but perform poorly in terms of corpus statistics, as they use the local window in training instead of global co-occurrence.

3.1.4 Bert Embedding (2018)

In 2018, a big model called BERT [38] was trained by Google engineers with lots of data (Wikipedia + Book Corpus) and made available for NLP researches. In 2017, Vaswani et al. at Google had published a paper entitled "Attention Is All You Need" [39] and introduced the concept of transformer neural network, which has been used in BERT. Two methods can be used to train BERT: Masked Language Model and Next Sentence Prediction. The given model can be used in two ways: feature extraction and fine-tuning. The BERT model is trained in two different sizes: The base BERT consists of 12 Encoder layers (called Transformer Blocks in the original article), and the larger network consists of 24 Encoder layers.

3.1.5 T5 Embedding (2020)

In 2020, Colin Raffel et al. has introduced their proposed framework, called T5: "Text-to-Text Transfer Transformer" [40]. The idea behind T5 is to consider each text processing problem as a Text to Text problem, i.e. taking the text as an input and generating the new text as an output. In the T5 architecture, the original Transformer architecture of Vaswani et al. [41] is used, but there are three differences: First, the Norm bias layer is removed. Secondly, the layer normalization is out of the residual path. Third, a different position embedding scheme is applied.

T5 is commonly used as pretrained. The pretrained T5 model is publicly available. Unlike machine vision, where the network is supervisely-pretrained on labeled data, in NLP, unsupervised learning is often used on unlabeled data. Therefore, a large-scale dataset, called C4, is used to train T5. C4, which stands for "Colossal Clean Crawled Corpus", has been gathered and introduced for T5 training for the first time. This dataset has been built as a source for unlabeled text, which includes hundreds of gigabytes of clean English text.

3.2. Similarity Metrics

Metrics are needed to measure the degree of similarity or distance among texts in text clustering. As we said, the texts will be converted to embedding vectors. Since Euclidean distance and cosine similarity are the most commonly used metrics in the vector space, they are used in this research. Suppose A = [a_1, a_2, …, a_n] and B = [b_1, b_2, …, b_n] are two vectors (points) in the n-dimensional Euclidean space. Their Euclidean distance is defined as:

$$d(A,B) = \sqrt{\sum_{i=1}^{n}(a_i - b_i)^2} \quad (1)$$

Since this distance is not normal, the normalized Euclidean distance is used, which is obtained as:

$$Dis_{Euc}(A,B) = \frac{d(A,B)}{\max_{i,j}\{d(A_i,B_j)\}} \quad (2)$$

If it is needed, the Euclidean distance is converted to Euclidean similarity using the following formula:

$$Sim_{\mathrm{Euc}}(A,B) = 1 - Dis_{\mathrm{Euc}}(A,B) \quad (3)$$

The cosine similarity between two vectors A and B is also calculated as:

$$Sim_{\mathrm{Cos}}(A,B) = \frac{A.B}{\|A\|\|B\|} = \frac{\sum_{i=1}^{n} a_i b_i}{\sqrt{\sum_{i=1}^{n} {a_i}^2}\sqrt{\sum_{i=1}^{n} {b_i}^2}} \quad (4)$$

The cosine similarity is converted to cosine distance using the following formula whenever it is needed:

$$Dis_{\mathrm{Cos}}(A,B) = 1 - Sim_{\mathrm{Cos}}(A,B) \quad (5)$$

3.3. Clustering Methods

The works on Topic Detection follow two different approaches: classification and clustering. In the classification approach, the number and type of classes are predetermined. This approach is suitable for applications in which classes are well known. For example, news agencies classify news into predefined classes such as economics, health, sports, and so on. However, in the field of social media, a new topic may occur at any moment, and the previous topics will fade over time. Therefore, classification algorithms will not be applicable for this task and clustering algorithms must be used.

In this paper, the common clustering algorithms used in topic detection are compared. k-means, DBSCAN [42], OPTICS [43], spectral [44] and Jarvis-Patrick [45] clustering algorithms are selected for this purpose. Among these methods, k-means is a baseline in clustering algorithms which belongs to partitioning

