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.

Elsevier hereby grants permission to make all its COVID-19-related research that is available on the COVID-19 resource centre - including this research content - immediately available in PubMed Central and other publicly funded repositories, such as the WHO COVID database with rights for unrestricted research re-use and analyses in any form or by any means with acknowledgement of the original source. These permissions are granted for free by Elsevier for as long as the COVID-19 resource centre remains active.
A new COVID-19 detection method from human genome sequences using CpG island features and KNN classifier

Hilal Arslan, Hasan Arslan

Department of Computer Engineering, Izmir Bakircay University, Izmir, Turkey
Department of Mathematics, Erciyes University, Kayseri, Turkey

Article history:
Received 13 October 2020
Revised 18 December 2020
Accepted 30 December 2020
Available online 9 January 2021

Keywords:
COVID-19
SARS-CoV-2
K-Nearest Neighbors
CpG islands
Human coronaviruses

Abstract

Various viral epidemics have been detected such as the severe acute respiratory syndrome coronavirus and the Middle East respiratory syndrome coronavirus in the last two decades. The coronavirus disease 2019 (COVID-19) is a pandemic caused by a novel betacoronavirus called severe acute respiratory syndrome coronavirus-2 (SARS-CoV-2). After the rapid spread of COVID-19, many researchers have investigated diagnosis and treatment for this terrifying disease quickly. Identifying COVID-19 from the other types of coronaviruses is a difficult problem due to their genetic similarity. In this study, we propose a new efficient COVID-19 detection method based on the K-nearest neighbors (KNN) classifier using the complete genome sequences of human coronaviruses in the dataset recorded in 2019 Novel Coronavirus Resource. We also describe two features based on CpG island that efficiently detect COVID-19 cases. Thus, genome sequences including approximately 30,000 nucleotides can be represented by only two real numbers. The KNN method is a simple and effective non-parametric technique for solving classification problems. However, performance of the KNN depends on the distance measure used. We perform 19 distance metrics investigated in five categories to improve the performance of the KNN algorithm. Some efficient performance parameters are computed to evaluate the proposed method. The proposed method achieves 98.4% precision, 99.2% recall, 98.8% F-measure, and 98.4% accuracy in a few seconds when any \(L_1\) type metric is used as a distance measure in the KNN.

Coronaviruses, which are positive sense and single-stranded RNA viruses, are known to have the largest viral genomes among all RNA viruses [1]. The family of coronaviruses has been classified into four genera [2]: alphacoronavirus (AlphaCoV), betacoronavirus (BetaCoV), gammacoronavirus (GammaCoV), and deltacoronavirus (DeltaCoV). AlphaCoV and BetaCoV infect mammalian hosts, however, GammaCoV and DeltaCoV mainly infect bird species [3].

There have been recorded various coronaviruses, which are the types of AlphaCoV and BetaCoV. Human coronavirus 229E (229E-CoV) and human coronavirus NL63 (NL63-CoV) are the types of AlphaCoV. Moreover, human coronavirus HKU1 (HKU1-CoV), Severe Acute Respiratory Syndrome coronavirus (SARS-CoV), and Middle East Respiratory Syndrome coronavirus (MERS-CoV) are the types of BetaCoV recorded recently. While 229E-CoV, NL63-CoV and HKU1-CoV cause simple respiratory traces, SARS-CoV and MERS-CoV cause fatal and dangerous respiratory infections.

SARS-CoV was first identified in the Guangdong province of southern China on 16 November 2002 [4] and MERS-CoV emerged in June 2012 in Jeddah, Saudi Arabia [5]. In late December 2019, a novel coronavirus SARS-CoV-2 emerged in China and spread rapidly throughout the world. The SARS-CoV-2 virus, which is genetically similar to SARS-CoV, caused a severe illness known as COVID-19 disease and a serious number of deaths worldwide. Since several early infected people visited a local seafood Market in December 2019 in Wuhan city of China, the virus is therefore thought to be a pathogen (SARS-CoV-2 pathogen) that jumped from an animal to a human and that caused an infectious disease. Similarity rate between SARS-CoV-2 and a bat coronavirus is 96% [6]. The outbreak COVID-19 was declared as a global pandemic on 11 March 2020 by the World Health Organization (WHO). As of 8 October 2020, the disease spread to 188 countries and territories has infected over 36.2 million people and has caused more than 1.05 million deaths; more than 25.2 million people have recovered from this illness.

https://doi.org/10.1016/j.jestch.2020.12.026
2215-0986/© 2020 Karabuk University. Publishing services by Elsevier B.V.
This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).
The most common symptoms of COVID-19 disease are cough, fever, gastrointestinal and musculoskeletal symptoms as well as loss of taste or smell. Shortness of breath and chest pressure or pain are among less common symptoms [7]. Because these common symptoms are similar to the common flu, it is difficult to conduct an early diagnosis of SARS-CoV-2. It is essential to quickly identify positive cases since it spreads rapidly and poses a threat to the public health system. Furthermore, there is still no specific antiviral drug recommended for the treatment of the novel coronavirus disease other than supportive care so far. In the period of pre-clinical researches, some medications already prescribed for other diseases have shown positive effects against COVID-19 virus. Due to the absence of an effective COVID-19 treatment, early detection of this life-threatening infectious disease plays a very important role in all medical therapies and in the prevention of the spread of the disease. Recently, many data scientists have been working extensively on remarkable features of the virus. For this purpose, artificial intelligence applications and machine learning methods have also been used successfully to speed up the diagnosis process of the COVID-19 cases [8–10]. Therefore, the learning based classification models will not only reduce the burden on healthcare professionals but will also facilitate early diagnosis of the disease.

