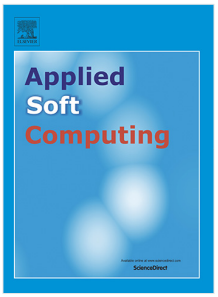
Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID-19. The COVID-19 resource centre is hosted on Elsevier Connect, the company's public news and information website.

Journal Pre-proof



Deep learning modelling of public's sentiments towards temporal evolution of COVID-19 transmission

Ying Wang, Alvin Wei Ze Chew, Limao Zhang

PII: S1568-4946(22)00777-3
DOI: https://doi.org/10.1016/j.asoc.2022.109728
Reference: ASOC 109728

To appear in: *Applied Soft Computing*

Received date : 5 March 2022
Revised date : 24 September 2022
Accepted date : 11 October 2022

Please cite this article as: Y. Wang, A.W.Z. Chew and L. Zhang, Deep learning modelling of public's sentiments towards temporal evolution of COVID-19 transmission, *Applied Soft Computing* (2022), doi: https://doi.org/10.1016/j.asoc.2022.109728.

Deep learning modelling of public's sentiments towards temporal evolution of COVID-19 transmission

Ying Wang[1], Alvin Wei Ze Chew[2], Limao Zhang[3,*]

1. Ph.D. Candidate, School of Civil and Environmental Engineering, Nanyang Technological University, 50 Nanyang Avenue, Singapore 639798. E-mail: ying006@e.ntu.edu.sg
2. Research Engineer, Ph.D., Bentley Systems Research Office, Singapore, 1 Harbourfront Pl, HarbourFront Tower One, Singapore 098633. Email: Alvin.Chew@bentley.com
3. Assistant professor, School of Civil and Environmental Engineering, Nanyang Technological University, 50 Nanyang Avenue, Singapore 639798. E-mail: limao.zhang@ntu.edu.sg

* Corresponding author: Limao Zhang (limao.zhang@ntu.edu.sg)

Abstract: Public sentiments towards global pandemics are important for public health assessment and disease control. This study develops a modularized deep learning framework to quantify public sentiments towards COVID-19, followed by leveraging the predicted sentiments to model and forecast the daily growth rate of confirmed COVID-19 cases globally, via a proposed G parameter. In the proposed framework, public sentiments are first modeled via a valence dimensional indicator, instead of discrete schemas, and are classified into 4 primary emotional categories: (a) neutral; (b) negative; (c) positive; (d) ambivalent, by using multiple word embedding models and classifiers for text sentiments analyses and classification. The trained model is subsequently applied to analyze large volumes (millions in quantity) of daily Tweets pertaining to COVID-19, ranging from 22 Jan 2020 to 10 May 2020. The results demonstrate that the global community gradually evokes both positive and negative sentiments towards COVID-19 over time compared to the dominant neural emotion at its inception. The predicted time-series sentiments are then leveraged to train a deep neural network (DNN) to model and forecast the G parameter by achieving the lowest possible mean absolute percentage error (MAPE) score of around 17.0% during the model's testing step with the optimal model configuration.

Keywords: Text sentiment classification, Global sentiment evolution, COVID-19 transmission, Deep learning, Twitter data, Natural language processing

1. Introduction

Since its inception in late 2019, the coronavirus disease 2019 (COVID-19) has already been regarded as one of the greatest crises faced by humanity in the 21st century. The virus COVID-19 is not as deadly as a severe acute respiratory syndrome (SARS), however, the novel virus is generally more infectious to the general population. In addition, COVID-19 can be more fatal to the elderly group as the death rate has already reached more than 8% for those aged between 70 and 79, and above 14.8% for those 80 years old and above [1]. To control the spread of COVID-19, imposing social distancing and community lockdown measures have since been implemented in many countries [2,3]. However, while generally effective, prolonged lockdown policies are likely to impose a negative emotional impact on the general population. This guess lacks statistical evidence.

Collaboration among the different nations is key to addressing the current pandemic situation as the daily growth rate in the number of confirmed COVID-19 cases indicates that no single country has, by far, been able to effectively control the virus spread while keeping their borders fully open. According to the Global Health Security Index proposed by Economist Intelligent Unit (EIU) in 2019, the global average score of preparedness level in handling an epidemic or pandemic is 40 out of 100, and even among the most developed and high-income countries, the average score is 51.9 [4]. The EIU assessment underlines an imperative and urgent need to modify the present strategies to manage the COVID-19 outbreak in the global context. However, it is time-consuming and tedious to collate near real-time information pertaining to the dynamic COVID-19 behavior globally, and especially difficult to manage the social mobility of individuals in this challenging period. Social media platforms, even with the risk of misinformation and privacy leakage [5], provide alternative avenues for data collection in the applications of pandemic monitoring [6], participatory governance development [7], interaction modelling among cities [8], and disaster recovery [9]. Effective policy on pandemic control and prevention relies on accurate information on the current pandemic situation and its potential trend soon. An overview of the data analytics for epidemic monitoring and control can be found in Feng et al. [10].

Epidemic models are used to analyze COVID-19 evolution for insight and future guidance. There are two types of epidemic models, i.e., data-driven models and mechanistic models. Data-driven models focus on predictions about the near future according to past data [11]. In comparison, mechanistic models simulate transmission dynamics [12]. Mechanistic models can project the complete situation from the onset of the pandemic to a global stable state, which is mathematically determined as pandemic-free states [13]. Compartment models, including the classic susceptible-exposed-infectious-removed (SEIR) model and its variants, are the representatives of mechanistic models [14]. However, at the early stage, little about the pandemic is known and it is hard to select the proper parameters (e.g., contact rate) to generate curves that fit the real situations. Instead, data-driven models do not rely on assumptions about parameters. A spike in coronavirus cases is affected by several factors that have complex underlying dependencies. The triggers can be changes to the coronavirus (e.g., the Delta variant in May 2021 and the Omicron variant in November 2021), human mobility patterns, infection prevention policies, the effectiveness of vaccines over time, and the

vulnerable populations. Historical data alone do not generate precise predictions, especially when a spike arises.

Public sentiments not only passively reflect the public's perception of COVID-19 but also actively affect their behaviors (e.g., mobility patterns), which in turn will affect the COVID-19 situations [15]. For example, traditional methods leveraging on questionnaires exposed that lockdowns can distort individuals' time perception, which further adversely affects their emotions, stress level, perceived task complexity, and other cognitive abilities [16]. In several megacities, sentiment shows a certain degree of correlation with the indicators about the ongoing COVID-19 conditions, such as quarantine, new cases, hospitalization, and deaths [17,18]. On the other hand, the triggered from emergencies can be used as cues for predictive analysis of ongoing situations, as exemplified by a recent study conducted in Houston, Texas (United States, US) [19]. Additionally, text mining techniques have been applied to internet information to detect possible signals having underlying important information about the transmission of COVID-19 [20], which thus serve as early warnings to the local and global communities [21].

There is no conclusive evidence of the public's sentiments towards COVID-19 since its inception considering geographic boundaries. Empirical studies show gradually developed anger in the global population during 2020 [22], varying degrees of augmentation in emotions within the Chinese local community [23], and both fear and passion towards lockdown [24]. Current studies on COVID-19 sentimental responses have the limitations of binary classification of sentiments (i.e., positive and negative) which ignores a common condition of neural posts [25], and ignorance of the time index [26,27]. Considering several separate periods, Marathe et al. concluded that negative sentiment had increased than stabilized during the four lockdowns in India based on Twitter data [28]. Interestingly, a study on sentiment towards online learning in the post-pandemic period showed that neural sentiment dominates according to Twitter analysis while negative sentiment dominates by questionnaires [29]. However, the sample size is small for both analyses (i.e., 5000 Tweets and less than 100 questionnaire responses). Efforts have been paid to seek other evidence for pandemic prediction and control, such as infection probability under varying distances from the source of infection [30], visitors' trajectory data for crowd control [31], and aggregators by demographic information [32]. In summary, empirical evidence of global sentiment variations with a fine time resolution (i.e., daily) towards the evolving COVID-19 is scares. Besides, the effect of collective sentiment on the prediction of the pandemic situation remains unknown.

To better inform the current pandemic situation and project it to the near future state, this study intends to (i) quantify the public's sentiment towards COVID-19 with a daily resolution by processing a large volume of COVID-19 Tweets, which are responses to historical COVID-19 situations, and (ii) to empirically test the feasibility of incorporating sentiment analysis results to predict future COVID-19 situation at a global scale, where it is assumed that sentimental responses towards COVID-19, in turn, will stabilize the predication of future pandemic situation. The study investigates the use of several deep learning pipelines, i.e., frameworks, to perform extensive sentiment classification analyses towards COVID-19 by exploiting the availability of emotional responses-related datasets, followed by performing transfer learning to analyze large volumes of daily COVID-19-associated

Twitter data (between January 2020 to May 2020) to develop a level of quantitative understanding of the general populations' emotional responses towards the current pandemic. Following this, the sentiment results are leveraged to forecast the temporal spread of COVID-19 via the daily increase rate in the number of confirmed cases globally. In summary, this study contributes in the following aspects: (i) developing a modularized deep learning model framework for text sentiment analysis and validating the prediction results on multiple open-source datasets; (ii) initiating a transfer learning process that enables the adaptation of the trained deep learning model to analyze the temporal evolution of the global sentiments towards COVID-19; and (iii) correlating sentiment category distribution, as derived from the preceding transfer learning step, to model and forecast, via deep learning, the growth rate in the confirmed number of COVID-19 cases globally.

The rest of the paper is structured as follows. Section 2 reviews the existing methods in the literature for text sentiment classification analysis. Section 3 describes the architecture of this modularized deep learning model(s) for processing and classifying text sentiments and followed a predictive analysis of the ongoing situation. Section 4 describes the open-source datasets leveraged to train, validate, and test the proposed deep learning model(s), results, and discussions. Finally, Section 5 succinctly summarizes the key findings obtained from this study, as well as future works.

2. Literature review on sentiment analysis

Sentiment analysis is an important task in the domain of natural language processing (NLP) and language modelling [33]. However, sentiment classification from short texts generally presents a two-fold challenge, namely: (i) there are currently no quick labeling methods for text sentiments which may hinder supervised learning for NLP analysis for effective basic deployment for multiple applications [34,35]; (ii) ambiguity is likely to occur which can affect the resulting model's classification accuracy [36]; (iii) content quality may differ across different datasets which again can hinder the training phase of the language model [31]. To address these issues, studies have been conducted to improve different variants of NLP models to improve performance for sentiment analysis, as well as other related machine learning tasks [37]. Generally, sentiment classification can be grouped into two main categories, namely lexicon-based and machine learning categories.

Lexicon-based methods skip the traditional model training step and instead focus on the semantic orientation of separate words. In corpus-based methods, certain words are correlated with positive or negative sentiments. For example, the word "excellent" is considered with a positive polarity, and the word "poor" is considered with a negative polarity [38]. In dictionary-based methods, where lexicon is used, keywords reflecting sentiments in the document are explicitly used to evaluate the sentiments of the text. The dependence on specific words, however, limits the generalization of the available corpus or lexicon in sentiment analysis towards other topics. Besides, text writers may not explicitly express their emotions, such as by using adjectives. Besides, statistical methods do not consider contextual information, and thus developed models may suffer from contextual polarity. Considering the large effort to build a topic-centered corpus or lexicon, domain adoption is an important issue to address effectively, where a general-purpose sentiment lexicon has since been introduced which

performs well as domain-specific lexicons [39]. Several sentiment lexicons have been combined and reviewed for domain adaption purposes for classifying available sentiments of product reviews [40].

Machine learning methods belong to supervised feature-based learning and can usually improve the performance of sentiment analyses. Common features include unigram, bigram, n-gram, word embedding, and parts of speech. These features are leveraged as inputs to different types of classifiers to model and predict the label of the sentiment itself. The derived classification results depend on several factors which include, but are not limited to, the type of feature engineering method used for data pre-processing, development of the classifier model, labeling quality of the raw/processed datasets, as well as the task objective itself. For instance, Naïve Bayes, maximum entropy, support vector machine (SVM) achieved around 80% classification accuracy for binary classification task on the dataset named *sentiment140* by using features derived from unigram and bigram [41]. In comparison, the performance of multi-class sentiment classification is of limited accuracy. For example, studies on the two open-source Twitter datasets, including *CrowdFlower* and *Electoral-Tweets*, reported low F1-scores of 0.32 and 0.31 respectively [35].

Deep learning architectures such as convolutional neural networks (CNN), recurrent neural networks (RNN), and Long Short-Term Memory (LSTM) [42,43], serve as good alternatives to analyze sentiments. Combinations of different architectures are expected to improve the accuracy performance of sentiment analysis. For example, a hybrid approach integrating CNN and LSTM has been shown to significantly improve the resulting accuracy for classifying different types of sentiments [44]. Attention neural networks have been used to address aspect-level sentiments [45] and multi-domain sentiments [46]. In summary, effective text sentiment classification requires the careful selection of text features and classifiers, which can be tedious and time-consuming. Hence, developing a modular architecture for sentiment analysis will allow for easy modifications and model refinements, as well as provide improved interpretability.

Transfer learning in sentiment classification is promising as it reduces the workload required to annotate new data and integrates the inherent characteristics of pre-existing labeled datasets used for training the model previously [47-49]. Cross-corpus sentiment classification problems, either on identical or diverse domains, can be approached by leveraging the source dataset(s) having rich sentiment labels to analyze the target dataset, where the latter lacks suitable labels for the supervised learning task. The key objective in performing a cross-corpus sentiment classification task is to extract corpus invariant features to bridge the source and the target (or unseen) dataset for integrating corpus invariant information between two datasets. Transfer learning has also been coupled with deep learning analysis for sentiment classification. For example, manifold regularization is used to enhance a semi-supervised framework for cross-corpus sentiment classification [50], while transfer network using deep learning has demonstrated good model performance in classifying sentiment polarity on cross-domain topics [47]. However, selecting the optimal source dataset and pre-trained classifier model remains an open research question in the domain of classifying text sentiments.

Overall, sentiment is a fundamental element in demonstrating one's cognition and brain activities, and responses to an action. Text sentiment analysis using the Twitter dataset contains more information about users' preferences and cognitive states. The sentiment analysis results can then be leveraged in

many applications such as social network analysis, recommender systems, and trend prediction. For example, sentiments and emotions from Tweets have been used to identify user clusters for recommendation purposes [51], where the "mood", i.e. fluctuations, of Twitter data has been determined to be an effective economic indicator for short- and long-term stock predictions [52,53]. In summary, sentiment analysis is an important topic in the NLP domain, hence the proposed modularized deep learning approach in this study aims to quantitatively investigate the complex relationship between global public sentiments and the transmissivity of COVID-19 on the global scale.

3. Methodology

The proposed modularized deep learning approach investigates the feasibility of using existing open-source datasets to analyze global sentiments towards COVID-19. The approach aims to first conduct sentiment analysis using relevant Twitter data where the generated sentiments results. Coupled with the growing number of COVID-19-related Tweets, the sentiment analysis results are used as model input features to model and forecast the growth rate in the confirmed number of COVID-19 cases globally. The in-built sentiment analysis in the proposed approach consists of a series of systematic analyses, where each addresses specific tasks as follows: (i) building word embedding models from the available corpus in the open datasets using deep learning techniques; (ii) constructing sentence representation using the built word embeddings from the preceding step; (iii) classifying the text sentiments based on the sentence representation.

Situational predictive analysis correlates and explores the quantitative relationship between different sentiments and the daily increase rate of confirmed COVID-19 cases globally. The framework of the proposed modular deep learning model for text sentiment and situational predictive analyses, coupled with transfer learning, is illustrated in Fig. 1. For model training, datasets with and without labeled sentiment classes are selected as the corpora to train the selected word embedding model. The word representations derived from the trained word embedding models are then leveraged as input model features for training and validating selected classifier(s). The predictive capability of the trained classifier(s) is then subsequently applied for the sentiment analysis of new unlabeled texts concerning the target objective. Finally, the derived quantitative sentiment results are then exploited to model and predict the daily increase rate of COVID-19 on 3 structurally different neural networks (NN) regressors.

3.1. Data preprocessing

Various preprocessing techniques are used to pre-process the available text data by removing punctuations, stop words, and non-English words for reducing the "noises" and feature dimensions of the learned word vector. As stop words are common words with high occurrence frequency but little semantic meaning, such as "ourselves", "hers", "with", etc., the stop words ae removed in the analysis using the stop words corpus in the open-source python library Natural Language Toolkit (NLTK). Similarly, non-English words are removed using the English words corpus from the NLTK library.
Every original, i.e., raw, Twitter post is tokenized into individual words, followed by comparing the tokenized words with the available pool of punctuations, stop words corpus, and English words corpus

for text cleaning. For example, considering an original Tweet as “When I couldn't find hand sanitizer at Fred Meyer, I turned to #Amazon. But $114.97 for a 2 pack of Purell??!!Check out how #coronavirus concerns are driving up prices. https://t.co/ygbipBflMY”. By removing the available punctuations, stop words, and non-English words, the remaining tokenized words are ['I', 'could', 'find', 'hand', 'Fred', 'I', 'turned', 'Amazon', 'pack', 'driving'].