DOI: 10.22034/tjee.2022.15744

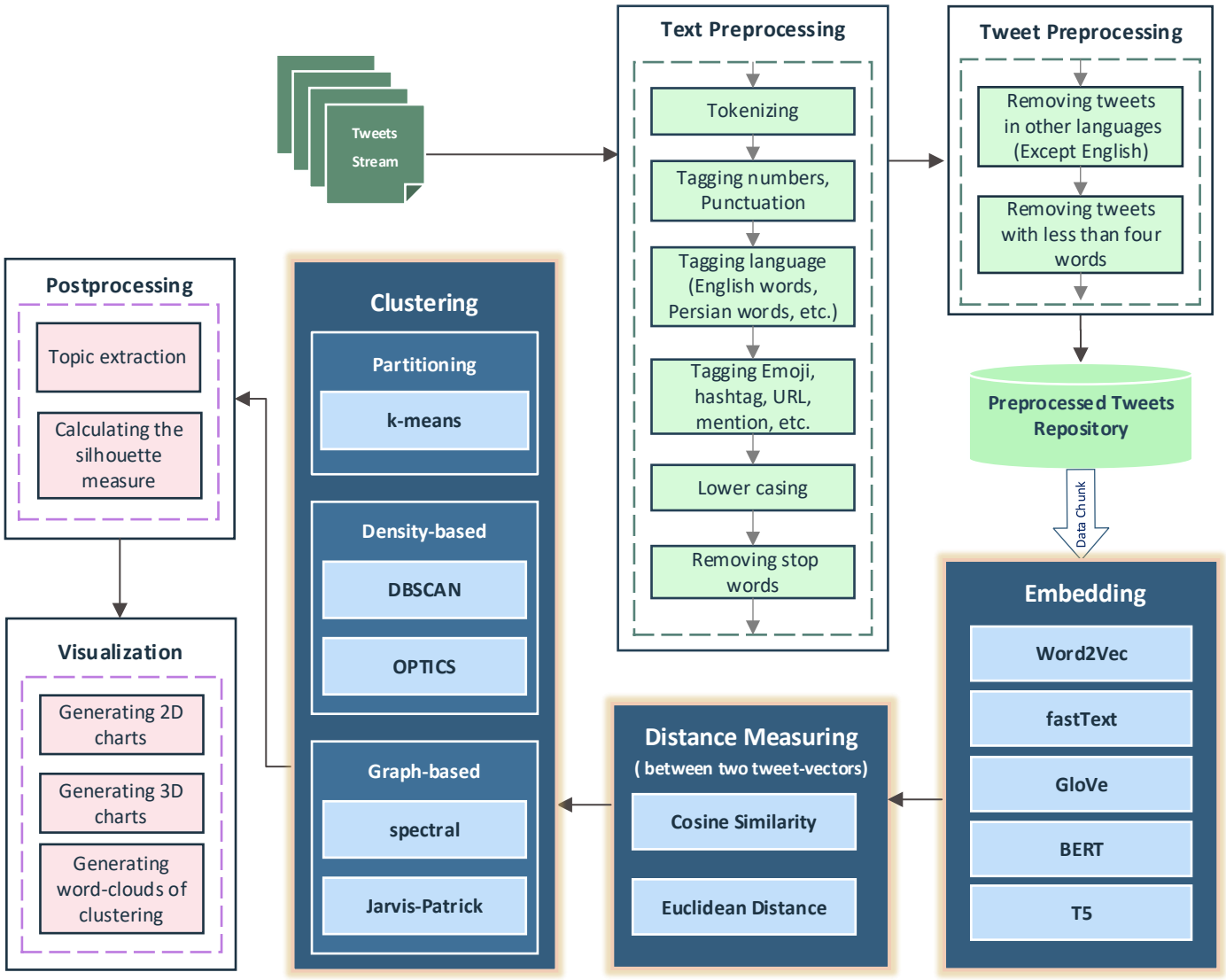

Figure 1: Overview of the research framework

category. DBSCAN and OPTICS, which belong to density-based category, are chosen because of their ability to find clusters of varying density and shape [34]. Spectral clustering can be used to uncover hidden manifold structures [46] and Jarvis-Patrick is less investigated in the field of topic detection. Therefore, these two algorithms are selected from graph-based category.

4. System Framework

The flowchart of our work is shown in Figure 1. As shown in the figure, the proposed framework consists of three phases: preprocessing, main processing, and postprocessing. Tweet stream fed into the preprocessing phase, which includes two components: First, text preprocessing is applied to each tweet's text. Then the tweets are filtered, which means a decision is made over keeping or removing the tweet. The results of this phase are stored temporarily. Then the messages are entered into the main processing phase as a chunk of tweets, which includes three components: embedding, distance measuring and clustering. Finally, in the third phase, postprocessing and visualization are applied to the results.

4.1. Preprocessing

The preprocessing phase has two components: text preprocessing and tweet preprocessing. Text preprocessing consists of the following steps: the input text is tokenized, a tag is added for tokens which are numbers or punctuation; also, the language tag (English word, Persian word, etc.), tag for emoji, hashtag, URL, mention and other special characters are added; then the remaining text is converted to lowercase, and stop words are removed. A special tokenizer developed in ComInSys lab is used for this purpose which is able to perform all these steps except lower case conversion and stop word removal. Tweet preprocessing is the next component in which tweets in other languages and tweets with less than four remaining words are removed.

4.2. Main Processing

The main processing phase consists of three components: embedding, distance measuring and clustering. First of all, the tweets are embedded by one

DOI: 10.22034/tjee.2022.15744

Table 1: Dimension of embedded vectors for each embedding method.

Embedding method	Word2Vec	fastText	GloVe	BERT	T5
Dimension of embedded vector	400	400	200	768	768

Table 2: The applied clustering methods, their parameters, range of their values and the number of experiments which is performed during parameter tuning.