In this paper, we introduce an effective classification method to distinguish SARS-CoV-2 from common human coronaviruses, which are AlphaCoV, BetaCoV-1, MERS-CoV, HKU1-CoV, NL63-CoV, and 229E-CoV using CpG island features and KNN classification method. Main contributions of this research can be summarized as follows:

- The choice of the differentiable features based on characteristic of SARS-CoV-2 is a critical step to improve classification performance. In this study, we propose CpG island features to differentiate SARS-CoV-2 from the other human coronaviruses.
- We propose a robust prediction method by using KNN classifier with any $L_1$ type distance metric selected from among 19 distance metrics investigated in five categories, which are $L_1$ type, $L_2$ type, vicissitude, inner product metrics, and the other types of metrics.
- We construct a larger dataset containing almost all types of these human coronaviruses which are genetically similar to SARS-CoV-2. They are AlphaCoV, BetaCoV-1, MERS-CoV, NL63-CoV, HKU1-CoV, and 229E-CoV.
- The proposed method efficiently detects COVID-19 cases in a few seconds on the relatively large dataset.

The rest of the paper is organized as follows. In Section 2, a literature survey on COVID-19 is presented. In Section 3, the K-nearest neighbors (KNN) method and distance measures used in the KNN method are summarized. The proposed classification strategy to detect COVID-19 cases is introduced in Section 4. Section 5 reports experimental results and evaluation of the experimental observations conducted by various distance metrics. In Section 6, we compare the proposed method with the other detection methods. Finally, Section 7 presents concluding remarks and future directions.

2. Related works

Deep learning and machine learning techniques have been widely used in a variety of research areas such as big data analysis [11–13], image classification [14], face detection [15,16], and disease prediction [17–21]. The computer based diagnostic systems developed by the help of artificial intelligence techniques will speed up early diagnosis of COVID-19 and thus will decrease the workload on health-workers. Chen et al. [22] published a comprehensive survey about studies performed by using artificial intelligence in the literature on the fight against COVID-19. In addition, they observed that machine learning, deep learning and artificial neural networks technologies have been successfully implemented at almost every stage of combating COVID-19 when compared to other coronaviruses such as SARS-CoV and MERS-CoV.

Randhawa et al. [23] suggested a combination of supervised machine learning with digital signal processing methods (ML-DSP) for an accurate and scalable taxonomic classification of genomic sequences. They matched each genomic sequence to discrete values corresponding to its genomic signals. They employed six supervised machine learning classifiers (linear discriminant, linear support vector machine, quadratic support vector machine, fine KNN, subspace discriminant, and subspace KNN) to detect SARS-CoV-2. They used 29 SARS-CoV-2 genome sequences and 20 genome sequences for each of alphacoronavirus, betacoronavirus, and delta-coronavirus. They observed that the linear discriminant method achieved 100% accuracy. Naem et al. [24] proposed a method to distinguish COVID-19, SARS-CoV, and MERS-CoV viruses by using the K-nearest neighbors and the trainable cascade-forward back-propagation neural network methods. They extracted genomic signal processing features using a dataset that contains 76 genome sequences for each type of coronavirus from the National Center for Biotechnology Information (NCBI). Their results showed that performance of the KNN algorithm was higher than the cascade-forward back-propagation neural network in all COVID-19/SARS-CoV, COVID-19/MERS-CoV and COVID-19/SARS-CoV/MERS-CoV classification processes, and achieved an accuracy of 100%. Batista et al. [25] developed a new method to predict whether patients in the emergency care unit are COVID-19 positive or not by using five supervised machine learning algorithms, which are neural networks, random forests, logistic regression, support vector machines, and gradient boosting regression trees. They collected data from 235 patients, and they used 15 different types of features such as age, gender, hemoglobin etc. Their experimental results showed that the support vector machine classifier achieved the highest performance with AUC value of 84.7% among the other classifiers.

Unal and Dudak [26] studied on diagnosis of COVID-19 viral disease. They applied Naive Bayes, KNN, support vector machines and decision tree algorithms to the dataset named as COVID-19 Mexico Patient Health Dataset. The dataset consists of 95839 cases recorded by Mexican government, and 19 different types of features like as the sex of the patient, age of the patient, the state of pneumonia and intubation as well as the state of many other diseases. They performed four types of supervised machine learning algorithms. Their experimental results showed that support vector machine achieved the best predictive performance with the classification accuracy of 100%.