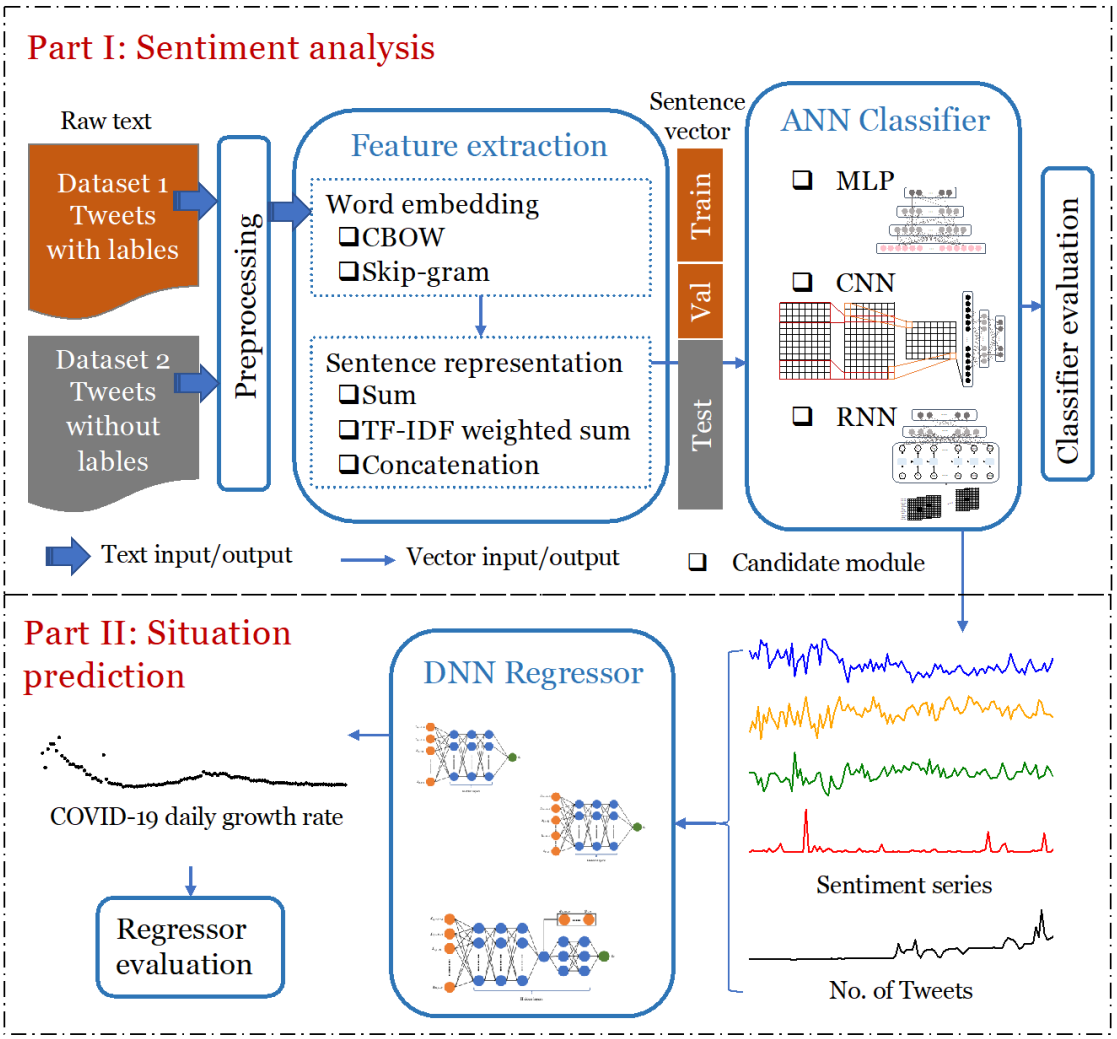


Fig. 1. The framework of deep learning in sentiment analysis and situation prediction towards COVID-19. The training dataset (dataset 1) and target dataset (dataset 2) are fed to the language model so that their word vectors are obtained on a shared corpus, after which the samples from the training dataset are used for training and validation to calibrate the classifier, and the samples from the target dataset are analyzed using the calibrated classifier. The sentiment analysis results, together with the no. of tweets, are used as features to predict the daily increase rate of confirmed COVID-19 cases.

3.2. Text feature extraction

3.2.1. Word embedding

Word embedding learns the word representation from the corpus. Representing tokenized words as word vectors is the first step in almost all NLP tasks. Word embedding is the representation of a unique word using a one-dimensional (1D) vector. There are two types of methods to derive the word vectors: singular value decomposition (SVD) and iteration-based methods. SVD methods solve the problem by counting the occurrence of a word in a document which is denoted as a word-document matrix, or by counting the co-occurrence of two words which is represented as a window-based co-occurrence matrix. These methods are often associated with underlying problems of dynamic word size, high dimensional words, and extremely sparse matrix as a significantly large number of English words do not co-occur. Instead of computing and storing huge datasets, iteration-based models update the probability in each iteration and thus solve the above-mentioned problems more efficiently. Example iteration-based methods are the unigram, bigram, continuous bag of words (CBOW) model, and Skip-gram model. CBOW and Skip-gram are two promising models of lexical semantics and have been demonstrated to perform markedly better in encapsulating semantic relatedness than other language analytic models [54]. Both CBOW and Skip-gram are described briefly below, which serve as the two primary candidates for the first component of the proposed modular sentiment classification architecture.

CBOW and Skip-gram models learn the word vectors by optimizing the probability of word occurrence in a stream of text via a neural network with one hidden layer, i.e., a shallow deep neural network model. CBOW predicts the missing center word given the context words. Conversely, Skip-gram predicts the context words given the center word. The objective function to be minimized in CBOW is the cross-entropy of probability as expressed in Eq. (1). Intuitively, CBOW finds the center word with the maximum probability given the context words. On the contrary, the cost function used in the Skip-gram model is expressed in Eq. (2). Skip-gram locates the output words with the maximum product of probabilities corresponding to each output word given the input center word. The input matrix, output matrix, and neural network parameters are solved using backpropagation and a stochastic optimizer.

$$\begin{aligned} \text{Minimize } J &= -\log P(w_c|w_{c-m},\dots,w_{c-1},w_{c+1},\dots,w_{c+m}) \\ &= -\log P(u_c|\hat{v}) \\ &= -\log \frac{\exp(u_c^T\hat{v})}{\sum_{j=1}^{|V|}\exp(u_j^T\hat{v})} \\ &= -u_c^T\hat{v} + \log\sum_{j=1}^{|V|}\exp(u_j^T\hat{v}) \end{aligned} \tag{1}$$

where w_c is the targeted center word, $w_{c-m},\dots,w_{c-1},w_{c+1},\dots,w_{c+m}$ are the context words around the center word with a window of size m, $u_c \in R^n$ is the $n \times 1$ output vector representation of a center word w_c, $\hat{v} = \frac{v_{c-m}+v_{c-m+1}+\cdots+v_{c+m}}{2m} \in R^n$ is the average of the input vectors corresponding to the input context words, and $|V|$ is the number of possible output words.

$$\begin{aligned}\text{Minimize } J &= -\log P(w_{c-m},\dots,w_{c-1},w_{c+1},\dots,w_{c+m}|w_c)\\ &= -\log \prod_{j=0,j\neq m}^{2m} P\left(w_{c-m+j}|w_c\right)\\ &= -\log \prod_{j=0,j\neq m}^{2m} P\left(u_{c-m+j}|v_c\right)\\ &= -\log \prod_{j=0,j\neq m}^{2m} \frac{\exp\left(u_{c-m+j}^T v_c\right)}{\sum_{k=1}^{|V|}\exp(u_k^T v_c)}\\ &= -\sum_{j=0,j\neq m}^{2m} u_{c-m+j}^T v_c + 2m\log\sum_{k=1}^{|V|}\exp\left(u_k^T u_c\right)\end{aligned} \tag{2}$$

where w_c is the targeted center word, $w_{c-m},\dots,w_{c-1},w_{c+1},\dots,w_{c+m}$ are the context words around the center word with a window of size m, $v_c \in R^n$ is the $n \times 1$ input vector representation of a center word w_c, $u_{c-m+j} \in R^n$ is the output vector representation of each context word, and $|V|$ is the number of possible output words.

3.2.2. Sentence representation

Sentiment classification is usually performed at a sentence- or paragraph-level for each text corpus. For consistency and convenience, "sentence" refers to short texts that have one sentiment label, even if it contains more than one sentence. This part explains the techniques used to construct sentence representation from available word vectors, including sum, term frequency-inverse document frequency (TF-IDF) weighted sum, and concatenation of word vectors, where the derived results will be fed to the proposed classification model in the subsequent stages.

By removing punctuations and non-English words from the original sentence, each tokenized list of words derived can be denoted as $S = [w_1, w_2, \dots, w_l]$ where w_i is a word in the sentence and l is the total number of words in this message. Some words will not have the required word embeddings in the trained word2vec model (CBOW or Skip-gram) as both models learn the word vectors from a certain corpus which will remove very rare-occurring words via a parameter termed "min_count". Those words without word vectors in the corpus are assigned zero values in all dimensions.

Both the sum and TF-IDF weighted sum methods maintain a common defined 1D size of sentence vector for consideration as a word-vector. The sum method constructs the sentence vector from tokenized words via Eq. (3). TF-IDF is one method that assesses the weight of each word considering its occurrence frequency and relative importance in the corpus. TF counts the occurrence frequency of each word in one document, which is then divided by the document length to avoid a preference for long documents. IDF is the log of total documents divided by the number of documents containing certain words. IDF measures the importance of each word assuming less importance of words that appear in more documents. TF-IDF is the product of the two scores TF and IDF. TF-IDF weight sum method constructs the sentence vector from tokenized words by Eq. (4).

$$s = v_1 + v_2 + \cdots + v_l \tag{3}$$

where $v_1, v_2, \dots, v_l \in R^n$ are the word vectors of the tokenized words in the given sentence, and $s \in R^n$ is the sentence vector.

$$s = c_1 v_1 + c_2 v_2 + \cdots + c_l v_l \quad (4)$$

where $v_1, v_2, \dots, v_l \in R^n$ are the word vectors of the tokenized words in the given sentence, $c_1, c_2, \dots, c_l$ are the corresponding weight of each tokenized word from the TF-IDF model, and $s \in R^n$ is the sentence vector.

Concatenation constructs the sentence representation from word vectors by connecting the 1D vectors into a 2D matrix according to Eq. (5). Compared to posting-padding, pre-padding is determined to be more effective in CNN and LSTM for NLP-related tasks [55]. Sentence vectors are pre-padded with zero vectors to the maximum length of the sentence sequence before they are fed into a classification model.

$$s = [v_1^T; v_2^T; \dots; v_l^T] \quad (5)$$

where $v_1, v_2, \dots, v_l \in R^n$ are the column word vectors of the tokenized words in the given sentence, and $s \in R^{n \times l}$ is the sentence matrix. Each row of s represents the word vector of one word in the sentence.

3.2.3. Co-corpus transfer learning

Given two independently collected datasets of text corpora, the main purpose of transfer learning is to develop one specific sentence transformation which can best represent the features of the samples from the two datasets under the same distribution of tokenized words. That is, one cannot distinguish which dataset the sample comes from using the transferred features [56]. To achieve this objective, the texts from the two datasets are pre-processed in the same manner and fed into one word embedding model, so that the resulting output representing the learned word-vector shares the same corpus.

3.3. Learning classifiers

The classifier is a key module that predicts the label given the word vectors or sentence vectors for the sentiment classification task. The neural network has demonstrated promising performance in NLP tasks, where multiple deep learning architectures have been proposed. There is no consistent conclusion about which neural networks collectively show an excellent performance towards a certain task. Three NN classifiers with varying manipulations of the position information are compared in terms of the performance of classification. Specifically, MLP does not include any position information about the context words. CNN provides local position information about the context words. RNN provides extra memory about the context words. As there is no rule of thumb about the best choice for sentiment classification tasks on a certain dataset, the three NN classifiers are tested on multiple datasets to select a more advanced network structure. These three NN classifiers, two-layer multilayer perceptron (MLP), CNN, and RNN are described below.

3.3.1. Two-layer MLP

A two-layer feed-forward neural network enables the non-linearity of the inputs in predicting the sentiment label from the sentence vector. The input layer is the sentence representation $s \in R^n$ from

the previous module. The neurons in the first hidden layer are calculated as the weighted sum of input neurons with a bias term activated by a nonlinear function, as shown in Eq. (6). The neurons in the second hidden layer are calculated as the weighted sum of neurons in the first hidden layer with a bias term activated by a non-linear function, as shown in Eq. (7). The neurons in the output layer are calculated as a weighted sum of the second hidden layer. The softmax function is used to get the probability over each class, as shown in Eq. (8). A visual illustration of the feed-forward neural network is shown in Fig. 2.

$$h_1 = \sigma(W^1 s + b^1) \tag{6}$$

where $W^1 \in R^{d_1 \times n}$ is the weight matrix connecting the input layer and the first hidden layer, d_1 is the number of neurons in the first hidden layer, $b^1 \in R^{d_1}$ is the bias terms for the input layer, $\sigma(\cdot)$ is the activation function, and h_1 is the vector representation for the neurons in the first hidden layer. In this work, the activation function uses an exponential linear unit (ELU).

$$h_2 = \sigma(W^2 h_1 + b^2) \tag{7}$$

where $W^2 \in R^{d_2 \times d_1}$ is the weight matrix connecting the first hidden layer and the second hidden layer, d_2 is the number of neurons in the second hidden layer, $b^2 \in R^{d_2}$ is the bias terms for the first hidden layer, $\sigma(\cdot)$ is the activation function, and h_2 is the vector representation for the neurons in the second hidden layer. In this work, the activation function uses the ELU.

$$p = \text{softmax}(W^3 h_2) \tag{8}$$

where $W^3 \in R^{d_o \times d_2}$ is the weight matrix connecting the second hidden layer to the output layer, d_o is the number of neurons in the output layer representing the number of classes, and $\text{softmax}(z_i) = \frac{\exp z_i}{\sum_{j=1}^{d_o} \exp z_j}$ transfers the output into normalized probability over each class.

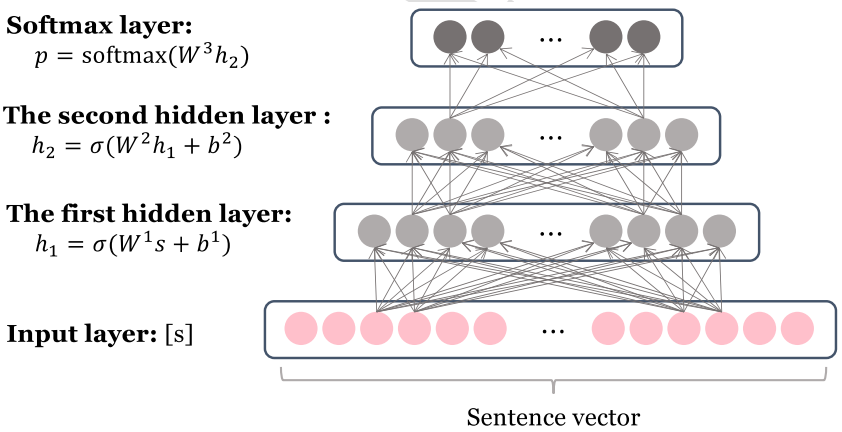


Fig. 2. Two-layer MLP for classifying texts.

3.3.2. CNN

CNN combines the local features of the input sentence matrix by sliding a convolutional window along a spatial dimension of the input matrix. The resultant feature map is further reduced by a max-pooling layer, which picks out the maximum value over a defined pooling window. The CNN used in this study includes one convolutional layer, one max-pooling layer, one flatten layer, and two dense layers. Denote the input sentence matrix as $s = [v_1^T; v_2^T; \ldots; v_l^T]$, and $s_{i:j} = [v_i^T; \ldots; v_j^T]$ is the

stitching of i^{th} to j^{th} word vectors in s. The convolution operation to obtain the feature map C is shown in Eq. (9). The feature map C is reduced by a factor of "pool_size", which defines the height of the pooling window. The reduced 2D feature map is then flattened to 1D features, which is fed to the dense layer.