Clustering Method	k-means	DBSCAN		OPTICS	spectral	Jarvis-Patrick	
Parameter	k	ε	Min Pts	Min Pts	K	k	k_t
Value Range	2-49	0.1-5	2-49	2-49	2-49	10-100	1-k
Number of	48	2400		48	48	5005	

Figure 2: A sample of word cloud produced by our system

of the given methods (i.e. each tweet is embedded by each embedding method). Then the distance between every two embedding vectors is measured by distance metrics (i.e. the distance between each vector and any other vector is measured by both Euclidean distance and cosine distance). Next, the embedding vectors of tweets, along with the distance of each pair, are given to each clustering method in order to extract clusters of tweets. In other words, each tweet is embedded by each embedding method, then the distance of each pair is measured by each distance metric, and finally, all tweets are clustered by each clustering method. As a result, the combination of these components makes 5×2×5=50 cases. Therefore, 50 tests have been performed for these 50 cases, and the results are given in the next section.

Among the embedding methods, Word2Vec, fastText, GloVe, BERT and T5 are used. The embedding vector dimension for each method is given in Table 1. Word2Vec is fine-tuned over COVID data, fastText and GloVe are fine-tuned over tweeter data, and basic versions of BERT and T5 are used. Therefore, transfer learning is used in this paper. Transfer learning in deep learning and specially NLP, enabled researchers to use pretrained models for various problems [47], [48], [49].

4.3. Postprocessing

The postprocessing phase has two components: postprocessing of clusters and visualization of the results. The postprocessing of clusters consists of topic extraction and calculating silhouette measure. The Silhouette measure lets us compare different combinations of the components discussed in the previous section. The main advantage of silhouette measure for our work is that it can be used without ground truth knowledge. Word cloud is one of the visualization results of the system. A sample of a word cloud produced by the system is shown in Figure 2.

5. Implementation Details

In this section, the details of implementation are considered. First, the dataset and then the evaluation metric are presented. Finally, the details of each experiment and its results are discussed.

5.1. Data Set

This research's dataset includes tweets from March 29, 2020, to August 30, 2020, taken from [50], [51]. This dataset contains tweets from users who have used the following hashtags: #coronavirus, #coronavirusoutbreak, #coronavirusPandemic, #covid19, #covid_19, #epitwitter, #ihavecorona, #StayHomeStaySafe, #TestTraceIsolate. The collected dataset has multiple fields such as: text of tweet, tweeted account, used hashtags, account location, tweet language, and retweet argument. As the processing has been done in English, English-related tweets are taken from the dataset (i.e. tweets whose "Lang" field is "En"). In addition, the retweets argument is set to FALSE. Therefore, the dataset does not contain retweets.

5.2. Evaluation Metric

Silhouette measure is used to evaluate performance and compare alternative methods in different experiments. The main advantage of silhouette measure is that it can be used without any ground truth knowledge.

For each data point i in the cluster C_m, silhouette value $s(i)$ is calculated as:

$$s(i) = \begin{cases} 0 & if\ |C_m| = 1 \\ \frac{b(i)-a(i)}{max\{a(i),b(i)\}} & if |C_m| > 1 \end{cases} \quad (6)$$

in which $a(i)$ and $b(i)$ are calculated as:

$$a(i) = \frac{1}{|C_m|-1}\sum_{j\in C_m,\ j\neq i} d(i,j) \quad (7)$$

$$b(i) = \min_{n\neq m} \frac{1}{|C_n|}\sum_{j\in C_n} d(i,j) \quad (8)$$

where $d(i, j)$ is the distance between data points i and j.

5.3. Experiments and Results

In this section, the experiments to answer the research questions are considered. Since the applied clustering methods have parameters to be set, it is necessary to perform experiments to set these parameters at the first stage. Table 2 lists the parameters of different methods along with the range of values assigned in the experiments. As it can be seen, some methods have a single parameter, and others have two parameters. For methods with two parameters, both two parameters have been tuned simultaneously. Figure 3 and Figure 4 show the results of some experiments as a sample. These figures can give us an overview of behaviour of these parameters and their best values. The optimal parameter value for single-parameter methods (k-means, OPTICS and spectral) and two-parameter methods (DBSCAN and Jarvis-Patrick) are selected by eq. (9) and eq. (10),

DOI: 10.22034/tjee.2022.15744

respectively.

$$p^* = \underset{p}{Argmax}\{Exp(i,j,k,p)\} \quad \forall p \epsilon Domain(p), i \epsilon Emb, j \epsilon Dis, k \epsilon Cls \tag{9}$$

$$\langle p_1{}^*, p_2{}^* \rangle = \underset{\langle p_1, p_2 \rangle}{Argmax}\{Exp(i,j,k,p_1,p_2)\} \quad \forall p_1 \epsilon Domain(p_1) \wedge \forall p_2 \epsilon Domain(p_2), i \epsilon Emb, j \epsilon Dis, k \epsilon Cls \tag{10}$$

where $Exp(i,j,k,p)$ represents the performed experiment by embedder i, distance metric j and clustering method k using p-value for the given parameter (similarly, in $Exp(i,j,k,p_1,p_2)$, using p_1 and p_2 values). Then, the experiments with the optimal value of the parameters have been used for the next steps, i.e. the results are obtained by eq. (11) and (12):

$$Res(i,j,k) = Exp(i,j,k,p^*) \tag{11}$$

$$Res(i,j,k) = Exp(i,j,k,p_1^*,p_2^*) \tag{12}$$

After tuning the parameters, the main experiments are performed to answer the main research questions. The experimental results are presented in Table 3. The first question of this research is "Which embedding method is more effective in topic detection?". It is necessary to neutralize the effect of other variables in order to be able to compare different embedding methods. For this purpose, the final result of each embedding method is obtained by eq. (13).

$$R_E(i) = (|Dis|.|Cls|)^{-1} \sum_{j \epsilon Dis} \sum_{k \epsilon Cls} Res(i,j,k) \tag{13}$$

where $R_E(i)$ represents the final result of the i^{th} embedding method. Figure 5 shows the obtained results. It can be seen that the T5 embedder has achieved the best result while the fastText fails to obtain good results. Word2Vec, BERT and GloVe have gained medium results. Table 3 shows that T5 has more stable results, as in most cases (6 of 10) it has the first rank. To verify it, the ranks of embedding methods are listed in Table 4.