Although there are only few studies detecting COVID-19 cases from genome sequences, a number of papers have been published for detecting COVID-19 cases from X-ray or computed tomography (CT) images [27–34], recently. Barstugan et al. [27] developed a COVID-19 classification method by using the support vector machine classifier on a dataset including different types of 150 abdominal CT images. Their experimental results showed that their proposed method achieved an accuracy of 99.6%. Ozturk et al. [28] applied the DarkCovidNet deep learning model to raw chest X-ray images and generated a binary classification (COVID-19, no findings). They also used the DarkCovidNet classifier method on the same dataset to create a triple classification (COVID-19, no-findings, pneumonia). They declared that the highest accuracy of the classifier was 98.08% for the binary classification and 87.02% for the triple case. Sekeroğlu et al. [29] proposed an alternative COVID-19 detection method by using deep learning and machine
learning classifiers on a publicly available dataset which contains 1583 healthy, 4292 pneumonia and 225 confirmed COVID-19 chest X-ray images. They stated that a convolutional neural network (CNN) without pre-processing and with minimized layers achieved an accuracy of 98.50%. Jain et al. [30] developed a new approach to detect COVID-19 cases among chest X-ray images of healthy, bacterial pneumonia, viral pneumonia and COVID-19 by using ResNet18, ResNet101, DenceNet121 and VGG-16 deep learning models. Their experimental results demonstrated that ResNet101 method achieved the highest accuracy of 98.93%. Apostolopoulos and Bessiana [31] used VGG19, Mobile Net, Inception, Xception and Inception ResNet v2 deep transfer learning classifiers to detect COVID-19 positive cases among 224 COVID-19, 700 bacterial pneumonia and 504 normal chest X-ray images. Their results presented that VGG19 achieved the best binary classification accuracy of 98.75% over the other CNN methods.

Ahuja et al. [32] used deep learning methods to detect COVID-19 positive cases from chest X-ray images. Asnaoui and Chawki [33] constructed a novel method for detecting COVID-19 from pneumonia chest X-ray and tomography images. They performed recent deep learning methods, VGG16, VGG19, DenseNet201, Inception_ResNet_V2, Inception_V3, Resnet50, and MobileNet_V2. They reported that the performance of Inception_Resnet_V2 demonstrated the best accuracy of 92.18%. Basu et al. [34] proposed an alternative screening rule. Their experimental results demonstrate that the proposed method is less sensitive to k by introducing the generalized mean distance-based KNN classifier. They stated that the proposed method is less sensitive to k over the KNN-based classifier.

3.2. Metrics

To improve the performance of the KNN classifier, we investigate the metrics used in the KNN algorithm under five categories by following the similar categorization in [39,40]. We list them as below:

1. **L₁ type metrics**: Six metrics are investigated in this category. They are Manhattan metric (ManM), Chebyshev metric (ChebM), Canberra metric (CanM), Sorensen metric (SM), Kuleyynski metric (KM), and Mean character metric (MCM), which are explained in 1.

2. **L₂ type metrics**: Seven metrics are investigated in this category. They are Euclidean metric (EM), Clark metric (ClaM), Neyman χ² metric (NCSM), Squared χ² metric (SquM), Divergence metric (DivM), and Squared Chi-squared metric (SCSM). We list these metrics in 2.

3. **Vicissitude metrics**: Three metrics are considered in this category. They are Vicis Symmetric 1 metric (VSDFM1), Vicis Symmetric 2 metric (VSDFM2) and Vicis Symmetric 3 metric (VSDFM3). Definitions of the mentioned metrics are given in 3.

4. **Inner product metrics**: Two metrics are investigated in this category. They are Dice metric (DicM) and Chord metric (ChoM). Definition of the mentioned metrics are demonstrated in 4.

5. **Other metrics**: In this category, we perform two metrics that significantly affect the performance of the KNN algorithm. These metrics are Motyka (MotM) and Hassanat (HasM) metrics. The description of them is shown in 5, where

\[
D(x_i, y_j) = \begin{cases} 
1 - \frac{\min(x_i, y_j)}{\max(x_i, y_j)}, & \text{if } \min(x_i, y_j) > 0 \\
1 - \frac{\min(x_i, y_j) - \min(x_i, y_j)}{\max(x_i, y_j) - \min(x_i, y_j)}, & \text{if } \min(x_i, y_j) < 0
\end{cases}.
\]

4. Proposed method

In this section, we present a new COVID-19 detection method based on CpG island features and the KNN classifier. The KNN is extremely useful for large data classification and its performance depends on the distance metric used [35–38]. Several studies have been conducted to detect optimum metrics for KNN algorithm in [39,40]. In this section, first, we provide a brief description of the KNN method and some recent studies based on the KNN. Second, we compare the metrics used in the KNN classifier under five categories to improve the performance of the KNN classifier on our model.

### 3.1. K-nearest neighbors

K-nearest neighbors (KNN), a supervised machine learning algorithm, can be efficiently used to solve classification problems. The KNN was introduced in 1951 by [41] and then recasted in 1967 by [42]. The KNN is a non-parametric classifier known as one of the simplest and laziest algorithms. That is, there is no need to create a learning model in this classification method. Despite the lazy structure of KNN, it was proposed as one of the 10 most effective methods in the process of analyzing information in a database given [43]. In the prediction process, the class to which a new observation data belongs is determined by calculating the shortest distance between the observation sample and its K-nearest neighbors samples.