A simple CNN architecture for classifying texts is shown in Fig. 3. An input sentence matrix is shown as an example of the input format. The sentence matrix formulation from the raw text is briefly illustrated below. For instance, the original Tweet text is "I should be sleep, but im not! thinking about an old friend who I want. but he's married now. damn, & he wants me 2! scandalous!". After pre-processing, the tokenized words are ['I', 'sleep', 'thinking', 'old', 'friend', 'I', 'want', 'married', 'damn', 'scandalous']. It can be observed that punctuations and some words that are recognized as stop-words or non-English words have been removed in the pre-processing stage. The sentence matrix is formed by concatenating the word vectors from the pre-trained word embedding model. It can be observed that the words 'I' and 'scandalous' are not in the model due to their too high or too low frequency in the whole corpus and are substituted with zero vectors. Pre-padding with zero vectors is used to unify the size of the input and to get the feature map of the convolution operation for the first several rows.

$$C_j = f(W \circ s_{i:j} + b) \tag{9}$$

where $\circ$ is the element-wise multiplication, W and b denote the weight matrix and bias terms for the convolution kernel, respectively, and f is the activation function.

A 1D convolution layer is used to perform the convolution operations. The number of filters is chosen among the dimension of the word vector n, $\frac{n}{2}$, and $\frac{n}{4}$. Kernel size is chosen among 3, 4, and 5. Causal padding is used, where zero vectors are padded for the first several times of implementation of a convolutional operator. The movement per step is set to 1. ReLU is used as the activation function.

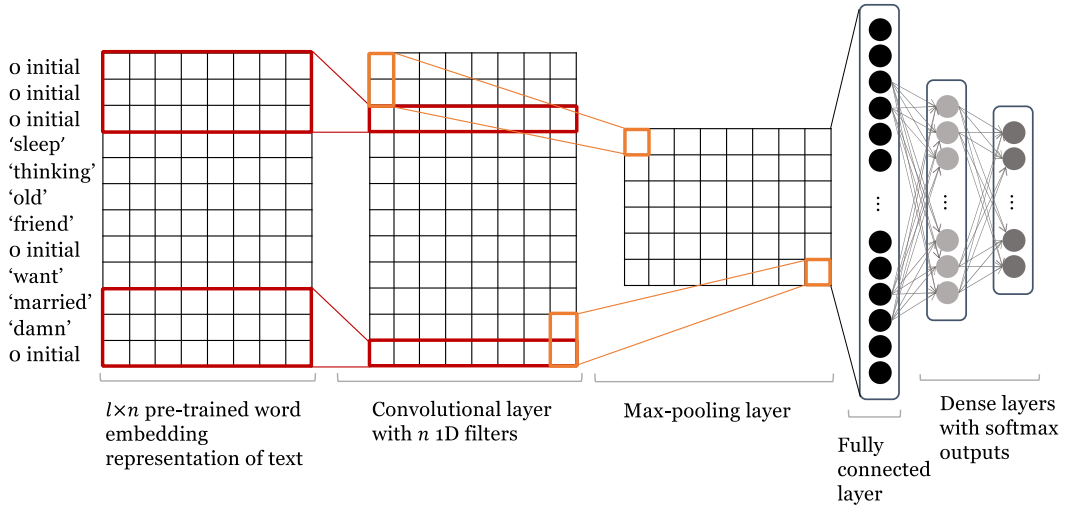


Fig. 3. A simple CNN architecture for classifying texts.

3.3.3. RNN

RNN propagates context information through faraway time-steps. The gradient descent method is often used to find the best weight matrices of RNN by optimizing the loss function. Learning long-

term dependency using RNN is challenging. For example, it suffers from a vanishing gradient problem, where the gradient value goes to zero in the backpropagation process [57]. LSTM is an improved model of RNN and performs better with the use of more complex units of activation. LSTM units are found to have more persistent memory and can selectively remember patterns for long durations of time. LSTM unit depends on both the old state $h^{(t-1)}$ and the input $x^{(t)}$.

LSTM unit contains three gates: input gate, forget gate, and output gate. The mathematical formulation of LSTM units is shown in Eqs. (10)-(15). Intuitively, new memory is generated based on the input words $x^{(t)}$ and the past hidden state $h^{(t-1)}$. The input gate estimates the importance of this newly generated memory. Similarly, the output gate evaluates the usefulness of memory in the calculation of current memory. The gated new memory and gated memory are combined to form the final memory. The output gate controls how much information in the final memory is stored in the hidden state, which will be passed to the next LSTM unit. A basic RNN architecture for classifying texts including the input layer, RNN layer, and two dense layers with softmax output is shown in Fig. 4. The number of units in the LSTM layer is picked among the three values, 20, 50, and 100 based on their performance.

$$i^{(t)} = \sigma\left(W^{(i)}x^{(t)} + U^{(i)}h^{(t-1)} + b^{(i)}\right) \tag{10}$$

$$f^{(t)} = \sigma\left(W^{(f)}x^{(t)} + U^{(f)}h^{(t-1)} + b^{(f)}\right) \tag{11}$$

$$o^{(t)} = \sigma\left(W^{(o)}x^{(t)} + U^{(o)}h^{(t-1)} + b^{(o)}\right) \tag{12}$$

$$u^{(t)} = \tanh\left(W^{(u)}x^{(t)} + U^{(u)}h^{(t-1)} + b^{(u)}\right) \tag{13}$$

$$c^{(t)} = i^{(t)} \circ u^{(t)} + f^{(t)} \circ c^{(t-1)} \tag{14}$$

$$h^{(t)} = o^{(t)} \circ \tanh\left(c^{(t)}\right) \tag{15}$$

where $i^{(t)}, f^{(t)}, o^{(t)}$ are the outputs of the input gate, forget gate, and output gate, respectively, $u^{(t)}$ is a new memory, $c^{(t)}$ is the final memory, $h^{(t)}$ is the new hidden state of the LSTM unit, $\sigma(\cdot)$ is the sigmoid activation function, $W^{(i)}$ and $U^{(i)}$ are weights for input gate, $W^{(f)}$ and $U^{(f)}$ are weights for forget gate, $W^{(o)}$ and $U^{(o)}$ are the weights for the output gate, $W^{(u)}$ and $U^{(u)}$ are the weights for generating new memory, $b^{(i)}, b^{(f)}, b^{(o)}$, and $b^{(u)}$ are the bias terms for the input gate, forget gate, output gate, and new memory generation, respectively, and $\circ$ denotes element-wise multiplication.

3.4. Classifier loss function and performance evaluation

Cross entropy is used as the loss function in training the ANN classifiers. Cross entropy for multi-class classification per observation is expressed in Eq. (16). The cross-entropy of each observation is summed to form the loss of the classifier.

$$L = -\sum_{c=1}^{m} y_{o,c} \log p_{o,c} \tag{16}$$

where m is the number of classes, $y_{o,c}$ is a binary indicator representing whether class label c is a correct observation o, and $p_{o,c}$ is the predicted probability that observation o is of class c.

Accuracy is the ratio of correctly classified samples to the total number of samples, as shown in Eq. (17). To test the modular architecture for text classification, a labeled dataset is split into training

samples (64%), validation samples (16%), and testing samples (20%). The performance of the model is evaluated by training accuracy, validation accuracy, and testing accuracy. In transfer learning, the labeled dataset is split into training samples (80%) and validation samples (20%), and the target dataset without labels is tested by the calibrated model using the labeled dataset.

$$Acc = \frac{no.of\ correctly\ classified\ samples}{no.of\ total\ samples} \tag{17}$$

where Acc denotes the accuracy.

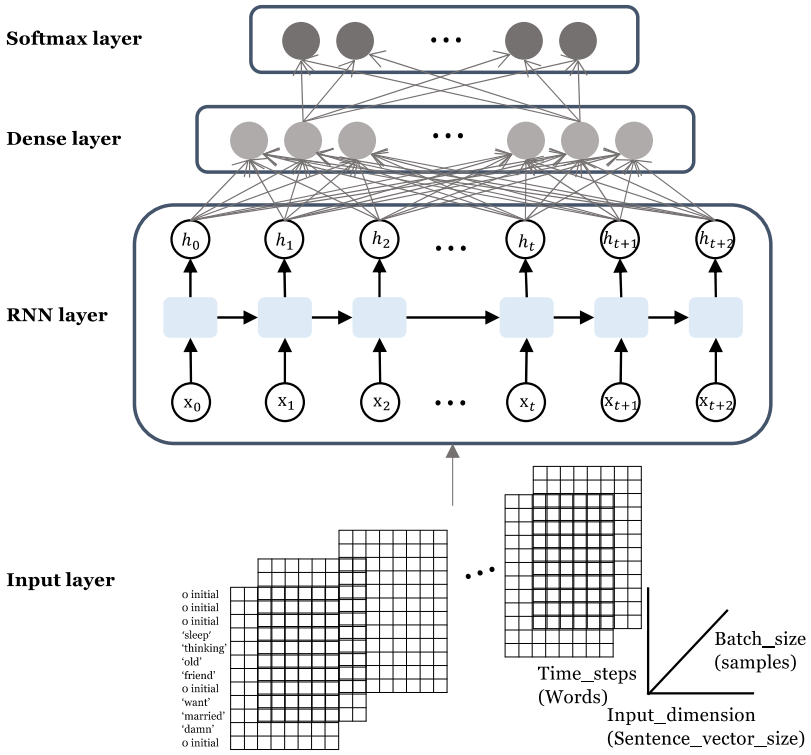


Fig. 4. A basic RNN architecture for classifying texts.

3.5. Modelling COVID-19 temporal evolution using predicted sentiments

The predicted sentiments from the trained classifiers, as part of the proposed modularized deep learning framework (Fig. 1), are subsequently leveraged as model input features to train, validate, and test personalized deep neural networks (DNNs) to model and forecast the temporal evolution in the total number of confirmed COVID-19 cases on a global context, via a proposed G_t parameter as defined in Eq. (18).

$$G_t = \frac{Y_t - Y_{t-1}}{Y_{t-1}} \times 100\% \tag{18}$$

where Y_t represents the global number of confirmed COVID-19 cases at time (t), and Y_{t-1} represents the global number of COVID-19 cases at time $(t-1)$ from the previous day. Ideally, the infected population should be a dynamic group with newly infected ones in and recovered ones out, however, currently, daily data recording does not track the recovery time for each infected case. Therefore, only

the newly confirmed cases are included but recovered cases are not excluded in Y_t. The effect of recovery time on mortality and recovery rates is highlighted in Bhapkar et al. [58].

The reported numbers of confirmed COVID-19 cases are collated from an open-source database (https://ourworldindata.org/coronavirus-data). This study analyzes the computed G_t values for the period ranging between 22 Jan 2020 and 10 May 2020, as illustrated in Fig. 5.

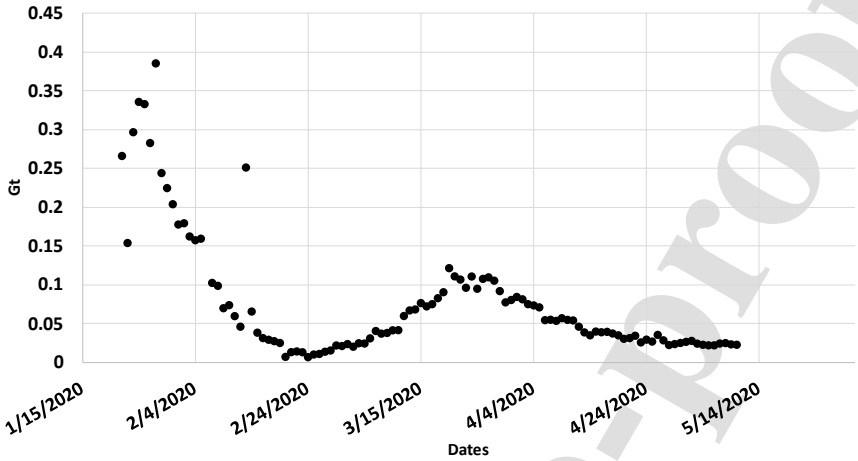


Fig. 5. Temporal variations of G_t between 22 Jan 2020 and 10 May 2020

Modelling the proposed G_t is performed using three unique scenarios, termed *Scenarios A to C*, with a defined number of multi-time steps based upon historical records, as measured in days. In all scenarios, the forecasting step is carried out with an additional fixed lead-time of 1 day, atop the defined number of multi-time steps for the respective scenario. For example, as generically exemplified in Fig. 6, to model and forecast the G_t parameter on 26 Jan 2020 with 3 days of multi-time steps and 1 lead day, the historical data for the period between 22 Jan 2020 and 24 Jan 2020 are used for the modelling step. Three customized DNNs are used to predict G_t, as shown in Fig. 7. Their hyperparameters are listed in Table 1. The exact descriptions of *Scenarios A to C* are given in the following:

- ***Scenario A*:** The G parameter (G_t), in its current state of time, is modeled directly as a function of 3 days, 5 days, 7 days, and 9 days of multi-time steps for the historical predicted sentiments as defined in Eq. (19), with a fixed lead-time of 1 day. The DNN design to model *Scenario A* is illustrated in Fig. 7(a).
- ***Scenario B*:** The G parameter (G_t), in its current state of time, is modeled directly as a function of 3 days, 5 days, 7 days, and 9 days of multi-time steps for the historical predicted sentiments and G_t for the same historical period as defined in Eq. (20), with a fixed lead-time of 1 day. The DNN design to model *Scenario B* is illustrated in Fig. 7(b).
- ***Scenario C*:** Built upon the same conditions as that of *Scenario B*, with the exception that the historical G_t values for the defined multi-time steps are assimilated or fused into selected hidden layers of the DNN model, the DNN design to model *Scenario C* is illustrated in Fig. 7(c).

$$G_t = f\left(X_{1,t-N-2,}X_{1,t-N-1,}\dots,X_{1,t-2,}\dots,X_{M,t-N-2,}X_{M,t-N-1,}\dots,X_{M,t-2}\right) \quad (19)$$

$$G_t = f\left(X_{1,t-N-2,}\dots,X_{1,t-2,}\dots,X_{M,t-N-2,}\dots,X_{M,t-2},G_{t-N-2,}\dots,G_{t-2}\right) \quad (20)$$

where X represents the predicted sentiments from the trained classifier, M the total number of features including all the unique predicted sentiments and the number of Tweets, and N the value of multi-time steps for the historical records.