The second research question is: "which distance metric has better performance?". Same as the previous question, the final result of each distance metric is obtained by eq. (14):

$$R_D(j) = (|Emb|.|Cls|)^{-1} \sum_{i \epsilon Emb} \sum_{k \epsilon Cls} Res(i,j,k) \tag{14}$$

where $R_D(j)$ represents the final result of the j^{th} distance metric. The results are shown in Figure 6. As it can be seen, the cosine distance is superior to the Euclidean distance. However, since its superiority is weak, we decide to perform more investigations. Therefore, we study the superiority of each one in the results of different embedding and clustering methods. The results are given in Table 5. According to the results, it is obvious that the superiority of cosine distance over Euclidean distance is weak (15 out of 25 cases). It is worth noting that when using k-means, cosine distance is a better metric than Euclidean distance. Also, this is almost true for Jarvis-Patrick.

Which clustering method outperforms others?" This is the third research question. Same as other questions, it

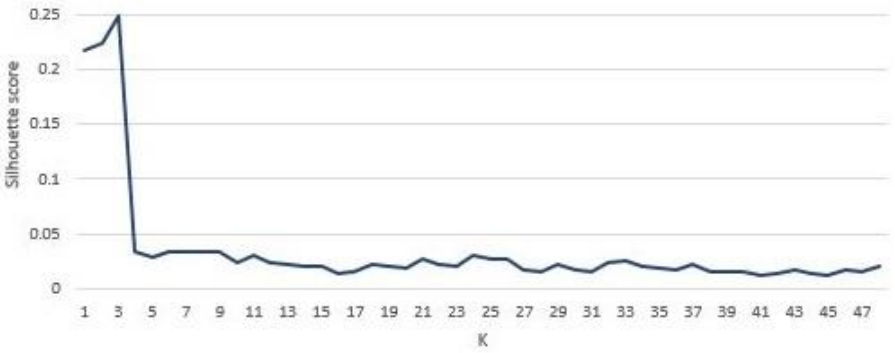

Figure 3: Spectral clustering performance measured by silhouette score for different values of k (measured on Word2Vec embedder and Euclidian distance)

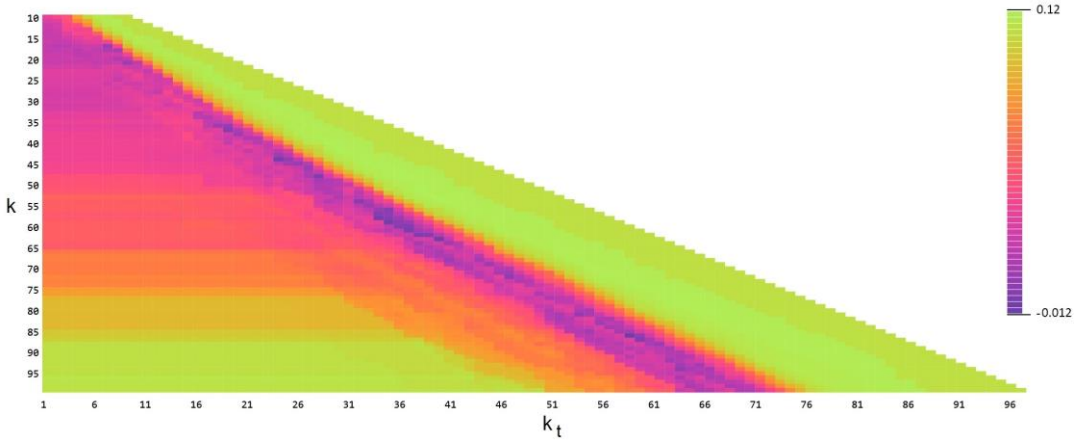

Figure 4: Heat map visualization of Jarvis-Patrick clustering performance on silhouette score. Each row corresponds to values of k, and each column corresponds to values of k_t. fastText embedder and cosine distance

DOI: 10.22034/tjee.2022.15744

Table 3: Experimental results for different embedding and clustering methods and different distance metrics measured by silhouette score (Res(i, j, k))