There are some recent studies to improve the performance of the KNN. To decrease the sensitivity of the neighborhood size of k and improve voting strategy in the region of neighborhoods, Gou et al. [44] proposed two k-nearest neighbor rules, which are the weighted representation-based k-nearest neighbor rule and the weighted local mean representation-based k-nearest neighbor rule. Their experimental results demonstrate that the proposed methods have a lower sensitiveness to k. Gou et al. [45] proposed another study to improve the selection of the neighborhood size of k by introducing the generalized mean distance-based KNN classifier. They stated that the proposed method is less sensitive to k over the KNN-based classifier.

### 3.2. Metrics

To improve the performance of the KNN classifier, we investigate the metrics used in the KNN algorithm under five categories by following the similar categorization in [39,40]. We list them as below:

1. **L₁ type metrics**: Six metrics are investigated in this category. They are Manhattan metric (ManM), Chebyshev metric (ChebM), Canberra metric (CanM), Sorensen metric (SM), Kuleyynski metric (KM), and Mean character metric (MCM), which are explained in 1.

2. **L₂ type metrics**: Seven metrics are investigated in this category. They are Euclidean metric (EM), Clark metric (ClaM), Neyman χ² metric (NCSM), Squared χ² metric (SquM), Divergence metric (DivM), and Squared Chi-squared metric (SCSM). We list these metrics in 2.

3. **Vicissitude metrics**: Three metrics are considered in this category. They are Vicis Symmetric 1 metric (VSDFM1), Vicis Symmetric 2 metric (VSDFM2) and Vicis Symmetric 3 metric (VSDFM3). Definitions of the mentioned metrics are given in 3.

4. **Inner product metrics**: Two metrics are investigated in this category. They are Dice metric (DicM) and Chord metric (ChoM). Definition of the mentioned metrics are demonstrated in 4.

5. **Other metrics**: In this category, we perform two metrics that significantly affect the performance of the KNN algorithm. These metrics are Motyka (MotM) and Hassanat (HasM) metrics. The description of them is shown in 5, where

\[
D(x_i, y_j) = \begin{cases} 
1 - \frac{\min(x_i, y_j)}{\max(x_i, y_j)}, & \text{if } \min(x_i, y_j) > 0 \\
1 - \frac{\min(x_i, y_j) - \min(x_i, y_j)}{\max(x_i, y_j) - \min(x_i, y_j)}, & \text{if } \min(x_i, y_j) < 0
\end{cases}.
\]
Table 2

| Abbreviation | Metric     | Definition                          |
|--------------|------------|-------------------------------------|
| EM           | Euclidean  | $\sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$ |
| ClaM         | Clark metric | $\sqrt{\sum_{i=1}^{n}\left(\frac{x_i}{x_i+y_i}\right)^2}$ |
| NCSM         | Neyman $\chi^2$ metric | $\sum_{i=1}^{n} \left(\frac{x_i}{x_i+y_i}\right)^2$ |
| SquM         | Squared $\chi^2$ metric | $\sum_{i=1}^{n} \left(\frac{x_i}{x_i+y_i}\right)^2$ |
| DivM         | Divergence metric | $2\sum_{i=1}^{n} \left(\frac{x_i}{x_i+y_i}\right)$ |
| SCSM         | Squared Chi-squared metric | $\sum_{i=1}^{n} \left(\frac{x_i}{x_i+y_i}\right)^2$ |

Table 3

| Abbreviation | Metric Definition |
|--------------|-------------------|
| VSDFM1       | Vicis Symmetric 1 metric | $\frac{\sum_{i=1}^{n}(x_i-y_i)^2}{\sum_{i=1}^{n}(x_i+y_i)}$ |
| VSDFM2       | Vicis Symmetric 2 metric | $\frac{\sum_{i=1}^{n}(x_i-y_i)^2}{\sum_{i=1}^{n}(x_i+y_i)}$ |
| VSDFM3       | Vicis Symmetric 3 metric | $\frac{\sum_{i=1}^{n}(x_i-y_i)^2}{\sum_{i=1}^{n}(x_i+y_i)}$ |

Table 4

| Abbreviation | Metric | Definition |
|--------------|--------|------------|
| DivM         | Dice metric | $1 - \frac{\sum_{i=1}^{n}(x_i-y_i)^2}{\sum_{i=1}^{n}(x_i+y_i)}$ |
| ChoM         | Chord metric | $\sqrt{2 - 2\sum_{i=1}^{n}(x_i-y_i)^2}$ |

Table 5

| Abbreviation | Metric     | Definition                          |
|--------------|------------|-------------------------------------|
| MotM         | Motyka metric | $\sum_{i=1}^{n}D(x_i, y_i)$ |
| HasM         | Hassanat metric | $\sum_{i=1}^{n}\frac{max(x_i, y_i)}{\sum_{j=1}^{n}D(x_i, y_i)}$ |

$CPGo = \frac{\text{ratio}(CG)}{\text{ratio}(C) \cdot \text{ratio}(G)}$ (2)

where $\text{ratio}(C)$, $\text{ratio}(G)$, and $\text{ratio}(CG)$ are computed as divided the number of occurrences of C, G, and CG in the sequence, respectively by the sequence length. Thus, each sequence containing 30 000 nucleotides is represented by two features only. 1 provides an example of how features are calculated from a part of the sequence.