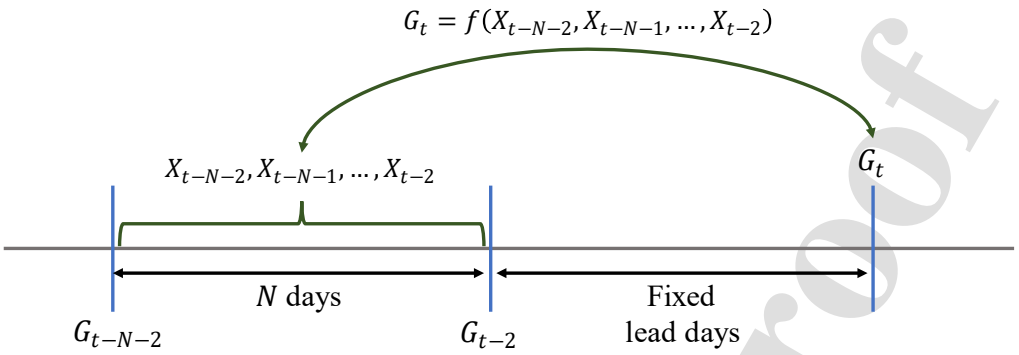
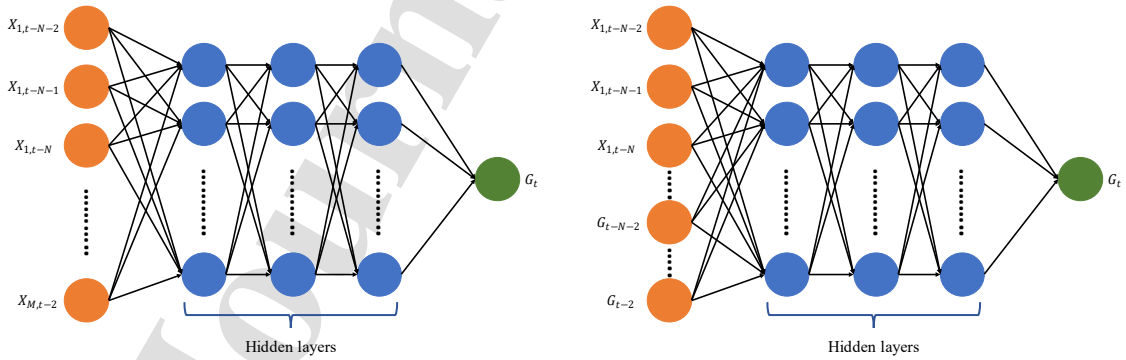
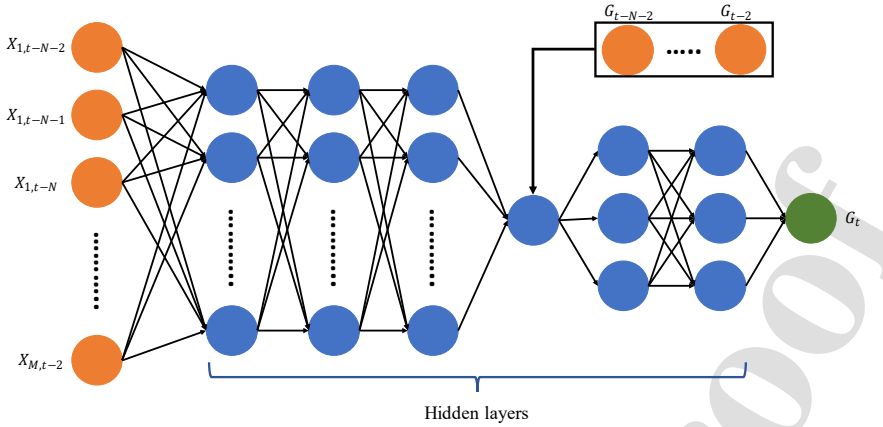


Fig. 6. Example for modelling and forecasting G_t parameter

Table 1. Summary of hyperparameter values for training DNNs in *Scenarios A-C*

Hyper-parameters	***Scenario A***	***Scenario B***	***Scenario C***
No. of neurons in the input layer	M × N	(M+1) × N	M × N
No. of neurons in hidden layer 1	int((M × N)/2)	int((M × N)/2)	int((M × N)/2)
No. of neurons in hidden layer 2	int((M × N)/3)	int((M × N)/3)	int((M × N)/3)
No. of neurons in hidden layer 3	int((M × N)/4)	int((M × N)/4)	int((M × N)/4)
No. of neurons in hidden layer 4	Nil	Nil	1 + N
No. of neurons in hidden layer 5	Nil	Nil	3
No. of neurons in hidden layer 6	Nil	Nil	3
No. of neurons in the output layer		1	
No. of lead days		1, 3, 5, 7, 9	
Batch Size		4, 8, 16	
Number of Epochs		500	
Learning rate		0.0001	
Activation function		Exponential Linear Unit (ELU)	
Optimization function		Adam	
Key cost function		Mean Squared Error (MSE)	





(c)

Fig. 7. DNN design for *Scenarios A to C* to model and forecast G_t parameter: (a) *Scenario A*; (b) *Scenario B*; and (c) *Scenario C*

3.6. Regressor performance evaluation

In all proposed scenarios (*Scenarios A to C*), evaluation of the respective models, during their testing phase, is performed with the following metrics, namely: (i) mean squared error (MSE) in Eq. (21); (ii) root mean squared error (RMSE) in Eq. (22); (iii) mean absolute percentage error (MAPE) in Eq. (23). MSE is selected as the key cost function for the model training step (see Table 2) to minimize the error difference between the measured and predicted G values, while RMSE and MAPE are also computed at the same time for a comprehensive analysis.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(G_{p,i} - G_{m,i}\right)^2 \tag{21}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(G_{p,i} - G_{m,i}\right)^2} \tag{22}$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{G_{p,i} - G_{m,i}}{G_{m,i}} \times 100\% \tag{23}$$

where N is the number of data samples being analyzed, $G_{p,i}$ the predicted G value at a specific day $(t = i)$, and $G_{m,i}$ the recorded G value on a specific day $(t = i)$.

4. Experimental studies

4.1. Background

Previous research studies on emotional response classification analyses encompass multiple text formats such as news headlines, blogs, Facebook dialogues, and Tweets [35]. Tweets differ from other literature materials in their text length, where the maximum number of characters for a single Twitter post is limited to 280 in quantity. Besides, misspellings and slang are commonly found in Tweets as compared to formal documents, such as reviews. This study selects one open-source emotional

responses dataset with known sentiment labels and another open-source unlabeled dataset as the corpora to train the deep learning word embedding model. The predictions from the trained word embedding model serve as the input features into the proposed classification model. The trained classification model is subsequently used to analyze new unlabeled emotional response data concerning the target COVID-19 topic. Several datasets with labels are evaluated separately using the proposed modular model for two purposes: (i) to investigate the performance of each modular model for selecting the best combination of modules as illustrated in Fig. 1; (ii) to evaluate the labeling quality of the extracted open-source dataset for selecting the specific dataset having the least volume of 'noise' (non-English words, punctuations, etc.), which extracted knowledge will be transferred to the new dataset. The training and target datasets are described in the following.

Sentiment classification has attracted attention from interdisciplinary research groups including computer science, psychology, and social science. Two fundamental viewpoints coexist in emotional response classification: (i) emotions are fundamentally distinct constructs; (ii) emotions can be characterized by feature dimensions. These two aspects correspond to two sets of theories dominating the discussion of emotions, namely: (a) discrete emotional theories; (b) dimensional emotional theories. The former considers a limited number of emotions, each with its characteristics, while the latter quantifies a specific emotion via dimensions. The two most important dimensions are emotional valence, indicating a positive or negative degree, and emotional arousal as indicative of emotional intensity.

In existing text sentiment datasets, each sentence is labeled with a specific emotional word. To unify the labels for automatic sentiment classification, this study considers the sentiment labels from the dimensional perspective. For simplicity, only the valence dimension is considered in this study. Valence is considered as two independent dimensions rather than a bipolar continuum [59]. This includes the situation where emotional experience is both positive and negative at the same time, which is named ambivalent. Taking binary values of positive and negative, valence can be classified into four categories: purely positive, purely negative, neutral, and ambivalent, as shown in Table 2.

Table 2 highlights that the two components of valence, "positive" and "negative", are independent rather than being the two ends of one scalar. For example, to judge the valence of a given text, if it is considered as the absence of negative emotion and the existence of positive emotion, it will be classified as positive.

There are two open-source datasets used for model training, namely: (i) IndianCovid19 and (ii) Covid19Tweets. Specifically, the IndianCovid19 dataset consists of around 3k Tweets from India on the topic of COVID-19 and lockdown. The Tweets have been collected between the dates 23rd March 2020 and 15th July 2020. The texts have been labeled into four categories including "fear, sad, anger, and joy". The dataset can be downloaded from Kaggle [60]. Covid19Tweets consists of around 45k Tweets from multiple regions on the topic of COVID-19. The Tweets are manually tagged with five labels "extremely positive, positive, neutral, negative, extremely negative". The dataset can be downloaded from Kaggle [61].

GlobalCOVID-19 is used as the target dataset. This dataset is hydrated by day according to the known Tweets ID, which can be downloaded from GitHub [62]. A total of 110 days of Tweets were

rehydrated from January 22nd, 2020 to May 10th, 2020 to analyze the global public sentiments towards the evolving pandemic situation. The daily number of Tweets varies from hundreds to millions in quantity as COVID-19 persists in the global community since January 2020.

The labels of the sentiment datasets are grouped into the same valence classification criteria, which considers the two components in the valence dimension as independent dimensions rather than two ends of the polarity. Thus, the positive and negative dimensions can be combined into four categories: neutral, positive, negative, and ambivalent. The re-mapping rules from the original labels to the unified labels are shown in Table 3.

Table 2. Valence categories that consider positive and negative as two independent dimensions.

Emotion dimension		Positive	
		0	1
Negative	0	Neutral	Positive
	1	Negative	Ambivalent

Table 3. Re-mapping of labels to the four categories of sentiment valence.

Dataset	Valence categories			
	Neutral	Positive	Negative	Ambivalent
IndianCovid19	-	Joy	Sad, fear, anger	-
Covid19Tweets	Neutral	Positive, extremely positive	Negative, extremely negative	-

Note: "-" denotes an empty class.

4.2. Hyperparameter tuning and training details

There are two components to the proposed NLP architecture which require hyperparameter tuning to achieve the desired level of accuracy in the coupled word embedding and classification models. The word embedding model serves as a feature extraction tool, while the classification model is built upon an ANN deep learning architecture.

"Size" and "window" are the two most important parameters for the CBOW and Skip-gram models. "Size" refers to the word vector dimensionality, while "Window" refers to the context window size. For example, with a window size of L, the L preceding words and L succeeding are used to predict the center word in the CBOW model and to be predicted given the center word in the Skip-gram model. Generally, a large dense vector dimension means a higher dimension of features and is expected to improve classification accuracy. While the increase in accuracy can plateau off when the word vector dimension goes over a certain value, a slower increase in accuracy can be observed for a large dense vector dimension when compared to an initial rapid increase in the accuracy score when the dense vector dimension is small. Similar patterns are observed when using CBOW and Skip-gram models. Hence, a proper word vector size must be selected to balance the model's accuracy performance and its resulting vector size.

The number of layers regardless of their types (fully connected layers, CNN layers, etc.) in the built ANN classifiers including MLP, CNN, and RNN, are kept consistent in the experimental runs.

The only difference between the three architectures is the type of hidden layers that follow the input layer as this study intends to compare the varying effects of dense layer, CNN layer, and RNN on the model's resulting classification accuracy. Figures 2, 3, and 4 respectively illustrate the different example designs of the deep learning architectures for MLP, CNN, and RNN models. The number of neurons in the output layer is kept consistent in all proposed architectures, while the number of neurons in the different dense layers is adjusted accordingly to achieve the desired level of accuracy performance. As a rule of thumb, the multiple deep learning classifiers are compiled using a common batch size value of 256 and using an Adam optimizer to minimize the in-built cost function. The value of epochs for MLP is set to 100 and model training for CNN and RNN is terminated at a relatively low number of epochs to avoid over-fitting of the respective networks.

4.3. Sentiment evolution analysis

As discussed previously, two sets of Tweet sentiment data are used to test the proposed modular architecture for text sentiment classification. IndianCovid19 and Covid19Tweets datasets provide Tweets on COVID-19 specifically. IndianCovid19 dataset contains around 3k Tweets which are considered relatively small in data quantity for NLP analysis. The Covid19Tweets dataset collects Tweets from multiple regions and is preferred due to its larger data quantity of around 45k. It is, however, worth noting that these selected datasets are still considered relatively small in data quantity. They are chosen due to their relevance and availability. The classification results obtained using the different combinations of modules (Fig. 1) are shown in Table 4 and Table 5, respectively, for the different datasets selected. Due to its direct relevance and labeling quality, Covid19Tweets is leveraged as the knowledge base in the transfer learning process of analyzing the evolution of the general public's sentiments from the much larger target dataset, Global COVID-19. The latter is collected by day and its size quantity is within the range of millions from March 2020 to May 2020. The execution times for compiling the different classifiers trained upon the Covid19Tweets dataset are listed in Table 6, while training curves (loss vs epochs, accuracy vs epochs) on the Covid19Tweets dataset are shown in Figs. 8-10. Stacked line plots of the daily number of Tweets classified into the four above-mentioned categories are shown in Fig. 11, while the percentages of the four sentiments and their evolution along time are shown in Fig. 12. The daily Tweets on COVID-19 are grouped by month and the percentages of Tweets in the four classes per month are shown in Fig. 13. Class-wise word cloud for Tweets on a specific day is shown in Fig. 14. The performance of the proposed architecture and the mined public's sentiments towards the pandemic situation is elaborated in detail in the following.

The best module combination for text sentiment classification is Skip-gram-Concatenation-RNN and the simplest architecture with relatively good performance is Skip-gram-Sum-MLP. The language model Skip-gram performs better than CBOW in solving the sentiment classification task, as the classification results on the datasetCovid19Tweets using Skip-gram (bottom half of Table 5) are always superior in comparison to the ones using CBOW (top half of Table 5). It should be noted that the derived results using the dataset of IndianCovid19 do not provide much insight or information for most of the modules. This could be due to the extremely small data size, which is one-tenth of the total number of Tweets in the other two small datasets used in this study. Conceptually, both CBOW and

Skip-gram function by minimizing the negative conditional occurrence probability. As described earlier, the main difference between Skip-gram and CBOW lies in their respective approaches to compute the word vectors of neighboring words around the center word. The obtained classification results generally show that the multiplication operator in the probability is better than the average operator on word vectors from the two models.

Among the sentence representation methods, the simple "Sum" operation is better than the "TF-IDF weighted sum". This may be caused by the low quality of learned/trained weightage values for the different words derived from the TF-IDF model. In terms of selecting the specific ANN classifier, the RNN classifier achieves the highest level of accuracy on all the experimental datasets. However, CNN and RNN are more vulnerable to over-fitting problems and it generally takes much longer to train those models as compared to the MLP model. As shown in Table 6, the execution time for compiling CNN and RNN models is around 8 and 14 times more than that of the MLP model, respectively. Referring to Fig. 9, overfitting occurs at around the limiting epoch value of 20. For RNN with small units, such as 20 in the current analysis, smoothed training curves can be obtained as shown in Fig. 10. Validation curves follow training curves, which indicates effective training. With the addition of more units in the LSTM layer, the RNN network requires a smaller number of epochs to attain the same level of accuracy performance and avoid overfitting. In comparison, the MLP model can be fine-tuned easily to achieve the best resulting accuracy and to achieve stable performance with all experimental datasets as shown in Fig. 8. Thus, the combination Skip_gram-Sum-MLP is selected to analyze the evolution of public sentiments towards COVID-19 to attain a balance between the model's resulting accuracy and its model training time.

Several trends in the evolution of public sentiments towards COVID-19 can be observed from this study, as elaborated below. (i) Public attention towards COVID-19 generally increased dramatically from late March 2020 to May 2020 (Fig. 11). As shown in Fig. 11, the daily number of Tweets peaked at millions in several waves whereas the volume after mid-April 2022 was stably higher than before. It indicates that before mid-April 2022, there was fluctuating attention towards COVID-19, and after mid-April 2022, the public remained highly interested in COVID-19. (ii) As the pandemic evolves, public sentiments shifted from neutral to polarity around late February 2020 (Fig. 12 (a)-(c)). Fig. 12 shows the raw percentage of the four sentiments and their 7-day rolling mean with SD. The indicators are normalized to show the distribution of sentiments over the four categories, which is irrelevant to the total Tweet volume. The rolling mean operation generally smooths the time series data to expose the trend. As shown in Fig. 10, during the period from Jan 2020 to May 2020, the quantity of neutral Tweets reduces from more than 40% to around 30% in absolute percentage values. Concurrently, an increasing trend can be observed for both percentages of negative and positive Tweets on the COVID-19 topic. Besides, negative sentiment exhibited three waves while positive sentiment decreased first, then increased, and stay flat. The statistical results are reasonable as the public's knowledge of COVID-19 accumulates with time and is most likely to develop their own opinions and understanding of COVID-19 as compared to a blank state in the initial stage. (iii) Even though both quantity percentages of negative and positive sentiments increased over time, the volume of negative sentiments was generally more dominant than that of the positive sentiments with an exceedance of

around 10% for the former emotion (Fig. 12 (b)-(c)). (iv) The public displayed dynamic emotions during the current pandemic (Fig. 12). The fluctuation and evolution of the public's sentiments on a global scale are in accord with the recognized trend of large vibrations at the early stage in other COVID-19 studies [63,64]. The sentiment coined as "ambivalent" can be difficult to model at this stage due to the lack of labeled data samples under the class "ambivalent". Hence, there are no ambivalent samples in the labeled dataset and the percentage of ambivalent Tweets can be considered an unknown class.

Box plot by month of the percentages of the Tweets in each category (Fig. 13) highlights the monthly trend. In detail, percentages of neutral Tweets decreased steadily till April 2020 and slightly increased in May 2020 (Fig. 13 (a)). Kind of reversely, the percentage of negative Tweets increased from Jan 2020 to Apr 2020 and decreased in May 2020 (Fig. 13 (b)). A more vibrating trend exhibits for the percentages of positive Tweets (Fig. 13(c)). It has two troughs (February 2020 and April 2020) and one peak (March 2020). A word cloud of the tweet contents on 23 March 2020, aggregated by their respective classes, is shown in Fig. 14 for illustration. The neutral Tweets contain general descriptions of COVID-19, while the negative Tweets contain negative words such as "sentenced" and "crisis". The positive Tweets, on the other hand, contain positive words such as "want" and "drug". The ambivalent, i.e., unknown class, Tweets contain both topics about "coronavirus" and "senate". Interestingly, the words related to "senate" also appeared at high frequency in the pool of negative and ambivalent Tweets such as "sentenced", "contracted", and "pass".

Table 4. Valence classification results of the IndianCovid19 dataset.