Clustering Method	Distance Metric	Embedding Method	Silhouette Coefficient
k-means	Euclidian	Word2Vec	0.03594
		GloVe	0.06927
		fastText	0.03286
		BERT	0.03931
		T5	0.18788
DBSCAN	Euclidian	Word2Vec	0.32647
		GloVe	0.33579
		fastText	0.38503
		BERT	-0.00844
		T5	0.42998
OPTICS	Euclidian	Word2Vec	0.24898
		GloVe	-0.04314
		fastText	-0.04976
		BERT	-0.00844
		T5	-0.04567
spectral	Euclidian	Word2Vec	0.24898
		GloVe	0.04000
		fastText	0.02457
		BERT	0.31212
		T5	0.16695
Jarvis-Patrick	Euclidian	Word2Vec	0.01229
		GloVe	0.01012
		fastText	0.00903
		BERT	0.03587
		T5	0.15009
k-means	Cosine	Word2Vec	0.06862
		GloVe	0.10865
		fastText	0.08649
		BERT	0.06052
		T5	0.23428
DBSCAN	Cosine	Word2Vec	0.10450
		GloVe	0.50837
		fastText	0.32180
		BERT	0.35840
		T5	0.00000
OPTICS	Cosine	Word2Vec	0.06945
		GloVe	0.04224
		fastText	-0.16361
		BERT	-0.06468
		T5	0.00982
spectral	Cosine	Word2Vec	0.10450
		GloVe	0.02087
		fastText	0.01714
		BERT	0.37993
		T5	0.38005
Jarvis-Patrick	Cosine	Word2Vec	0.08688
		GloVe	0.03796
		fastText	0.01260
		BERT	0.12950
		T5	0.14816

is necessary to obtain the final result for each clustering method at the first stage. It has been done by eq. (15):

$$R_C(k) = (|Emb|.|Dis|)^{-1} \sum_{i \epsilon Emb} \sum_{j \epsilon Dis} Res(i,j,k) \quad (15)$$

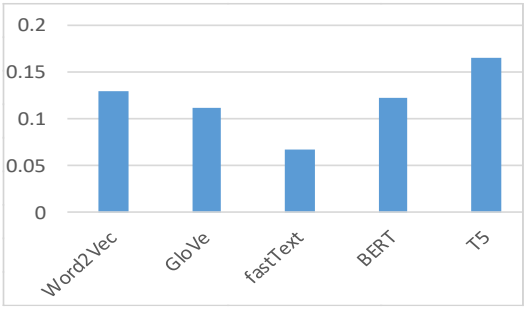

Figure 5: Final results for different embedding methods (R_E) calculated by eq. (13). (Averaged silhouette score.)

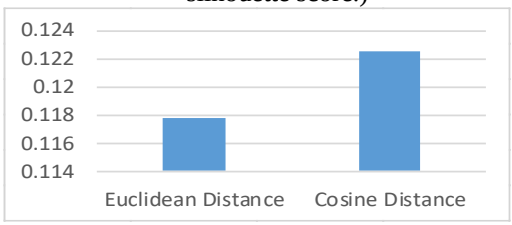

Figure 6: Final results for different distance metrics (R_D) obtained by eq. (14). (Averaged silhouette score.)

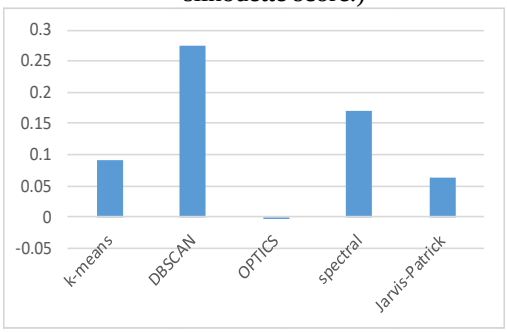

Figure 7: Final results for different clustering methods (R_C) calculated by eq. (15). (Averaged silhouette score.)

where $R_c(k)$ represents the final result of the k^{th} clustering method. Figure 7 shows the obtained results. As it can be seen, DBSCAN has better results over the other clustering methods, followed by spectral. Studying the partial results in different embedding and distance metric methods also indicates that in some cases, spectral has better results than DBSCAN. However, in general, DBSCAN is superior to spectral.

6. Discussion

In the previous section, experiments and their results are presented. Comparing the results for different distance metrics shows cosine distance is a little bit better than Euclidian distance but the difference is low. Therefore, both can be used for topic detection.

T5 performs better than other embedding methods. Embedding of T5 is contextualized i.e. context of word is considered in its embedding. fastText has the worst performance and the difference between its performance and others' is high. Embedding vector of word in fastText is calculated from embedding vectors of its parts. Therefore, embedding a word by its parts is not a suitable solution for topic detection, in contrast to embedding of whole word, specially, contextualized word embedding.

DBSCAN has the best results within clustering methods.

DOI: 10.22034/tjee.2022.15744

Table 4: Ranks of different embedding methods based on Table 3.