Algorithm 1. Proposed Algorithm

Require:
S: training genome sequences,
L: COVID-19 positive, COVID-19 negative
seq: a test sequence, and k: the neighborhood size
Ensure:
Determine the class label of the test sequence seq
Step 1: Feature Extraction
1: for all genome sequences do
2: $feature_1 = \text{ratio}(C) + \text{ratio}(G)$
3: $feature_2 = \frac{\text{ratio}(CG)}{\text{ratio}(C) \cdot \text{ratio}(G)}$
4: end for
Step 2: Apply KNN and use $L_1$ type metric
5: Compute the distance between seq and every sample in S using any $L_1$ type metric
6: Choose k samples in S that are nearest to seq
7: Assign seq to majority class

5. Experimental results

In this section, we evaluate performance of the proposed method. Before discussing the results, first we explain our dataset, second we mention experimental setup, and then we provide a brief information about the performance measures. Finally, we will give the results of the experimental observations conducted by using different distance metrics.

5.1. Dataset

The 2019 novel coronavirus resource (2019nCoVR) [49] at China National Center for Bioinformation is one of the most important sources of various types of coronaviruses. It integrates various important databases including the GISAID, NCBI, NMDC and CNCB/NGDC. In this study, we used complete genomic sequences of human coronaviruses obtained from 2019nCoVR. Genome sequence of each coronavirus has approximately a length of 30 000 nucleotides. The properties of human coronavirus sequences are presented in 6. The dataset includes various types of human coronaviruses such as AlphaCoV, BetaCoV-1, MERS-CoV, NL63-CoV, HKU1-CoV and 229E-CoV as well as SARS-CoV-2. We refer SARS-CoV-2 sequences as COVID-19 positive, and the sequences that do not include SARS-CoV-2 are referred as COVID-19 negative. In addition to 1000 SARS-CoV-2 sequences, we used 592 genome sequences of other human coronaviruses in our experiments. We note that all available genome sequences of human coronaviruses, which are different from SARS-CoV-2 are downloaded.

5.2. Experimental setup

The experiments were performed using a core i7, 2.7 GHz processor, 16 GB RAM under Linux operating system. Feature extrac-

Table 6

| Human coronaviruses | The number sequences | Label |
|---------------------|----------------------|-------|
| SARS-CoV-2          | 1000                 | Positive |
| AlphaCoV            | 88                   | Negative |
| BetaCoV-1           | 140                  | Negative |
| MERS-CoV            | 258                  | Negative |
| NL63-CoV            | 61                   | Negative |
| HKU1-CoV            | 18                   | Negative |
| 229E-CoV            | 27                   | Negative |
tion and the classification process were performed using Python programming language. Precision, recall, F-measure, and accuracy in predicting COVID-19 positive case are used as the major performance measures. Next, we define these performance measures.

5.3. Performance measures

COVID-19 prediction as a binary classification problem has four prediction outcomes. True Positive (TP) is the number of genomic sequences which are correctly classified as COVID-19 positive, True Negative (TN) is the number of genome sequences which are correctly classified as COVID-19 negative, False Positive (FP) is the number of genome sequences which are incorrectly classified as COVID-19 positive, and False Negative (FN) is the number of sequences which are incorrectly classified as COVID-19 negative. The performance evaluation metrics such as precision, recall, F-measure, and accuracy are defined by using these outcomes and presented in confusion matrix by assigning actual and predicted labels in 2. Next, we briefly explain performance parameters used in this study.

Precision is the accuracy of the classifier in the presence of false positive case. It is computed as the ratio of the number of correctly classified positive samples to the number of samples labeled by the system as positive and it is exhibited in Eq. 3.

\[
\text{Precision} = \frac{TP}{TP + FP}
\]  

Recall refers to as the number of positive class predictions made out all positive samples in the data set. It is computed as in Eq. 4.

\[
\text{Recall} = \frac{TP}{TP + FN}
\]

F-Measure is determined by the harmonic mean of precision and recall, and computed using Eq. 5.

\[
F - \text{Measure} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}
\]

Accuracy is computed by dividing the total number of true cases by all cases and it is indicated in Eq. 6.

\[
\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}
\]

5.4. Evaluation of the proposed features and KNN classifier with respect to metrics

In this section, we evaluate the efficiency of proposed features and the performance of the KNN classifier with respect to metrics using a fivefold cross-validation that was performed on the dataset. The dataset is randomly divided into two sets which are training and testing. Eighty percent of the entire human genome sequences is used for training and the remaining 20% of the dataset is used for validation. The experiments are repeated five times as presented in 3. In our experiments, we increase the neighborhood size of k between 1 and 20, and observe that the classification performance of the proposed method remains the same or slightly decreases with increasing the neighborhood size of k. For this reason, in all experiments, the k value is basically set to 1.