Model modules setting								Performance (Accuracy)		
Word embedding		Sentence representation			Classifier			Train (64%)	Val (16%)	Test (20%)
CBOW	Skip-gram	Sum	TF-IDF weighted sum	Concatenation	MLP	CNN	RNN			
✓		✓			✓			0.7683	0.7556	0.7557
✓			✓		✓			0.7648	0.7758	0.7670
✓				✓		✓		0.7638	0.7697	0.7670
✓				✓			✓	0.7623	0.7717	0.7670
	✓	✓			✓			**0.8098**	**0.7879**	**0.8042**
	✓		✓		✓			0.7693	0.7778	0.7670
	✓			✓		✓		0.7623	0.7717	0.7670
	✓			✓			✓	0.7623	0.7717	0.7670

Note: A word vector dimension of 72 and a window size of 7 are used for the word embedding training. The ANN classifiers' specifications are listed below. MLP: 72_24_8_4, epochs=100; CNN: (CNN filters=36, kernel_size=3)_16_4, epochs=20; RNN: (LSTM units=20)_16_4, epochs=100. The tick symbol "✓" represents that the module is selected, and the values in bold represent the best results.

Table 5. Valence classification results of the Covid19Tweets dataset.

Model modules setting								Performance (Accuracy)		
Word embedding		Sentence representation			Classifier			Train (64%)	Val (16%)	Test (20%)
CBOW	Skip-gram	Sum	TF-IDF weighted sum	Concatenation	MLP	CNN	RNN			
✓		✓			✓			0.5850	0.5761	0.5727
✓			✓		✓			0.5459	0.5448	0.5465
✓				✓		✓		0.5894	0.5683	0.5756
✓				✓			✓	0.6300	0.5989	0.5999
	✓	✓			✓			0.6420	0.6159	0.6173
	✓		✓		✓			0.5807	0.5658	0.5610
	✓			✓		✓		0.6529	0.6167	0.6206
	✓			✓			✓	**0.6929**	**0.6523**	**0.6502**

Note: A word vector dimension of 72 and a window size of 7 are used for the word embedding training. The ANN classifiers' specifications are listed below. MLP: 72_24_8_4, epochs=100; CNN: (CNN filters=18, kernel_size=3)_8_4, epochs=20; RNN: (LSTM units=20)_16_4, epochs=100. The tick symbol "✓" represents that the module is selected, and the values in bold represent the best results.

Table 6. Comparison of execution time in compiling the ANN classifiers on the Covid19Tweets dataset. The classifiers have the same structures specified in Table 5 and are trained on 100 epochs.

Classifier	Execution time (s)
MLP	12.99
CNN	102.97
RNN	169.52

Note: CPU specification: Intel(R) Xeon(R) W-2123 CPU @ 3.60GHz.

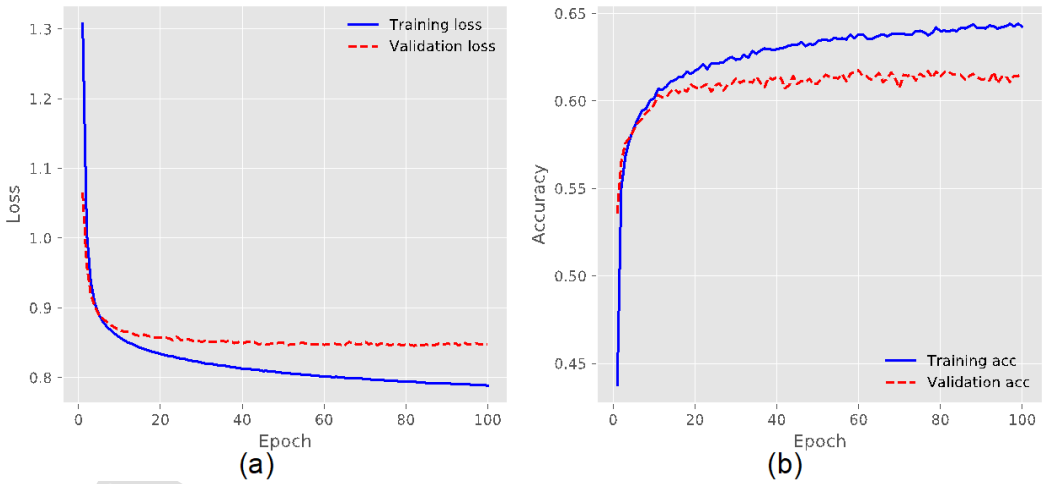


Fig. 8. Training curves of the classifier MLP on the Covid19Tweets dataset: (a) Loss; (b) Accuracy.

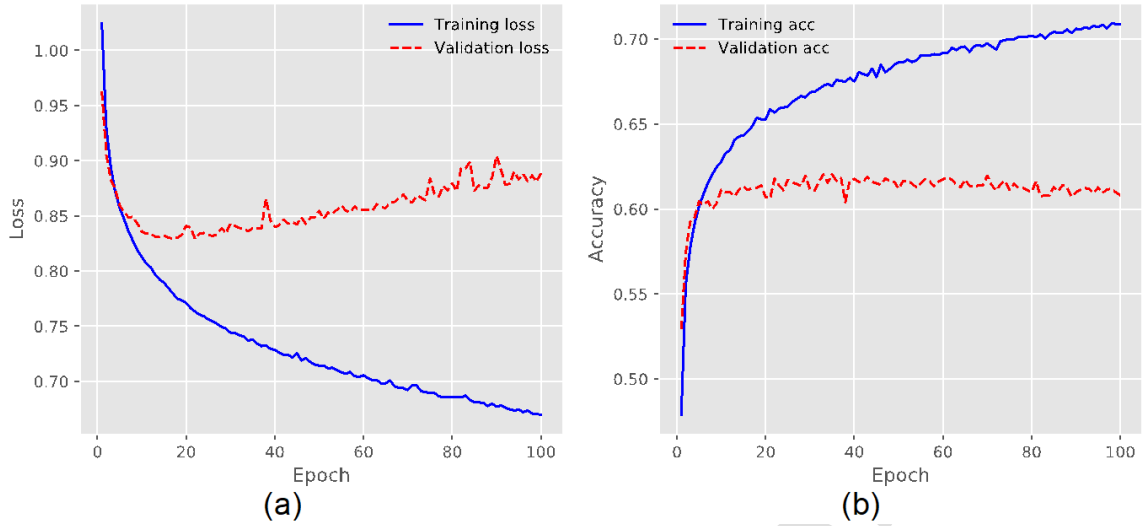


Fig. 9. Training curves of the classifier CNN on the Covid19Tweets dataset: (a) Loss; (b) Accuracy.

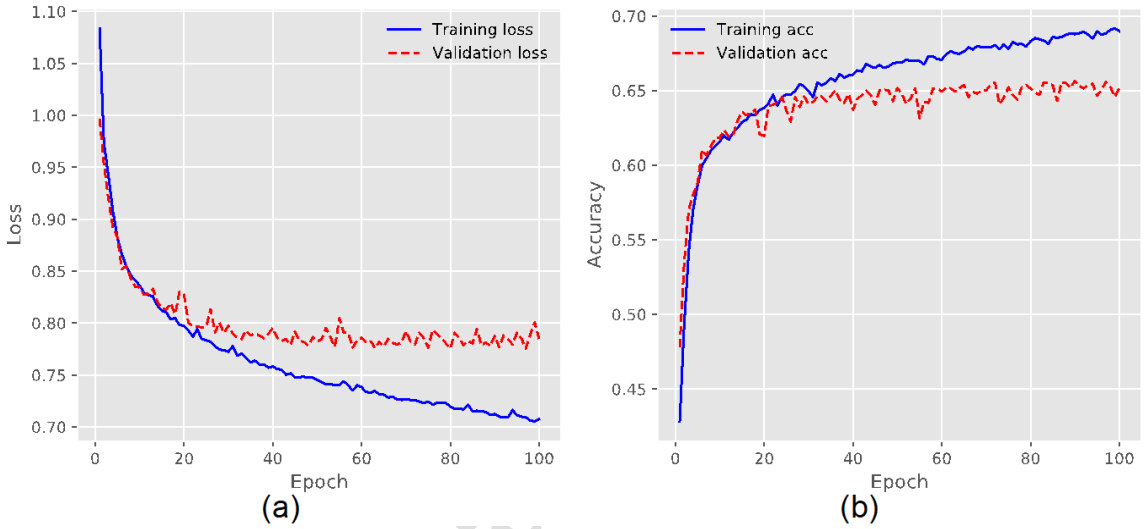


Fig. 10. Training curves of the classifier RNN on the Covid19Tweets dataset: (a) Loss; (b) Accuracy.

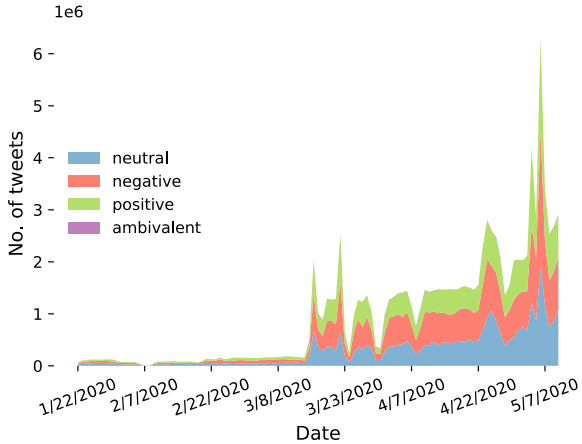
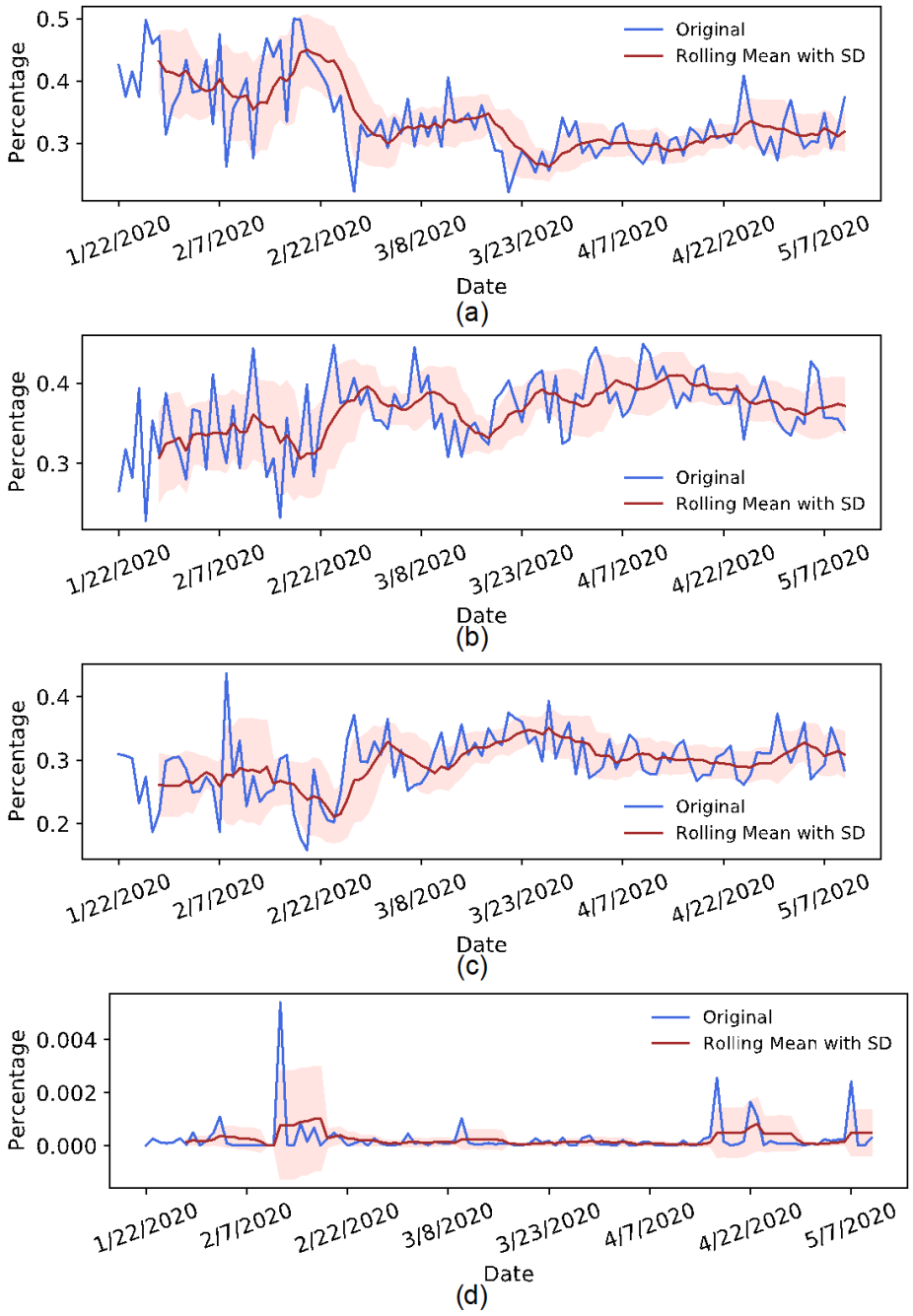


Fig. 11. Stacked line plot of daily no. of Tweets in the four categories.



676

677 **Fig. 12.** Percentages of Tweets and their 7-day rolling mean with standard deviation (SD), indicated
678 by shadow, in the sentiment categories: (a) Neutral; (b) Negative; (c) Positive; (d) Ambivalent.

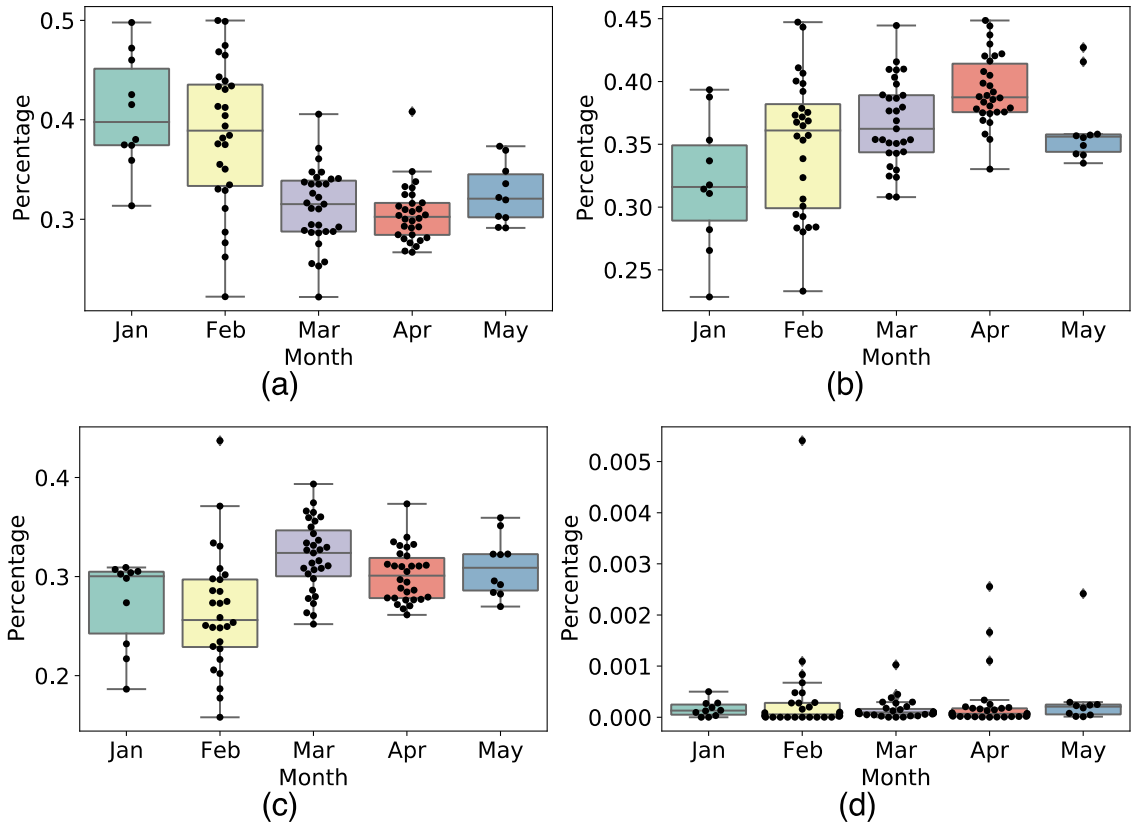


Fig. 13. Boxplot of the percentage of Tweets in the four categories grouped by month: (a) Neutral; (b) Negative; (c) Positive; (d) Ambivalent.