	Euclidian Distance					Cosine Distance				
	k-means	DBSCAN	OPTICS	spectral	Jarvis-Patrick	k-means	DBSCAN	OPTICS	spectral	Jarvis-Patrick
Word2Vec	4	4	1	2	3	4	4	1	3	3
GloVe	2	3	3	4	4	2	1	2	4	4
fastText	5	2	5	5	5	3	3	5	5	5
BERT	3	5	2	1	2	5	2	4	2	2
T5	1	1	4	3	1	1	5	3	1	1

Table 5: Rank of cosine distance in comparison with Euclidian distance for different embedding and clustering methods.

	k-means	DBSCAN	OPTICS	spectral	Jarvis-Patrick
Word2Vec	1	2	2	2	1
GloVe	1	1	1	2	1
fastText	1	2	2	2	1
BERT	1	1	2	1	1
T5	1	2	1	1	2

It has much better result than k-means which is a baseline in clustering. k-means can find spherical clusters while DBSCAN can find clusters in any shape. Therefore, it can be concluded that clusters of this dataset are non-spherical. Spectral clustering has the second best result. Spectral clustering can be used to uncover hidden manifold structures [46]. Therefore, its good result is another evidence for non-spherical clusters in the dataset. Spectral uses spectrum of the similarity matrix to reduce dimension before clustering to cluster dataset in fewer dimension. One of the features of nonlinear dimension reduction techniques, is their ability to reduce errors from noise or outliers. Therefore, it can be concluded that outliers exists in dataset and it causes low performance in other methods.

7. Conclusion

This paper follows a comparative approach. This comparison is performed on topic detection. The COVID-19 dataset is also selected for this research. There are several approaches for topic detection from which the clustering approach is chosen in this paper. Clustering requires distance, and distance calculation needs embedding. Therefore, three goals are considered: performance evaluation of 1) embedding methods, 2) distance metrics, and 3) clustering methods. One of the advantages of this work is simultaneously investigating the three factors of embedding methods, distance metrics and clustering methods as well as their interaction. This research has three major questions: 1) Which embedding method has better performance in topic detection on COVID-19 tweets? 2) Which distance metric performs better in topic detection on COVID-19 tweets? 3) Which clustering method outperforms in topic detection on COVID-19 tweets?

Among the embedding methods, five methods are selected: Word2Vec, fastText, GloVe, BERT and T5, including earlier to new methods. Five methods of k-means, DBSCAN, OPTICS, spectral and Jarvis-Patrick, are investigated as clustering methods. Euclidian distance and cosine distance are also studied as the most important distance metrics for topic detection. First, parameter tuning experiments are performed, including more than 7500 cases. Then, all combinations of embedding methods with distance metrics and clustering methods with silhouette score are investigated. The number of these combinations consists of 50 cases. At first, the results of these 50 tests are studied. Then, the rank of each method is considered in all the experiments. At last, the independent variables of the research (embedding methods, distance metrics and clustering methods) are studied separately. In this case, the averaging is applied to neutralize the effect of control variables.

The experimental results show that T5 outperforms other embedding methods in terms of silhouette metric. In addition, T5 has the first rank in most cases when the clustering methods and distance metrics are changed. Therefore, it can be concluded that T5 is strongly better than other embedding methods. Word2Vec is in the second rank after T5. fastText has the weakest results since it has the last rank in most cases while the clustering methods and distance metrics are changed. Comparing distance metrics, cosine distance is weakly better. The cosine distance has better performance when k-means is used. This is also slightly true for the Jarvis-Patrick clustering method. Analysing clustering methods shows that DBSCAN is superior to the other clustering methods.

8. Decleration

The authors declare no conflict of interest in this study.

9. References

[1] Webpage, "Worldometers: Real Time World Statistics," 2022. https://www.worldometers.info/coronavirus/?zarsrc=130.

[2] U.S. CDC, "CDC COVID Data Tracker," *U.S. Centers for Disease Control and Prevention*, 2022. https://covid.cdc.gov/covid-data-tracker/#datatracker-home.

[3] X. Wang, S. Hegde, C. Son, B. Keller, A. Smith, and F. Sasangohar, "Investigating mental health of US college students during the COVID-19 pandemic: Cross-sectional survey study," *J. Med. Internet Res.*, vol. 22, no. 9, p. e22817, Sep. 2020, doi: 10.2196/22817.

[4] A. Zandifar and R. Badrfam, "Iranian mental health during the COVID-19 epidemic," *Asian Journal of Psychiatry*, vol. 51. Elsevier B.V., p. 101990, Jun. 2020, doi: 10.1016/j.ajp.2020.101990.

[5] A. Rafea and N. A. Gaballah, "Topic Detection Approaches in Identifying Topics and Events from Arabic Corpora," *Procedia Comput. Sci.*, vol. 142, pp. 270–277, 2018, doi: 10.1016/j.procs.2018.10.492.