7 presents precision, recall, and F-measure values of the KNN classifier with respect to five metric groups, which are L1 type, L2 type, vicissitude, inner product metrics, and the other types of metrics. The results of the metrics in the same group are close to each other; thus, we present the average of the results of the experiments performed by the metrics in the same group. First, we analyze the results of the KNN equipped with L1 type metrics, which are KM, ChebM, ManM, SM, CanM, and MCM. The KNN with L1 type metrics presents the best result, and achieves a precision of 98.4%, a recall of 99.2%, and an F-measure of 98.8%. This means that almost all sequences are classified correctly. Next we look at the results of the KNN with L2 type metrics, which are ClAM, EM, SquM, NCsM, DivM, and SCSM. The KNN classifier with this group metrics achieves a precision of 96.0%, a recall of 98.2%, and an F-measure of 97.1% on average. When we look at the results of the KNN with vicissitude metrics, it presents better result than KNN with L2 type metrics. It achieves a precision of 98.4%, a recall of 99.0%, and an F-measure of 98.7%. Next, we investigate the results of the KNN with inner product metrics, ChoM and DiCM. This group of metrics presents the worst results among the other groups, and achieves a precision of 94.4%, a recall of 92.8%, and an F-measure of 93.4%. Finally, we investigate the other type of metrics, which are HasM and MotM. The results of KNN with HasM are close to the results of the KNN with inner product type metrics, and achieves a precision of 94.4%, a recall of 92.9%, and an F-measure of 93.4%. On the other hand, the results of the KNN with MotM is close to them of L2 type metrics, and are better than that of HasM. It achieves a precision of 96.4%, a recall of 98.2%, and an F-measure of 97.3%.

In addition to precision, recall, and F-measure values, we present accuracy results. We take the average of the accuracies of KNN classifier obtained with the metrics in the same group, and show the results for each metric group separately in 4. The accuracy results of the KNN classifier using inner product metrics and HasM are close to each other and have the worst accuracy, 91.7%, and 91.8%, respectively. When L2 type metrics are used, the method achieves an accuracy of 96.2%. The KNN with MotM metric has remarkable accuracy values, which is 96.5%. The accuracy values of the KNN with L1 and Viciissitude type metrics are close to each other, and our method achieves the best accuracy with 98.4% when L1 type metrics are used.
In this part, we compare the results of the proposed method with the state-of-the-art methods. Table 7 provides a simple comparison of the results in terms of the method, dataset, class, and accuracy. We investigate these studies under two categories: the studies using genome sequences datasets and the other datasets.

First, we compare our method with the studies which detect COVID-19 cases from genome sequences. Randhawa et al. [23] combined supervised machine learning methods with digital signal processing. They used 29 sequences of SARS-CoV-2, and 20 sequences for each of alphacoronavirus, betacoronavirus, and deltacoronavirus. They achieved 100% accuracy when linear discriminant method was used. Main limitation of their proposed method is the number of sequences used in their dataset. Furthermore, Randhawa et al. used delta coronaviruses genomes. However, deltacoronavirus mainly causes an infectious disease among bird species rather than human. When the number of sequences in their dataset is increased and human coronavirus sequences genetically similar to SARS-CoV-2 sequences are added to their dataset, overall accuracy of their method may decrease. Another study predicting COVID-19 from genome sequences is introduced by Naeem et al. [24]. They used the classical KNN method to distinguish SARS-CoV-2 sequences among SARS-CoV-2, SARS-CoV and MERS-CoV genome sequences. They used 76 sequences for each of SARS-CoV-2, SARS-CoV, and MERS-CoV, and achieved an accuracy of 100%. They extracted features by using Discrete Fourier transform, Discrete cosine transform, and Seven Moment Invariants methods. When comparing to our method, their feature

Table 7
Average of the precision, recall, and F-measure values.

| Group        | Metric name | Precision (%) | Recall (%) | F-measure (%) |
|--------------|-------------|---------------|------------|---------------|
| L1           | KM          | 98.4          | 99.2       | 98.8          |
|              | ChebM       |               |            |               |
|              | ManM        |               |            |               |
|              | SM          |               |            |               |
|              | CanM        |               |            |               |
|              | MCM         |               |            |               |
| L2           | ClaM        | 96.0          | 98.2       | 97.1          |
|              | EM          |               |            |               |
|              | SquM        |               |            |               |
|              | NCSM        |               |            |               |
|              | DivM        |               |            |               |
|              | SCSM        |               |            |               |
| Vicissitude  | VSDFM1      | 98.4          | 99.0       | 98.7          |
|              | VSDFM2      |               |            |               |
|              | VSDFM3      |               |            |               |
| Inner Product| ChoM        | 94.4          | 92.8       | 93.4          |
|              | DicM        |               |            |               |
| Other        | HasM        | 94.4          | 92.9       | 93.4          |
|              | MotM        | 96.4          | 98.2       | 97.3          |

Fig. 4. Accuracy of the proposed method with respect to each group of metric.