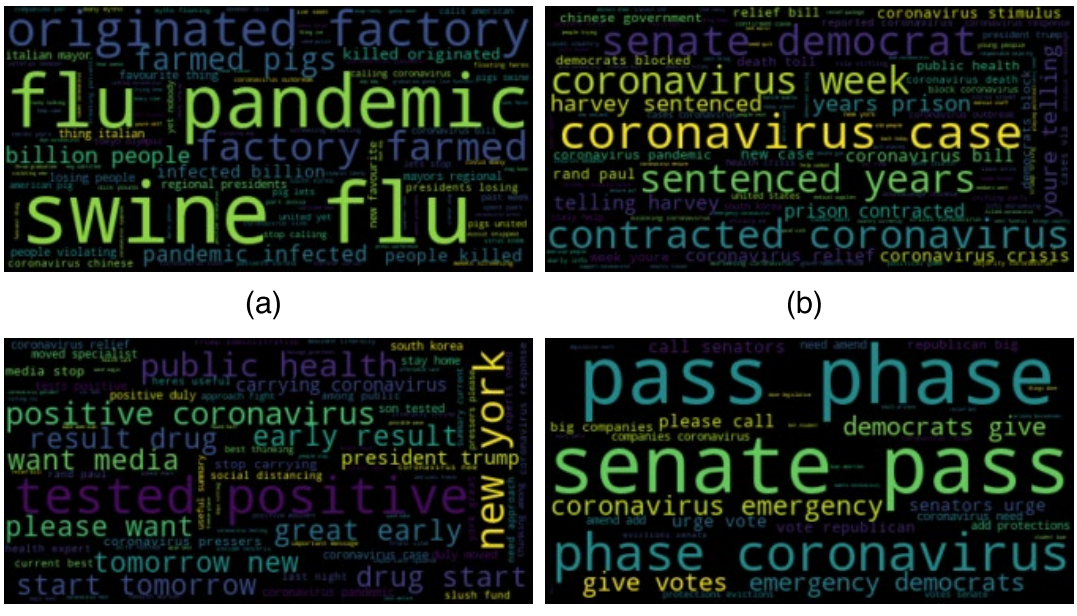


Fig. 14. Word cloud of the Tweet text on 23rd Mar 2020 in the class (a) Neutral; (b) Negative; (c) Positive; (d) Ambivalent.

4.4. Predictions of G_t (*Scenarios A-C*)

As discussed earlier, *Scenarios A-C* (Fig. 7) are investigated with varying batch sizes (4, 8, 16) and multi-time steps (1, 3, 5, 7, 9 days) to model and forecast G_t in its current state of time with a fixed lead-time of 1 day. Figs. 15-17 depicts the MSE cost function versus the number of epochs for the model's training and validation steps in corresponding scenarios, while Table 7 shows the testing results (i.e., RMSE, MAE, and MAPE scores) for each of the model configurations. Note that the time-series dataset for G_t (Fig. 5) is split into 85% for model training and validation, and the remaining 15% for testing. No random shuffling is performed prior to the data split for the model training and validation phase. At this stage, the key findings are summarized below:

- The model's predictive capability generally improves from *Scenario A to C*, i.e., decreasing RMSE, MAE, and MAPE scores as shown in Table 7, due to the advantage provided when leveraging on the previous days of G_t values to model the same parameter in its current state of time. Relatively, the proposed DNN design in Fig. 7(c) provides an additional edge to the modelling step by assimilating/fusing the previous days of G_t values at the intermediate layers of the DNN model, where the corresponding values are concatenated with the transformed input features from the predicted emotional classes to set up a newly processed intermediate input layer to model the G_t parameter.
- For *Scenario C* (with data assimilation for G_t), the best model configuration, which provides the lowest possible MAPE score of 17%, is based upon the batch size of 4 and uses 1 day of multi-time steps for the data assimilation step. Comparison of the corresponding model predictions, using this configuration, with the respective monitored G_t values from the model's testing step are shown in Fig. 18. There is a reasonably good agreement between the predicted and monitored G_t values, which hence provides the possibility of using the same trained model configuration to undergo model re-training with additional datasets for the G_t parameter for near real-time predictions with a fixed lead-time of 1 day. For example, the current lowest MAPE score of 17% for the G_t prediction suggests that if the actual/monitored number of confirmed COVID-19 cases from the previous 2nd day (with a lead-time of 1 day) is 100,000, then the present model is likely to forecast the number of cases to range between 84,000 and 117,000. However, in extreme cases of a very large number of COVID-19 cases globally by day, then it becomes imperative to lower the current MAPE scores for decision-makers to better estimate the level of responses to handle any sudden spikes in COVID-19 cases.
- For *Scenarios A and B*, the use of larger multi-time steps generally improves the model's predictive capability during its testing step, as shown in Table 7, however, without achieving a MAPE score of less than 30% in any of the modeled cases in the respective scenarios. On the contrary, for *Scenario C* which involves the proposed data assimilation component, the use of a smaller number of multi-time steps results in a better model's predictive performance (see Table 7 and Fig. 16). At the same time, the results from *Scenario C* underlines strong volatility/fluctuations, i.e., low level of seasonality, in the G_t values, hence smaller number of multi-time steps (e.g., previous 1 day of data) can better encapsulate any sudden changes/variations in the monitored G_t values over time.

The empirical runtime of the proposed framework (Fig. 1) is provided. Table 8 shows the average computational time for the different components of the proposed framework. Note that for brevity, the average computational runtime for the different steps in Part II is averaged across the varying multi-time steps and batch sizes for each of *Scenarios A to C*. The total average runtime for the proposed model framework does not exceed 24 hours, hence enabling it to be used for the daily near real-time predictive analysis for predicting the G_t, in its current state of time, with a fixed lead-time of 1 day.

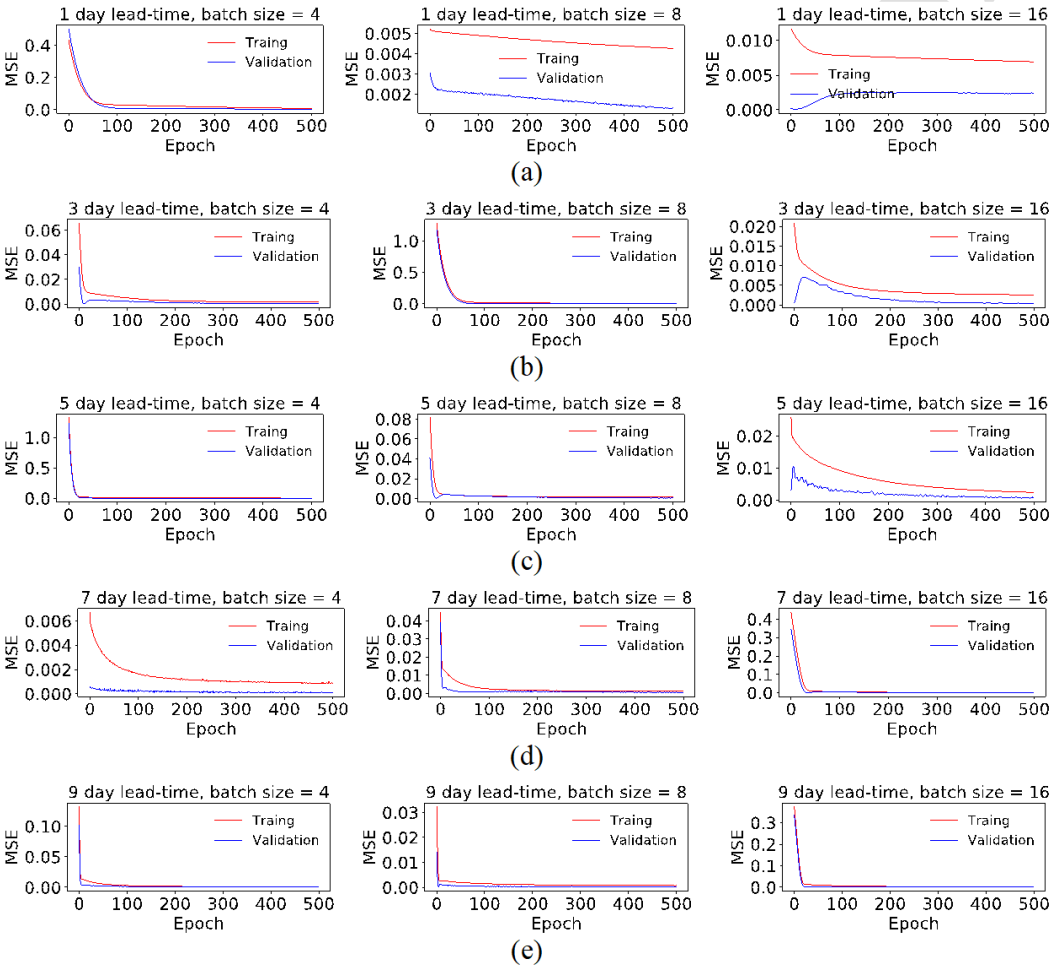


Fig. 15. Training and validation losses (*Scenario A*) for batch sizes of 4, 8, and 16 with a fixed total number of epochs of 500 at varying lead-times: (a) 1 day lead-time; (b) 3 days lead-time; (c) 5 days lead-time; (d) 7 days lead-time; and (e) 9 days lead-time.

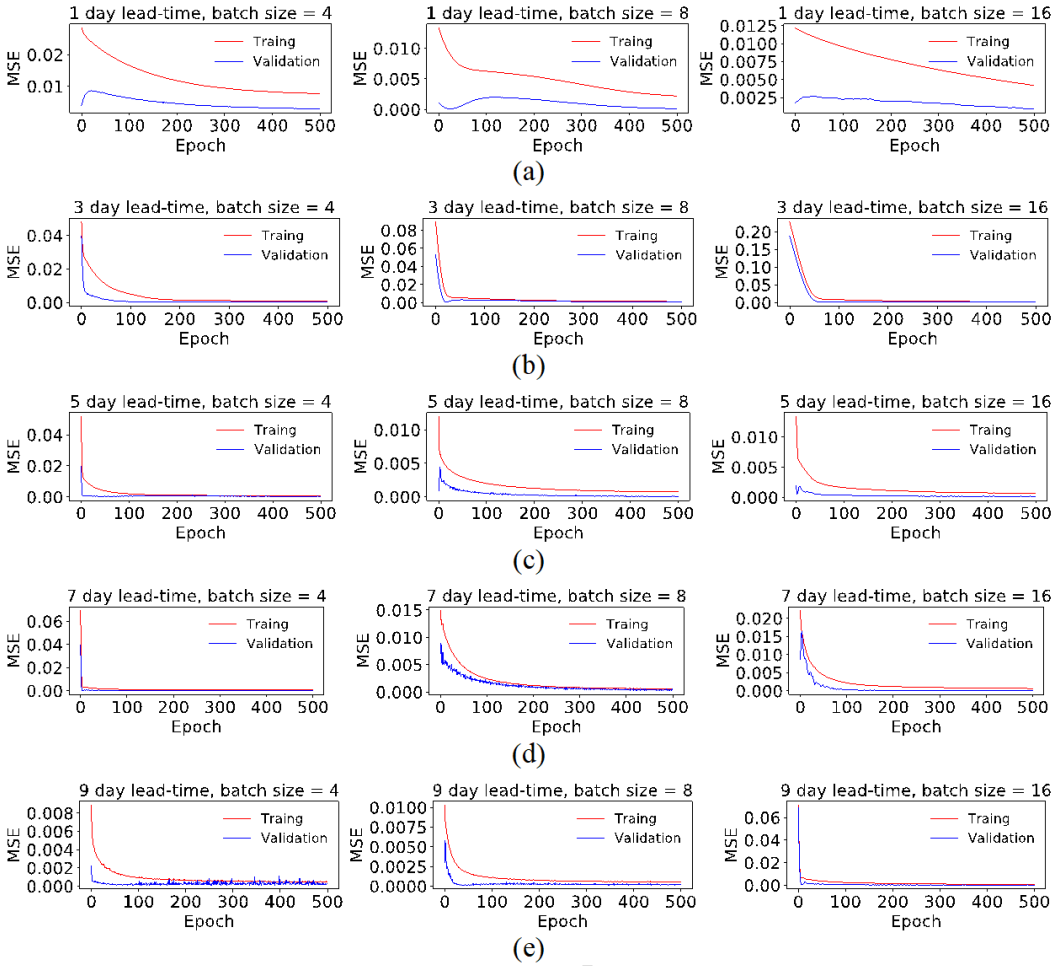


Fig. 16. Training and validation losses (*Scenario B*) for batch sizes of 4, 8, and 16 with a fixed total number of epochs of 500 at varying lead-times: (a) 1 day lead-time; (b) 3 days lead-time; (c) 5 days lead-time; (d) 7 days lead-time; and (e) 9 days lead-time.

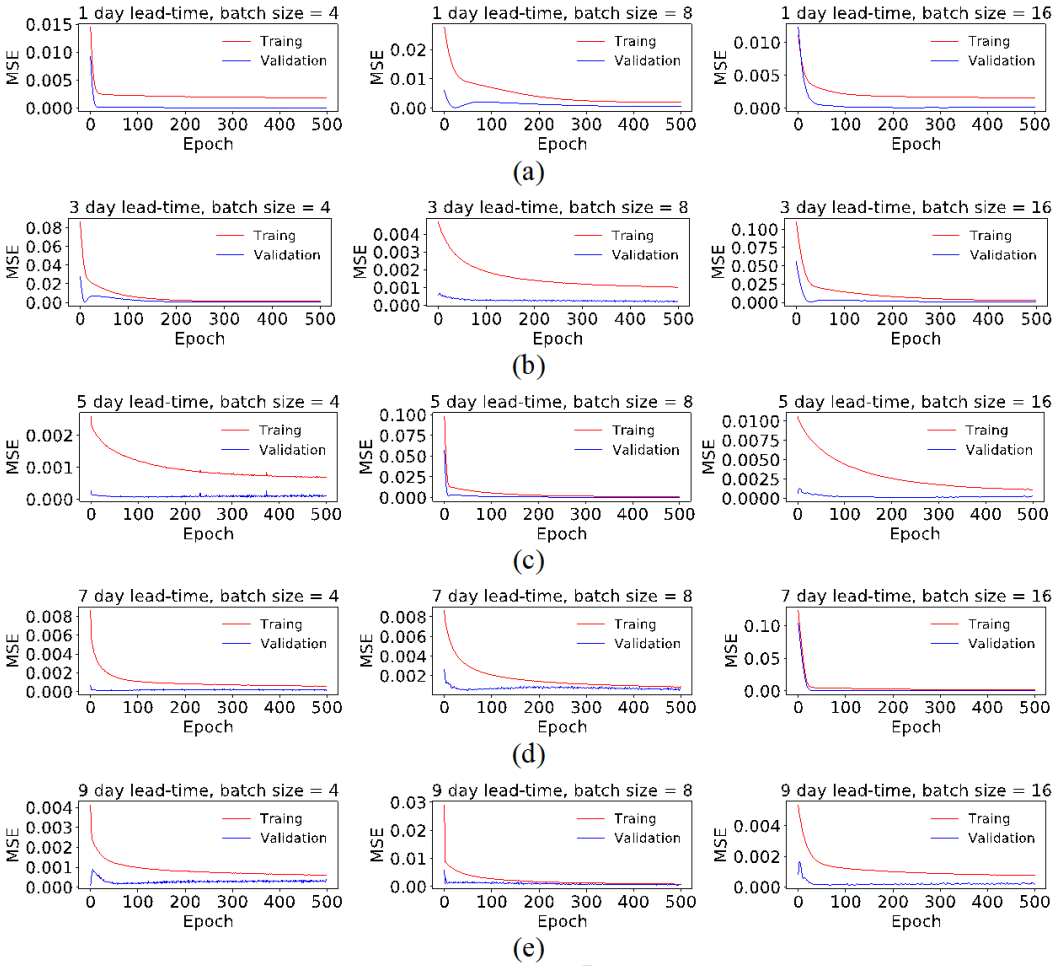


Fig. 17. Training and validation losses (*Scenario C*) for batch sizes of 4, 8, and 16 with a fixed total number of epochs of 500 at varying lead-times: (a) 1 day lead-time; (b) 3 days lead-time; (c) 5 days lead-time; (d) 7 days lead-time; and (e) 9 days lead-time.

744 **Table 7.** Summary of estimated RMSE, MAE, and MAPE scores for predicting G_t under *Scenarios A-*
745 *C* during the model's testing phase at varying multi-time steps and batch size.