[6] F. Atefeh and W. Khreich, "A survey of techniques

DOI: 10.22034/tjee.2022.15744

for event detection in Twitter," *Comput. Intell.*, vol. 31, no. 1, pp. 133–164, Feb. 2015, doi: 10.1111/coin.12017.
[7] M. Hasan, M. A. Orgun, and R. Schwitter, "A survey on real-time event detection from the Twitter data stream," *J. Inf. Sci.*, vol. 44, no. 4, pp. 443–463, 2018, doi: 10.1177/0165551517698564.
[8] R. Ibrahim, A. Elbagoury, M. S. Kamel, and F. Karray, "Tools and approaches for topic detection from Twitter streams: survey," *Knowl. Inf. Syst.*, vol. 54, no. 3, pp. 511–539, 2018, doi: 10.1007/s10115-017-1081-x.
[9] Z. Mottaghinia, M.-R. Feizi-Derakhshi, L. Farzinvash, and P. Salehpour, "A review of approaches for topic detection in Twitter," *J. Exp. Theor. Artif. Intell.*, pp. 1–27, Jun. 2020, doi: 10.1080/0952813X.2020.1785019.
[10] M. Asgari-Chenaghlu, N. Nikzad-Khasmakhi, and S. Minaee, "Covid-Transformer: Detecting Trending Topics on Twitter Using Universal Sentence Encoder," Sep. 2020, [Online]. Available: http://arxiv.org/abs/2009.03947.
[11] S. R. Nayak, D. R. Nayak, U. Sinha, V. Arora, and R. B. Pachori, "Application of deep learning techniques for detection of COVID-19 cases using chest X-ray images: A comprehensive study," *Biomed. Signal Process. Control*, vol. 64, p. 102365, Feb. 2021, doi: 10.1016/j.bspc.2020.102365.
[12] M. Ahishali *et al.*, "Advance Warning Methodologies for COVID-19 Using Chest X-Ray Images," *IEEE Access*, vol. 9, pp. 41052–41065, 2021, doi: 10.1109/ACCESS.2021.3064927.
[13] M. S. Iraji, M.-R. Feizi-Derakhshi, and J. Tanha, "COVID-19 Detection Using Deep Convolutional Neural Networks and Binary Differential Algorithm-Based Feature Selection from X-Ray Images," *Complexity*, vol. 2021, pp. 1–10, Oct. 2021, doi: 10.1155/2021/9973277.
[14] V. Ravi, H. Narasimhan, C. Chakraborty, and T. D. Pham, "Deep learning-based meta-classifier approach for COVID-19 classification using CT scan and chest X-ray images," *Multimed. Syst.*, Jul. 2021, doi: 10.1007/s00530-021-00826-1.
[15] L. L. Wang *et al.*, "CORD-19: The COVID-19 Open Research Dataset," Apr. 2020, [Online]. Available: http://arxiv.org/abs/2004.10706.
[16] X. Guo, H. Mirzaalian, E. Sabir, A. Jaiswal, and W. Abd-Almageed, "CORD19STS: COVID-19 Semantic Textual Similarity Dataset," Jul. 2020, [Online]. Available: http://arxiv.org/abs/2007.02461.
[17] S. Zong, A. Baheti, W. Xu, and A. Ritter, "Extracting COVID-19 Events from Twitter," Jun. 2020, [Online]. Available: http://arxiv.org/abs/2006.02567.
[18] C. E. Lopez, M. Vasu, and C. Gallemore, "Understanding the perception of COVID-19 policies by mining a multilanguage Twitter dataset," Mar. 2020, [Online]. Available: http://arxiv.org/abs/2003.10359.
[19] E. Chen, K. Lerman, and E. Ferrara, "Tracking Social Media Discourse About the COVID-19 Pandemic: Development of a Public Coronavirus Twitter Data Set," *JMIR Public Heal. Surveill.*, vol. 6, no. 2, p. e19273, May 2020, doi: 10.2196/19273.
[20] R. Tang *et al.*, "Rapidly Bootstrapping a Question Answering Dataset for COVID-19," Apr. 2020, [Online]. Available: http://arxiv.org/abs/2004.11339.
[21] D. Dimitrov *et al.*, "TweetsCOV19 - A Knowledge Base of Semantically Annotated Tweets about the COVID-19 Pandemic," in *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, Oct. 2020, pp. 2991–2998, doi: 10.1145/3340531.3412765.
[22] R. K. Gupta, A. Vishwanath, and Y. Yang, "COVID-19 Twitter Dataset with Latent Topics, Sentiments and Emotions Attributes," 2020. http://arxiv.org/abs/2007.06954.
[23] R. Lamsal, "Design and analysis of a large-scale COVID-19 tweets dataset," *Appl. Intell.*, pp. 1–15, Nov. 2020, doi: 10.1007/s10489-020-02029-z.
[24] J. Samuel, G. G. M. N. Ali, M. M. Rahman, E. Esawi, and Y. Samuel, "COVID-19 Public Sentiment Insights and Machine Learning for Tweets Classification," *Information*, vol. 11, no. 6, p. 314, Jun. 2020, doi: 10.3390/info11060314.
[25] H. Jelodar, Y. Wang, R. Orji, and H. Huang, "Deep Sentiment Classification and Topic Discovery on Novel Coronavirus or COVID-19 Online Discussions: NLP Using LSTM Recurrent Neural Network Approach," *IEEE J. Biomed. Heal. Informatics*, pp. 1–1, Jun. 2020, doi: 10.1109/jbhi.2020.3001216.
[26] J. Xue, J. Chen, C. Chen, C. Zheng, S. Li, and T. Zhu, "Public discourse and sentiment during the COVID-19 pandemic: using Latent Dirichlet Allocation for topic modeling on Twitter," May 2020.
[27] H. Yin, S. Yang, and J. Li, "Detecting Topic and Sentiment Dynamics Due to COVID-19 Pandemic Using Social Media," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 12447 LNAI, pp. 610–623, Jul. 2020, doi: 10.1007/978-3-030-65390-3_46.
[28] R. Chandrasekaran, V. Mehta, T. Valkunde, and E. Moustakas, "Topics, Trends, and Sentiments of Tweets about the COVID-19 Pandemic: Temporal Infoveillance Study," *J. Med. Internet Res.*, vol. 22, no. 10, p. e22624, Oct. 2020, doi: 10.2196/22624.
[29] A. Kruspe, M. Häberle, I. Kuhn, and X. X. Zhu, "Cross-language sentiment analysis of European Twitter messages during the COVID-19 pandemic," 2020.
[30] O. Gencoglu, "Large-Scale, Language-Agnostic Discourse Classification of Tweets During COVID-19," *Mach. Learn. Knowl. Extr.*, vol. 2, no. 4, pp. 603–616, Nov. 2020, doi: 10.3390/make2040032.
[31] L. Li *et al.*, "Characterizing the Propagation of Situational Information in Social Media during COVID-19 Epidemic: A Case Study on Weibo," *IEEE Trans. Comput. Soc. Syst.*, vol. 7, no. 2, pp. 556–562, Apr. 2020, doi: 10.1109/TCSS.2020.2980007.
[32] Q. Jiao and S. Zhang, "A Brief Survey of Word Embedding and Its Recent Development," *IAEAC 2021 - IEEE 5th Adv. Inf. Technol. Electron. Autom. Control Conf.*, vol. 2021, pp. 1697–1701, 2021, doi: 10.1109/IAEAC50856.2021.9390956.
[33] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed Representations of Words and Phrases and their Compositionality," *Adv. Neural Inf. Process. Syst.*, Oct. 2013.
[34] D. Nabergoj, A. D'Alconzo, D. Valerio, and E.