6. Comparisons with other methods

In this part, we compare the results of the proposed method with the state-of-the-art methods. 8 provides a simple comparison...
| Study                  | Method used                                      | Dataset                                      | Classes                                    | Accuracy (%) |
|-----------------------|--------------------------------------------------|----------------------------------------------|--------------------------------------------|--------------|
| Randhawa et al. [23]  | Linear Discriminant                               | 29 COVID-19 virus sequences                 | COVID-19 vs other types of viruses         | 100          |
|                       | Linear SVM                                        | 20 alphacoronavirus sequences                |                                            |              |
|                       | Quadratic SVM                                     | 20 betacoronavirus sequences                |                                            |              |
|                       | Fine KNN                                          | 20 deltacoronavirus sequences               |                                            |              |
|                       | Subspace Discriminant                             |                                              |                                            |              |
|                       | Subspace KNN                                      |                                              |                                            |              |
| Naeem et al. [24]     | Discrete Fourier Transform with KNN               | 76 COVID-19 sequences                       | COVID-19 vs SARS-CoV                       | 100          |
|                       | Discrete Cosine transform with KNN                | 76 SARS-CoV sequences                       |                                            |              |
|                       | Seven Moment Invariants with KNN                  | 76 MERS-CoV sequences                       | COVID-19 vs MERS-CoV                       |              |
| Proposed Method       | KNN with L1 type metrics                          | 1000 COVID-19 sequences                     | COVID-19 vs other type coronaviruses       | 98.4         |
|                       | CpG based features                                 | 592 other coronavirus sequences             |                                            |              |
| Unal and Dudak [26]   | Naive Bayes                                       | 95839 case with 19 type of laboratory findings | COVID-19 vs other diseases               | 100          |
|                       | KNN                                               |                                              |                                            |              |
|                       | Support Vector Machines                           |                                              |                                            |              |
|                       | Decision Tree Algorithm                          |                                              |                                            |              |
| Barstugan et al. [27] | Support Vector Machine                            | 150 Abdominal CT images                     | COVID-19 vs other viral pneumonias         | 99.6         |
| Apostolopoulos and Bessiana [31] | Deep Transfer Learning                      | 224 COVID-19 Chest X-ray images            | COVID-19 vs Pneumonia vs Normal           | 98.75        |
|                       |                                                  | 700 images of bacterial pneumonia           |                                            |              |
|                       |                                                  | 504 images of normal conditions             |                                            |              |
| Ozturk et al. [28]    | DarkCovidNet                                      | raw chest X-ray images                      | COVID-19 vs healthy vs pneumonia           | 87.02        |
| Jain et al. [30]      | VGG16                                            | chest X-ray images                          | COVID-19 vs bacterial pneumonia vs viral pneumonia vs healthy | 98.93        |
There are a number of the existing studies detecting COVID-19 cases from image dataset in the literature. In 8, we also exhibit six image based studies achieving remarkable accuracy values. These image based methods used different deep learning or machine learning methods. Although image based studies have a higher accuracy, when considering a high mutation rate of SARS-CoV-2, extracting features from image dataset is large, their dataset only represents the features of a specific region. Furthermore, they used 19 different features including the sex of the patient, age of the patient, the state of pneumonia and intubation as well as the state of many other damages. One of the advantages of our model over the method of Unal and Dudak [26] is to use powerful and effective two features derived from the complete genome sequences of human coronaviruses. Thus, our proposed method detects COVID-19 positive cases within a few seconds. However, Unal and Dudak [26] did not provide any explanation about the detection time. Another different aspects of the proposed method from their method is that they use the classical KNN classifier with default parameters although we use the KNN classifier with an optimum distance metric in our proposed model.

There are a number of the existing studies detecting COVID-19 cases from image dataset from the literature. In 8, we also exhibit six image based studies achieving remarkable accuracy values. These image based methods used different deep learning or machine learning methods. Although image based studies have a higher accuracy, when considering a high mutation rate of SARS-CoV-2, using genomic sequences is extremely beneficial when tracking coronavirus genes that change frequently as the disease spreads from one person to another. Moreover, the radiation generated by X-ray or CT scanning machines may cause permanent damages to people. For this reason, X-ray or CT scanning may not be obtained for some people, which can be considered as a disadvantage of image based studies.