Scenario	Multi-time steps (days)	Batch size	RMSE	MAE	MAPE
A	1	4	0.059	0.058	236.3%
	1	8	0.058	0.058	233.6%
	1	16	0.058	0.058	235.3%
	3	4	0.034	0.033	135.9%
	3	8	0.036	0.035	143.2%
	3	16	0.040	0.039	160.1%
	5	4	0.033	0.032	131.5%
	5	8	0.035	0.034	139.2%
	5	16	0.035	0.034	139.6%
	7	4	0.032	0.030	125.1%
	7	8	0.031	0.031	126.8%
	7	16	0.033	0.032	130.6%
	9	4	0.032	0.029	123.3%
	9	8	0.030	0.029	119.4%
	9	16	0.033	0.031	129.5%
B	1	4	0.032	0.031	126.5%
	1	8	0.042	0.042	172.0%
	1	16	0.044	0.044	178.8%
	3	4	0.014	0.013	55.6%
	3	8	0.015	0.014	57.6%
	3	16	0.018	0.017	68.9%
	5	4	0.017	0.015	61.3%
	5	8	0.016	0.014	59.1%
	5	16	0.016	0.015	62.0%
	7	4	0.018	0.015	63.3%
	7	8	0.018	0.017	68.3%
	7	16	0.018	0.017	69.9%
	9	4	0.022	0.020	84.0%
	9	8	0.020	0.019	77.2%
	9	16	0.019	0.017	72.6%
C	**1**	**4**	**0.003**	**0.004**	**17.0%***
	1	8	0.015	0.013	54.6%
	1	16	0.009	0.008	33.7%
	3	4	0.015	0.013	54.2%
	3	8	0.013	0.012	49.0%
	3	16	0.033	0.029	117.9%
	5	4	0.015	0.013	53.1%
	5	8	0.019	0.016	65.9%
	5	16	0.020	0.017	69.6%
	7	4	0.016	0.013	54.9%
	7	8	0.023	0.022	87.4%
	7	16	0.022	0.019	77.1%
	9	4	0.020	0.018	75.2%
	9	8	0.016	0.014	56.3%
	9	16	0.012	0.010	42.7%

746 **** the lowest MAPE score obtained from the best possible model configuration***

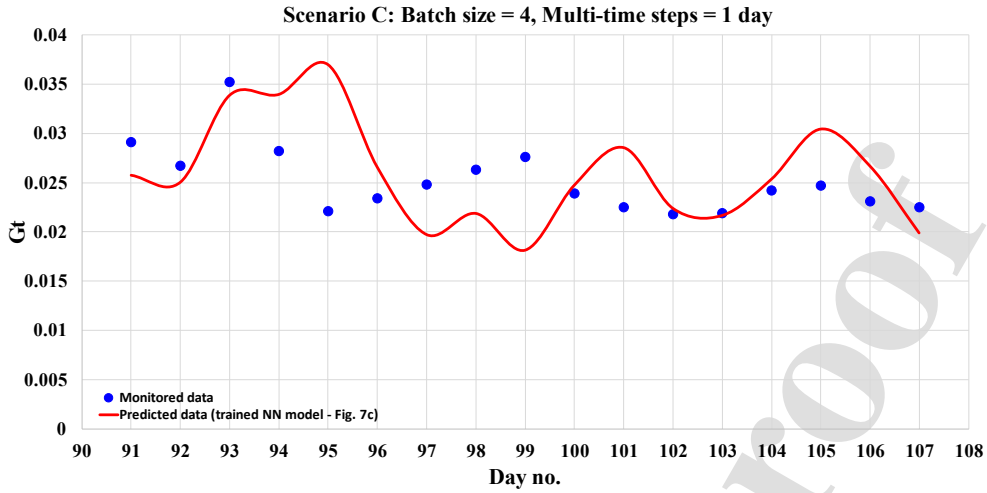


Fig. 18. Comparison between the predicted G_t values (using the best model configuration) and monitored G_t values.

Table 8. Summary of average computational runtime for proposed framework involving Part I and Part II.

Part	Step	Average runtime
I	Data Hydration (Tweet data ranging between 22 Jan 2020 & 10 May 2020)	6hrs
	Data Pre-Processing + Features Extractions	20mins
	MTV using MLP (100 epochs, model configuration from Table 5)	13.0s
	MTV using CNN (100 epochs, model configuration from Table 5)	103.0s
	MTV using RNN (100 epochs, model configuration from Table 5)	170.0s
II	MTV for Scenario A (500 epochs, model configuration Table 1)	47.5s
	MTV for Scenario B (500 epochs, model configuration Table 1)	48.0s
	MTV for Scenario C (500 epochs, model configuration Table 1)	50.0s
	Trained Model Restoration	20.0s
	Model Predictions in Near Real-Time	10.0s

* *MTV – Model Training & Validation*

In this study, distribution over the four sentiments induced in the early 109 days are tested and show promising performance (i.e., MAPE of 17% for *Scenario C*) in the prediction of G_t value. Testing results from *Scenarios A to C* indicate that historical sentimental responses towards COVID-19 can serve as an additional input, together with the historical COVID-19 records, to inform the near-future COVID-19 situations. Besides, sentiments contribute to the prediction of G_t values not as parallel inputs with historical G_t values but require some processing to get an abstract value as facilitated by the first part NN (Fig. 7).

In the past around 1000 days, there have been several COVID-19 waves, exhibiting recurrence patterns of surges in new cases followed by declines, as shown in Fig. 19(a). The study empirically investigates the effect of sentiments towards COVID-19 predictions in the early days. It does not

extend to a later period as (i) a deluge of Tweets mentioned COVID-19 as it became a common topic, which makes the retrieval of data (limited by the company twitter) and the processing extremely slow; and (ii) there is a selection problem because the later Tweets often mentioned COVID-19 casually rather than talking about it, however, in NLP, topic modelling remains an active research topic [65,66]. Besides the technical issues in obtaining sentiment indicators, generalization and adaptation of the proposed method in pandemic prediction concern several scales, such as adaptation across events [67], adaptation to different stages within an event, and adaptation to countries or cities [68,69]. The prediction based purely on case data is solved by transfer learning [67,68]. This study utilizes both sentiment data and case data. It does not investigate the long-term sentiment variations towards COVID-19. However, the importance of selecting the proper output variables is highlighted below for future practices.

Three indicators (i.e., total confirmed cases, new confirmed cases, and daily growth rate of confirmed cases) of COVID-19 situations since its inception to the most recent data are shown in Fig. 19. It is observed that the curve of total cases has a smooth line (Fig. 19(a)), which is hard to expose the waves. A relative measure called new cases can be obtained by the minus operator between the records of total cases corresponding to two consequent days. Its raw values and smoothed curves (Fig. 19(b)) consistently exhibit 5-6 waves, all of which happened in the middle or later period. Another relative measure termed G_t is obtained by minor and division operators (Eq. (18)). Its raw values and smoothed curves (Fig. 19(c)) expose the vibrating trend in the early stage, which is not depicted by the previous two measures, though the later curves are stabilized by the large total cases. Therefore, it is claimed that at different stages, various measures are required to expose the details of the pandemic evolution for disease control and prevention. For example, at the early stage G_t is a good measure to expose the day-to-day difference while later the measure, new cases, is a better one.

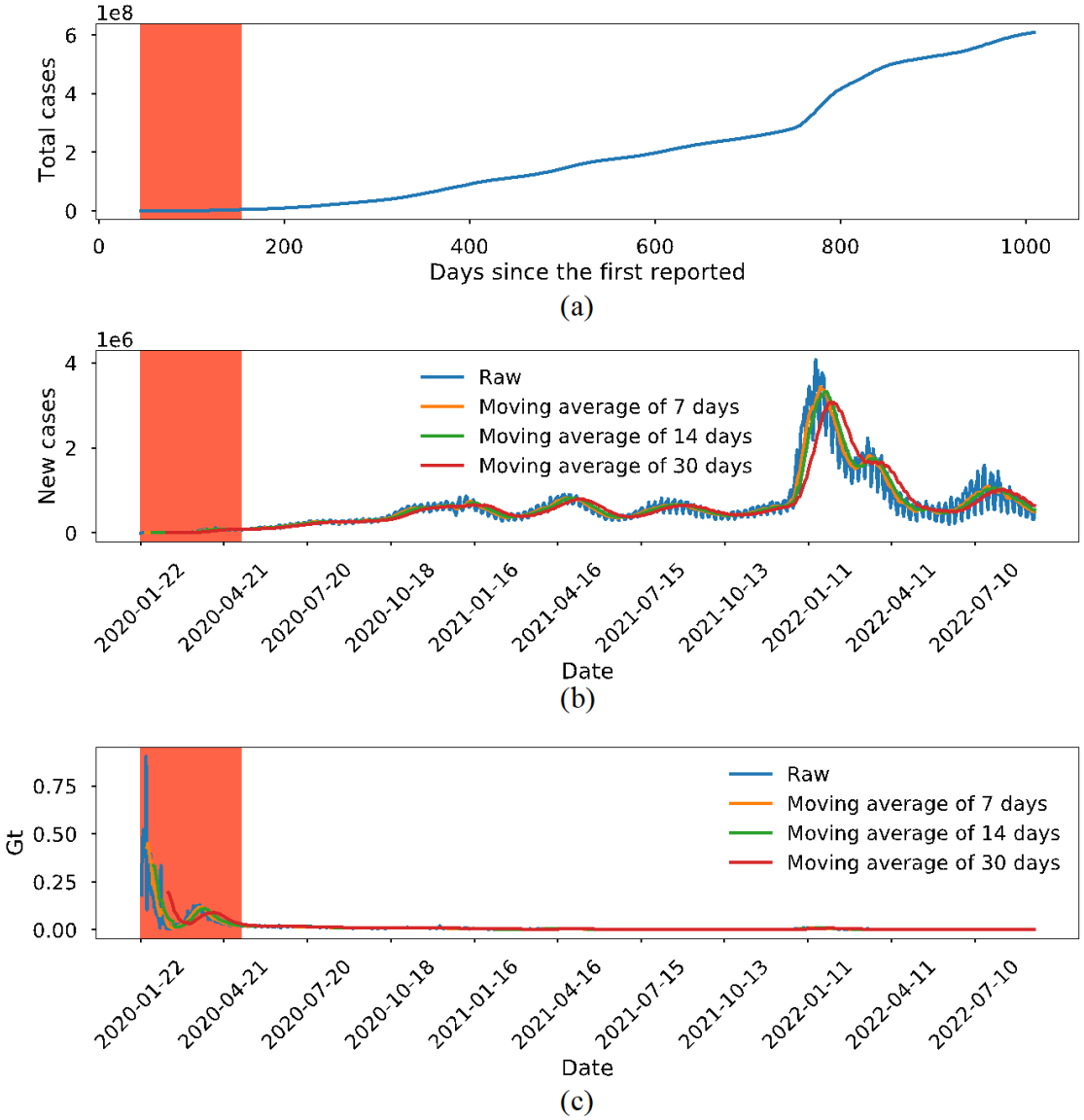


Fig. 19. Global data of COVID-19 evolution since the first reported case: (a) Total cases; (b) New cases; and (c) Daily growth rate of confirmed cases, i.e., the defined G_t values. Note: Highlighted region is the studied period when little of COVID-19 was known.

5. Conclusions and future works

This paper develops a modular deep learning framework for COVID-19-related text sentiments classification and its application in transfer learning to analyze the public sentiment towards a specific topic, followed by leveraging on the predicted sentiments to model and forecast the temporal evolution in the number of confirmed COVID-19 cases globally. The proposed language architecture is first trained and validated on open-source sentiment datasets, where the subsequent classification results on the testing datasets demonstrate that the proposed *Skip-gram-Concatenation-RNN* module combination provides the best predictive performance. At the same time, the alternative module

combination in *Skip-gram-Sum-MLP* also results in acceptable model performance, while also providing an additional advantage of the relatively simple model design which can reduce the total computational cost for model training and validation.

As the COVID-19 pandemic continues to evolve, the present classification results demonstrate regular patterns in the predicted sentiments. Overall, the results indicate that the general populations gradually exhibit positive or negative sentiments towards COVID-19, as compared to neutral responses towards the pandemic during the 1st two months of 2020 for the virus' inception. At around late February 2020, the percentages of neutral, negative, and positive Tweets also gradually changed from 40%, 34%, and 26% to 30%, 40%, and 30% respectively. Generally, the total amount of negative sentiments generated towards COVID-19 is greater than that of the positive sentiments by around 10%. The predicted sentiments (four classes in total) in time-series profiles, coupled with the increased rate in the total number of COVID-19 related Tweets, are subsequently leveraged as unique model input features to train, validate, and test DNNs to model and forecast the growth rate in the total number of COVID-19 cases globally, via a G parameter, for the period between 22 Jan 2020 and 10 May 2020 via multiple scenarios of data selections and fusions. By far, the best possible model configuration of batch-size hyperparameter value of 4 and multi-time steps of 1 day can train a prediction DNN model which produces an average mean absolute percentage error (MAPE) score of around 17.0% on the testing dataset for forecasting the proposed G parameter.

The limitations of this study are stated as follows. Firstly, on the technical issues of multi-class text sentiment classification, this study does not access the quality of the Tweets where in reality users vary in the capability and willingness to express their emotions in text. Secondly, this study proposes a logically complete sentiment valence classification system but remains limited in identifying the rich sentiment Tweets called "ambivalent" due to the lack of labeled Tweets. Finally, this study provides an assessment of the public sentiments towards COVID-19 but does not aggregate the sentiments by topics or geography. Accordingly, future work can add filters in the Tweets retrieval process to gain insights into specific contexts. Besides, unifying the labeling process (e.g., text sentiment labels) will benefit the identification of human sentiment expressed in text format. In summary, the proposed framework can be incorporated into pandemic monitoring and control for providing quantified indicators of public sentiments and pandemic situations.

Authorship contributions statement

Ying Wang: Formal analysis, Methodology, Visualization, Writing – original drift. **Alvin Wei Ze Chew:** Conceptualization, Data hydration, Writing – review and editing. **Limao Zhang:** Conceptualization, Supervision, Methodology, Writing – review and editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The Ministry of Education Tier 1 Grants, Singapore (No. 04MNP000279C120, No. 04MNP002126C120) and the Start-Up Grant at Nanyang Technological University, Singapore (No. 04INS000423C120) are acknowledged for their financial support of this research. The 1st is grateful to the SINGA scholarship, which supports the author's study and research in Nanyang Technological University (NTU), Singapore. The 2nd author is grateful to Microsoft Corporation for the AI for Health Covid-19 Azure Compute Grant of ID:00011000272, which has been instrumental to support the team's simulation runs (around 40h computing time) on Azure's NC12s V2 N-Series virtual machine (VM).

References

[1] H. Gusterson, COVID-19 Darwinism, Social Anthropology/Anthropologie sociale 28 (2) (2020) pp. 275-276, https://doi.org/10.1111/1469-8676.12880.

[2] A. Maiti, Q. Zhang, S. Sannigrahi, S. Pramanik, S. Chakraborti, A. Cerda, F. Pilla, Exploring spatiotemporal effects of the driving factors on COVID-19 incidences in the contiguous United States, Sustainable Cities and Society 68 (2021) p. 102784, https://doi.org/10.1016/j.scs.2021.102784.

[3] B. Li, Y. Peng, H. He, M. Wang, T. Feng, Built environment and early infection of COVID-19 in urban districts: A case study of Huangzhou, Sustainable Cities and Society 66 (2021) p. 102685, https://doi.org/10.1016/j.scs.2020.102685.

[4] J.M.L.P. Montesclaros, CO20030 | Beyond COVID-19: Global Priorities Against Future Contagion, (2020). https://www.rsis.edu.sg/rsis-publication/nts/beyond-covid-19-global-priorities-against-future-contagion/#.YywP9-xBzQ0 (accessed 22 September, 2022).

[5] S. Borra, N. Dey, Misinformation About COVID-19 and Confidential Information Leakage: Impacts on the Psychological Well-being of Indians, Current Psychiatry Research and Reviews 16 (4) (2021) pp. 283-287, https://doi.org/10.2174/2666082216999200917143247.

[6] G. Wu, X. Deng, B. Liu, Managing urban citizens' panic levels and preventive behaviours during COVID-19 with pandemic information released by social media, Cities 120 (2022) p. 103490, https://doi.org/10.1016/j.cities.2021.103490.

[7] A. Gorska, D. Dobija, G. Grossi, Z. Staniszewska, Getting through COVID-19 together: Understanding local governments' social media communication, Cities (2021) p. 103453, https://doi.org/10.1016/j.cities.2021.103453.

[8] X. Ye, S. Li, Q. Peng, Measuring interaction among cities in China: A geographical awareness approach with social media data, Cities 109 (2021), https://doi.org/10.1016/j.cities.2020.103041.