DOI: 10.22034/tjee.2022.15744

Štrumbelj, "Topic extraction by clustering word embeddings on short online texts," *Elektroteh. Vestnik/Electrotechnical Rev.*, vol. 89, no. 1–2, pp. 64–72, 2022.
[35] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching Word Vectors with Subword Information," *Trans. Assoc. Comput. Linguist.*, vol. 5, pp. 135–146, Dec. 2017, doi: 10.1162/tacl_a_00051.
[36] A. Joulin, E. Grave, P. Bojanowski, and T. Mikolov, "Bag of Tricks for Efficient Text Classification," 2017.
[37] J. Pennington, R. Socher, and C. D. Manning, "GloVe: Global Vectors for Word Representation," in *Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 1532–1543.
[38] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in *NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Oct. 2019, vol. 1, pp. 4171–4186, doi: 10.18653/v1/N19-1423.
[39] M. T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," in *Conference Proceedings - EMNLP 2015: Conference on Empirical Methods in Natural Language Processing*, Aug. 2015, pp. 1412–1421, doi: 10.18653/v1/d15-1166.
[40] C. Raffel *et al.*, "Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer," 2020.
[41] A. Vaswani *et al.*, "Attention is all you need," in *Advances in Neural Information Processing Systems*, Jun. 2017, vol. 2017-Decem, pp. 5999–6009.
[42] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise," 1996.
[43] M. Ankerst, M. M. Breunig, H. P. Kriegel, and J. Sander, "OPTICS: Ordering Points to Identify the Clustering Structure," *SIGMOD Rec. (ACM Spec. Interes. Gr. Manag. Data)*, vol. 28, no. 2, pp. 49–60, Jun. 1999, doi: 10.1145/304181.304187.
[44] A. Y. Ng and M. I. Jordan, "On Spectral Clustering: Analysis and an algorithm," in *Advances in neural information processing systems*, 2002, pp. 849–856.
[45] E. A. Patrick, "Clustering Using a Similarity Measure Based on Shared Near Neighbors," *IEEE Trans. Comput.*, vol. C–22, no. 11, pp. 1025–1034, 1973, doi: 10.1109/T-C.1973.223640.
[46] A. Mirzal, "Statistical Analysis of Microarray Data Clustering using NMF, Spectral Clustering, Kmeans, and GMM," *IEEE/ACM Trans. Comput. Biol. Bioinforma.*, vol. 19, no. 2, pp. 1173–1192, 2022, doi: 10.1109/TCBB.2020.3025486.
[47] M. Asgari-Chenaghlu, M.-R. Feizi-Derakhshi, L. Farzinvash, M.-A. Balafar, and C. Motamed, "TopicBERT: A Transformer transfer learning based memory-graph approach for multimodal streaming social media topic detection," Aug. 2020.
[48] S. Dehghani, V. Derhami, A. M. Zare Bidoki, and M. E. Basiri, "Persian Opinion Mining based on Transfer Learning," *Tabriz J. Electr. Eng.*, vol. 50, no. 3, pp. 1215–1224, 2020.
[49] M. A. Z. C. S. Sharifatzadeh, "Compilation Instance Transfer and Feature-representation Transfer for Cross Project Defect Prediction," *Tabriz J. Electr. Eng.*, vol. 48, no. 1, pp. 101–112, 2018.
[50] S. Smith, "Coronavirus (covid19) Tweets - early April," *Kaggle.com*, 2020. https://www.kaggle.com/smid80/coronavirus-covid19-tweets-early-april.
[51] S. Smith, "Coronavirus (covid19) Tweets - late April | Kaggle," 2020. https://www.kaggle.com/smid80/coronavirus-covid19-tweets-late-april.