7. Conclusion

COVID-19 is an existing epidemic that sets new records in terms of cumulative and daily numbers for global infection. The pandemic is an unprecedented situation in healthcare systems worldwide, and to overcome this pandemic, it is essential to accurately detect COVID-19 cases by analyzing data of patients in a minimum amount of time. In this study, we propose an accurate and fast method to detect COVID-19 positive cases from genome sequences of human coronaviruses. In the proposed method, first, the features that significantly differentiate COVID-19 cases are extracted from the complete genome sequences of human coronaviruses. In this step, we propose to use CpG island features. Each genome sequence of human coronavirus, which includes about 30,000 nucleotides is represented by two real numbers only. Feature extraction step takes just a few seconds. Second, the KNN method is used for the classification of COVID-19 positive cases from the other types of human coronaviruses, AlphaCoV, Beta-CoV-1, MERS-CoV, NL63-CoV, HKU1-CoV, and 229E-CoV. The KNN classifier is the simplest method and has high flexibility for solving complex classification problems. The accuracy of the KNN is higher than state-of-the-art classifiers in certain cases, and it often produces efficient performance. However, the performance of the KNN greatly depends on the metric performed. To detect the most appropriate metric, we review five groups of metrics used in the KNN classifier. The selection of different distance metrics for the KNN can result in a variation in accuracy outcomes for the same dataset. Experimental results reveal that the proposed method achieves the highest accuracy, which is 98.4% on average in a few seconds when L1 type metrics are used as a distance measure in the KNN. In future studies, we will compare human SARS-CoV-2 sequences to other types of coronavirus sequences such as bat SARS-CoV-like coronaviruses 2, and propose a similarity based feature to increase overall accuracy. In addition, the factors affecting the recovery status of patients suffering from COVID-19 may be investigated in future studies by combining machine learning and parallel computing methods with effective features.

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.

References

[1] D. Schoeman, B. Fielding, Coronavirus envelope protein: current knowledge, Virol. J. 16 (2020) 10.1186/s12985-019-1182-0.
[2] R.J. de Groot, S. Baker, R. Baric, L. Enjuanes, A. Gorbalenya, K. Holmes, S. Perlman, L. Poon, P. Rottier, P. Talbot, et al., Family coronaviridae, Virus Taxon. (2012) 806–828.
[3] J. Cui, F. Li, Z.L. Shi, Origin and evolution of pathogenic coronaviruses, Nat. Rev. Microbiol. 17 (2019) 181–192, https://doi.org/10.1038/s41579-018-0093-0.
[4] A chronicle on the sars epidemic, Chin. Law Govera. 36 (4) (2003) 12–15, arXiv:https://doi.org/10.2753/CLG0009-4693060412, doi: 10.2753/CLG0009-4693060412.
[5] Z. Zumila, D.S. Hui, S. Perlman, Middle east respiratory syndrome, The Lancet 386 (9997) (2015) 955–1007, https://doi.org/10.1016/S0140-6736(15)60454-8, http://www.sciencedirect.com/science/article/pii/S0140673615604548.
[6] L. Xingguang, W. Wei, Z. Xiaofang, Z. Junjie, Z. Qiang, L. Yi, C. Antoine, Transmission dynamics and evolutionary history of 2019-ncov, J. Med. Virol. 92 (2020) 501–511, https://doi.org/10.1002/jmv.25701.

---

Table 8 (continued)

| Study                          | Method used                  | Dataset                                      | Classes                                       | Accuracy (%) |
|-------------------------------|------------------------------|----------------------------------------------|-----------------------------------------------|--------------|
| Asnaoui and Chawki [33]       | VGG16                        | 1583 normal X-ray and CT images              | COVID-19 vs other viral pneumonias            | 92.18        |
|                               | VGG19                        | 231 COVID-19 images                          |                                               |              |
|                               | DenseNet201                  | 2780 images of bacterial pneumonia           |                                               |              |
|                               | Inception ResNet V2         | 1493 coronavirus images                      |                                               |              |
|                               | Inception V3                |                                              |                                               |              |
|                               | ResNet50                    |                                              |                                               |              |
|                               | MobileNet V2                |                                              |                                               |              |
| Basu et al. [34]              | Domain Extension Transfer Learning | chest X-ray images                          | COVID-19 vs normal vs pneumonia vs other diseases | 95.3         |

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.

References

[1] D. Schoeman, B. Fielding, Coronavirus envelope protein: current knowledge, Virol. J. 16 (2020) 10.1186/s12985-019-1182-0.
[2] R.J. de Groot, S. Baker, R. Baric, L. Enjuanes, A. Gorbalenya, K. Holmes, S. Perlman, L. Poon, P. Rottier, P. Talbot, et al., Family coronaviridae, Virus Taxon. (2012) 806–828.
[3] J. Cui, F. Li, Z.L. Shi, Origin and evolution of pathogenic coronaviruses, Nat. Rev. Microbiol. 17 (2019) 181–192, https://doi.org/10.1038/s41579-018-0093-0.
[4] A chronicle on the sars epidemic, Chin. Law Govera. 36 (4) (2003) 12–15, arXiv:https://doi.org/10.2753/CLG0009-4693060412, doi: 10.2753/CLG0009-4693060412.
[5] Z. Zumila, D.S. Hui, S. Perlman, Middle east respiratory syndrome, The Lancet 386 (9997) (2015) 955–1007, https://doi.org/10.1016/S0140-6736(15)60454-8, http://www.sciencedirect.com/science/article/pii/S0140673615604548.
[6] L. Xingguang, W. Wei, Z. Xiaofang, Z. Junjie, Z. Qiang, L. Yi, C. Antoine, Transmission dynamics and evolutionary history of 2019-ncov, J. Med. Virol. 92 (2020) 501–511, https://doi.org/10.1002/jmv.25701.