[9] R.I. Ogie, S. James, A. Moore, T. Dilworth, M. Amirghasemi, J. Whittaker, Social media use in disaster recovery: A systematic literature review, International Journal of Disaster Risk Reduction 70 (2022), https://doi.org/10.1016/j.ijdrr.2022.102783.

[10] S.J. Fong, N. Dey, J. Chaki, AI-Empowered Data Analytics for Coronavirus Epidemic Monitoring and Control, Artificial Intelligence for Coronavirus Outbreak, 2021, pp. 47-71.

[11] M. Gupta, R. Jain, S. Taneja, G. Chaudhary, M. Khari, E. Verdu, Real-time measurement of the uncertain epidemiological appearances of COVID-19 infections, Applied Soft Computing 101 (2021) p. 107039, https://doi.org/10.1016/j.asoc.2020.107039.

[12] J.H. Jones, A. Hazel, Z. Almquist, Transmission-dynamics models for the SARS Coronavirus-2, American Journal of Human Biology 32 (5) (2020) p. e23512, https://doi.org/10.1002/ajhb.23512.

[13] V. Srivastava, S. Srivastava, G. Chaudhary, F. Al-Turjman, A systematic approach for COVID-19 predictions and parameter estimation, Personal and Ubiquitous Computing (2020) pp. 1-13, https://doi.org/10.1007/s00779-020-01462-8.

[14] Y. Xiang, Y. Jia, L. Chen, L. Guo, B. Shu, E. Long, COVID-19 epidemic prediction and the impact of public health interventions: A review of COVID-19 epidemic models, Infectious Disease Modelling 6 (2021) pp. 324-342, https://doi.org/10.1016/j.idm.2021.01.001.

[15] A.H. Alamoodi, B.B. Zaidan, A.A. Zaidan, O.S. Albahri, K.I. Mohammed, R.Q. Malik, E.M. Almahdi, M.A. Chyad, Z. Tareq, A.S. Albahri, H. Hameed, M. Alaa, Sentiment analysis and its applications in fighting COVID-19 and infectious diseases: A systematic review, Expert Systems with Applications 167 (2021) p. 114155, https://doi.org/10.1016/j.eswa.2020.114155.

[16] R.S. Ogden, The passage of time during the UK Covid-19 lockdown, PLOS ONE 15 (7) (2020) p. e0235871, https://doi.org/10.1371/journal.pone.0235871.

[17] Z. Yao, J. Yang, J. Liu, M. Keith, C. Guan, Comparing tweet sentiments in megacities using machine learning techniques: In the midst of COVID-19, Cities 116 (2021), https://doi.org/10.1016/j.cities.2021.103273.

[18] K. Chakraborty, S. Bhatia, S. Bhattacharyya, J. Platos, R. Bag, A.E. Hassanien, Sentiment Analysis of COVID-19 tweets by Deep Learning Classifiers—A study to show how popularity is affecting accuracy in social media, Applied Soft Computing 97 (2020) p. 106754, https://doi.org/10.1016/j.asoc.2020.106754.

[19] P.S. Desai, News Sentiment Informed Time-series Analyzing AI (SITALA) to curb the spread of COVID-19 in Houston, Expert Systems with Applications 180 (2021) p. 115104, https://doi.org/10.1016/j.eswa.2021.115104.

[20] K. Garcia, L. Berton, Topic detection and sentiment analysis in Twitter content related to COVID-19 from Brazil and the USA, Applied Soft Computing 101 (2021) p. 107057, https://doi.org/10.1016/j.asoc.2020.107057.

[21] M. El Akrouchi, H. Benbrahim, I. Kassou, End-to-end LDA-based automatic weak signal detection in web news, Knowledge-Based Systems 212 (2021) p. 106650, https://doi.org/10.1016/j.knosys.2020.106650.

[22] M.O. Lwin, J. Lu, A. Sheldenkar, P.J. Schulz, W. Shin, R. Gupta, Y. Yang, Global Sentiments Surrounding the COVID-19 Pandemic on Twitter: Analysis of Twitter Trends, JMIR Public Health and Surveillance 6 (2) (2020) p. e19447, https://doi.org/10.2196/19447.

[23] L. Zhao, X. Ding, F. Yu, Public moral motivation during the COVID-19 pandemic: Analysis of posts on Chinese social media, Social Behavior and Personality: An International Journal 48 (11) (2020) pp. 1-14, https://doi.org/10.2224/sbp.9829.

[24] N. Dey, R. Mishra, S.J. Fong, K.C. Santosh, S. Tan, R.G. Crespo, Covid-19: Psychological and Psychosocial Impact, Fear, and Passion, Digital Government: Research and Practice 2 (1) (2021) pp. 1-4, https://doi.org/10.1145/3428088.

[25] C.R. Machuca, C. Gallardo, R.M. Toasa, Twitter Sentiment Analysis on Coronavirus: Machine Learning Approach, Journal of Physics: Conference Series 1828 (1) (2021) p. 012104, https://doi.org/10.1088/1742-6596/1828/1/012104.

[26] I. Gupta, I. Chatterjee, N. Gupta, Sentiment Analysis of COVID-19 Tweets, 2022 1st International Conference on Informatics (ICI), 2022, pp. 229-231, https://doi.org/10.1109/ici53355.2022.9786887.

[27] A. Tareq, N. Hewahi, Sentiment Analysis of Tweets During COVID-19 Pandemic Using BLSTM, 2021 International Conference on Data Analytics for Business and Industry (ICDABI), 2021, pp. 245-249, https://doi.org/10.1109/icdabi53623.2021.9655932.

[28] A. Marathe, A. Mandke, S. Sardeshmukh, S. Sonawane, Leveraging Natural Language Processing Algorithms to Understand the Impact of the COVID-19 Pandemic and Related Policies on Public Sentiment in India, 2021 International Conference on Communication information and Computing Technology (ICCICT), 2021, pp. 1-5, https://doi.org/10.1109/iccict50803.2021.9510070.

[29] S. Daulatkar, A. Deore, Post Covid-19 Sentiment Analysis of Success of Online Learning: A Case Study of India, 2022 9th International Conference on Computing for Sustainable Global Development (INDIACom), 2022, pp. 460-465, https://doi.org/10.23919/INDIACom54597.2022.9763272.

[30] W. Qian, S. Bhowmick, M. O'Neill, S. Ramisetty-Mikler, A.R. Mikler, Applying a Probabilistic Infection Model for studying contagion processes in contact networks, Journal of Computational Science 54 (2021), https://doi.org/10.1016/j.jocs.2021.101419.

[31] P. Centorrino, A. Corbetta, E. Cristiani, E. Onofri, Managing crowded museums: Visitors flow measurement, analysis, modeling, and optimization, Journal of Computational Science 53 (2021), https://doi.org/10.1016/j.jocs.2021.101357.

[32] A. El Azzaoui, S.K. Singh, J.H. Park, SNS Big Data Analysis Framework for COVID-19 Outbreak Prediction in Smart Healthy City, Sustainable Cities and Society 71 (2021) p. 102993, https://doi.org/10.1016/j.scs.2021.102993.

[33] X. Zou, J. Yang, W. Zhang, H. Han, Collaborative community-specific microblog sentiment analysis via multi-task learning, Expert Systems with Applications 169 (2021) p. 114322, https://doi.org/10.1016/j.eswa.2020.114322.

[34] M. Salathé, S. Khandelwal, Assessing Vaccination Sentiments with Online Social Media: Implications for Infectious Disease Dynamics and Control, PLOS Computational Biology 7 (10) (2011) p. e1002199, https://doi.org/10.1371/journal.pcbi.1002199.

[35] L.A.M. Bostan, R. Klinger, An Analysis of Annotated Corpora for Emotion Classification in Text, The 27th International Conference on Computational Linguistics, Association for Computational Linguistics, 2018, pp. 2104–2119.

[36] O. Appel, F. Chiclana, J. Carter, H. Fujita, Successes and challenges in developing a hybrid approach to sentiment analysis, Applied Intelligence 48 (5) (2018) pp. 1176-1188, https://doi.org/10.1007/s10489-017-0966-4.

[37] T. Young, D. Hazarika, S. Poria, E. Cambria, Recent Trends in Deep Learning Based Natural Language Processing, IEEE Computational Intelligence Magazine 13 (3) (2018) pp. 55-75, https://doi.org/10.1109/MCI.2018.2840738.

[38] Y. Dang, Y. Zhang, H. Chen, A Lexicon-Enhanced Method for Sentiment Classification: An Experiment on Online Product Reviews, IEEE Intelligent Systems 25 (4) (2010) pp. 46-53, https://doi.org/10.1109/MIS.2009.105.

[39] C.S.G. Khoo, S.B. Johnkhan, Lexicon-based sentiment analysis: Comparative evaluation of six sentiment lexicons, Journal of Information Science 44 (4) (2017) pp. 491-511, https://doi.org/10.1177/0165551517703514.

[40] H. Cho, S. Kim, J. Lee, J.-S. Lee, Data-driven integration of multiple sentiment dictionaries for lexicon-based sentiment classification of product reviews, Knowledge-Based Systems 71 (2014) pp. 61-71, https://doi.org/10.1016/j.knosys.2014.06.001.

[41] A. Go, R. Bhayani, L. Huang, Sentiment Classification using Distant Supervision, Stanford, 2009, pp. 1-6.

[42] G. Zhang, Y. Pan, L. Zhang, Semi-supervised learning with GAN for automatic defect detection from images, Automation in Construction 128 (2021) p. 103764, https://doi.org/10.1016/j.autcon.2021.103764.

[43] Y. Pan, L. Zhang, Roles of artificial intelligence in construction engineering and management: A critical review and future trends, Automation in Construction 122 (2021) p. 103517, https://doi.org/10.1016/j.autcon.2020.103517.

[44] B. Chen, Q. Huang, Y. Chen, L. Cheng, R. Chen, Deep Neural Networks for Multi-class Sentiment Classification, 2018 IEEE 20th International Conference on High Performance Computing and Communications; IEEE 16th International Conference on Smart City; IEEE 4th International Conference on Data Science and Systems (HPCC/SmartCity/DSS), 2018, pp. 854-859, https://doi.org/10.1109/HPCC/SmartCity/DSS.2018.00142.

[45] Z. Ren, G. Zeng, L. Chen, Q. Zhang, C. Zhang, D. Pan, A Lexicon-Enhanced Attention Network for Aspect-Level Sentiment Analysis, IEEE Access 8 (2020) pp. 93464-93471, https://doi.org/10.1109/ACCESS.2020.2995211.

[46] C. Yue, H. Cao, G. Xu, Y. Dong, Collaborative attention neural network for multi-domain sentiment classification, Applied Intelligence 51 (6) (2021) pp. 3174-3188, https://doi.org/10.1007/s10489-020-02021-7.

[47] Y. Du, M. He, L. Wang, H. Zhang, Wasserstein based transfer network for cross-domain sentiment classification, Knowledge-Based Systems 204 (2020) p. 106162, https://doi.org/10.1016/j.knosys.2020.106162.

[48] F. Zhao, J. Bao, D. Ming, Battle Damage Assessment for Building based on Multi-feature, 2020 IEEE 5th Information Technology and Mechatronics Engineering Conference (ITOEC), 2020, pp. 57-60, https://doi.org/10.1109/ITOEC49072.2020.9141701.

[49] Z. Yuan, S. Wu, F. Wu, J. Liu, Y. Huang, Domain attention model for multi-domain sentiment classification, Knowledge-Based Systems 155 (2018) pp. 1-10, https://doi.org/10.1016/j.knosys.2018.05.004.

[50] R. Gupta, S. Sahu, C. Espy-Wilson, S. Narayanan, Semi-Supervised and Transfer Learning Approaches for Low Resource Sentiment Classification, 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2018, pp. 5109-5113, https://doi.org/10.1109/ICASSP.2018.8461414.

[51] K. Sailunaz, R. Alhajj, Emotion and sentiment analysis from Twitter text, Journal of Computational Science 36 (2019), https://doi.org/10.1016/j.jocs.2019.05.009.

[52] J. Bollen, H. Mao, X. Zeng, Twitter mood predicts the stock market, Journal of Computational Science 2 (1) (2011) pp. 1-8, https://doi.org/10.1016/j.jocs.2010.12.007.

[53] D. Leitch, M. Sherif, Twitter mood, CEO succession announcements and stock returns, Journal of Computational Science 21 (2017) pp. 1-10, https://doi.org/10.1016/j.jocs.2017.04.002.

[54] G. Hollis, Estimating the average need of semantic knowledge from distributional semantic models, Memory & Cognition 45 (8) (2017) pp. 1350-1370, https://doi.org/10.3758/s13421-017-0732-1.

[55] M. Dwarampudi, N.V.S. Reddy, Effects of padding on LSTMs and CNNs, ArXiv (2019) pp. 1-5, https://doi.org/10.48550/arXiv.1903.07288.

[56] Z. Lan, O. Sourina, L. Wang, R. Scherer, G.R. Müller-Putz, Domain Adaptation Techniques for EEG-Based Emotion Recognition: A Comparative Study on Two Public Datasets, IEEE Transactions on Cognitive and Developmental Systems 11 (1) (2019) pp. 85-94, https://doi.org/10.1109/TCDS.2018.2826840.

[57] Y. Zhang, F. Xiao, F. Qian, X. Li, VGM-RNN: HRRP Sequence Extrapolation and Recognition Based on a Novel Optimized RNN, IEEE Access 8 (2020) pp. 70071-70081, https://doi.org/10.1109/ACCESS.2020.2986027.

[58] H.R. Bhapkar, P.N. Mahalle, N. Dey, K.C. Santosh, Revisited COVID-19 Mortality and Recovery Rates: Are we Missing Recovery Time Period?, Journal of Medical Systems 44 (12) (2020) p. 202, https://doi.org/10.1007/s10916-020-01668-6.

[59] B.B. Briesemeister, L. Kuchinke, A.M. Jacobs, Emotional Valence: A Bipolar Continuum or Two Independent Dimensions?, SAGE Open 2 (4) (2012) p. 2158244012466558, https://doi.org/10.1177/2158244012466558.

[60] S. Kumar, Covid19 Indian Sentiments on covid19 and lockdown, (2020). https://www.kaggle.com/datasets/surajkum1198/twitterdata (accessed 22 September, 2022).

[61] A. Miglani, Coronavirus tweets NLP - Text Classification, (2020). https://www.kaggle.com/datasets/datatattle/covid-19-nlp-text-classification?select=Corona_NLP_test.csv (accessed 22 September, 2022).

[62] C. Lopez, COVID19_Tweets_Dataset, (2020). https://github.com/lopezbec/COVID19_Tweets_Dataset (accessed 22 September, 2022).

[63] Y. Yang, Y. Zhang, X. Zhang, Y. Cao, J. Zhang, Spatial evolution patterns of public panic on Chinese social networks amidst the COVID-19 pandemic, International Journal of Disaster Risk Reduction 70 (2022) p. 102762, https://doi.org/10.1016/j.ijdrr.2021.102762.

[64] J. Wang, C. Guo, X. Wu, P. Li, Influencing factors for public risk perception of COVID-19 -- perspective of the pandemic whole life cycle, International Journal of Disaster Risk Reduction 67 (2022) p. 102693, https://doi.org/10.1016/j.ijdrr.2021.102693.

[65] D.E. Cahyani, A.W. Putra, Relevance Classification of Trending Topic and Twitter Content Using Support Vector Machine, 2021 International Seminar on Application for Technology of Information and Communication (iSemantic), 2021, pp. 87-90, https://doi.org/10.1109/iSemantic52711.2021.9573243.

[66] Y. Cao, Y. Wu, J. Qi, Z. Chen, Multimodal Learning Approach for Multi-topic Twitter Summarization, 2022 7th International Conference on Cloud Computing and Big Data Analytics (ICCCBDA), 2022, pp. 139-144, https://doi.org/10.1109/ICCCBDA55098.2022.9778873.

[67] K. Roster, C. Connaughton, F.A. Rodrigues, Forecasting new diseases in low-data settings using transfer learning, Chaos, Solitons and Fractals 161 (2022) p. 112306, https://doi.org/10.1016/j.chaos.2022.112306.

[68] G. Panagopoulos, G. Nikolentzos, M. Vazirgiannis, Transfer Graph Neural Networks for Pandemic Forecasting, The Thirty-Fifth AAAI Conference on Artificial Intelligence, Vol. 35, 2021, pp. 4838-4845, https://doi.org/10.1609/aaai.v35i6.16616.

[69] M. Gupta, R. Jain, S. Arora, A. Gupta, M. Javed Awan, G. Chaudhary, H. Nobanee, AI-enabled COVID-9 Outbreak Analysis and Prediction: Indian States vs. Union Territories, Computers, Materials & Continua 67 (1) (2021) pp. 933-950, https://doi.org/10.32604/cmc.2021.014221.