



## Article

# A Convolution Neural Network-Based Representative Spatio-Temporal Documents Classification for Big Text Data

Byoungwook Kim <sup>1</sup>, Yeongwook Yang <sup>2</sup>, Ji Su Park <sup>3</sup> and Hong-Jun Jang <sup>3,\*</sup>

<sup>1</sup> Department of Computer Science and Engineering, Dongshin University, Naju 58245, Korea; bwkim@dsu.ac.kr

<sup>2</sup> Division of Computer Engineering, Hanshin University, Osan 18101, Korea; yeongwook.yang@hs.ac.kr

<sup>3</sup> Department of Computer Science and Engineering, Jeonju University, Jeonju 55069, Korea; jisupark@jj.ac.kr

\* Correspondence: hongjunjang@jj.ac.kr; Tel.: +82-63-220-2372

**Abstract:** With the proliferation of mobile devices, the amount of social media users and online news articles are rapidly increasing, and text information online is accumulating as big data. As spatio-temporal information becomes more important, research on extracting spatiotemporal information from online text data and utilizing it for event analysis is being actively conducted. However, if spatiotemporal information that does not describe the core subject of a document is extracted, it is rather difficult to guarantee the accuracy of core event analysis. Therefore, it is important to extract spatiotemporal information that describes the core topic of a document. In this study, spatio-temporal information describing the core topic of a document is defined as ‘representative spatio-temporal information’, and documents containing representative spatiotemporal information are defined as ‘representative spatio-temporal documents’. We proposed a character-level Convolution Neuron Network (CNN)-based document classifier to classify representative spatio-temporal documents. To train the proposed CNN model, 7400 training data were constructed for representative spatio-temporal documents. The experimental results show that the proposed CNN model outperforms traditional machine learning classifiers and existing CNN-based classifiers.

**Keywords:** convolution neural network; spatio-temporal document; document classification; big text data



**Citation:** Kim, B.; Yang, Y.; Park, J.S.; Jang, H.-J. A Convolution Neural Network-Based Representative Spatio-Temporal Documents Classification for Big Text Data. *Appl. Sci.* **2022**, *12*, 3843. <https://doi.org/10.3390/app12083843>

Academic Editors: Wei Wang and Ka Lok Man

Received: 20 January 2022

Accepted: 7 April 2022

Published: 11 April 2022

**Publisher’s Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Since social media-based data or online media data is composed of natural language, it has a much larger and more complex structure than existing transaction data [1,2]. Recently, the media distributes news articles online in order to quickly deliver news to consumers, online news articles can identify current social trends and behavioral patterns of members of society [3]. The social trend analysis technology for content published in online media has the advantage of being less expensive and faster than the analysis by existing expert groups. Therefore, research to detect and monitor current major issues by analyzing unstructured text information from social media or online news posts and extracting useful knowledge is being actively conducted.

For social trend analysis, it is important to identify event sentences from text documents such as social media or online news articles [4]. The event sentence refers to a sentence in which specific content about a specific topic, i.e., who, where, when, what, what, etc. is expressed. The temporal and spatial information included in news articles is used to detect the early onset of disease and to determine the time and location of disease outbreaks [5]. The temporal and spatial information presented in online news articles plays a decisively important role in understanding social trends.

Existing research to detect spatial and temporal information from text focuses on how accurately all temporal and spatial information contained within a document is extracted [6–8]. A document can contain many pieces of information about time and space.

In this study, among various spatial and temporal information included in a document, temporal and spatial information describing the core topic of the document is defined as ‘representative spatio-temporal information’. The document including representative spatio-temporal information is defined as a ‘representative spatiotemporal document’. If not only representative spatio-temporal information but also a large number of general spatio-temporal information are extracted from one document, the accuracy of core event analysis based on spatio-temporal information can be lowered. In order to increase the accuracy of core event analysis through artificial intelligence, it is necessary to remove unnecessary spatio-temporal information from one document and extract only the representative spatio-temporal information that accurately describes the core event in the document. Since extracting representative spatio-temporal information from a single document is a high-cost task, it is difficult to treat all documents from big data such as social media-based data or online news articles as analysis targets. Therefore, in order to efficiently analyze core events through representative spatio-temporal information, it is important to select documents from which representative spatio-temporal information is extracted.

Research using machine learning (Naïve Bayes [9,10], SVM [11,12] and Random Forest [13,14], etc.) in automatic document classification problems have been conducted so far. Recently, as deep learning-based Convolution Neuron Network (CNN) has been used for document classification, the performance of automatic document classification has been greatly improved [15]. CNN started to attract attention in the field of artificial intelligence as it showed excellent performance in image classification or object detection in the early days [16–18]. Classification technology using CNN has expanded its field of application from images to texts [19]. Recently, document classification using CNN is characterized as an area that classifies documents (patent documents [20], contracts [21], infectious disease documents [22], etc.) of a specific domain.

In this paper, we propose a character-level CNN-based representative spatio-temporal document classification model. First, we built 7400 learning data from online news articles provided by the National Institute of the Korean Language [23]. We developed a character-level CNN-based document classifier (a.k.a. RepSTDoc\_ConvNet) that can classify representative spatio-temporal documents. RepSTDoc\_ConvNet has a deeper CNN layer and a fully-connected layer than the existing CNN-based document classification model. In order to prove the performance of the proposed CNN model, we compared RepSTDoc\_ConvNet with three baseline machine learning classifiers (Gaussian Naïve Bayes, linear SVM, and random forest) and three deep learning-based models (ConvNet, DocClass\_ConvNet [22] and DocClass\_ConvNet\_Mod).

The final goal of our study is to extract representative spatio-temporal information from a large amount of documents. In order to extract representative spatio-temporal information, it is first necessary to classify representative spatio-temporal documents having representative spatio-temporal information in a large number of documents. This paper corresponds to the stage of classification of representative spatio-temporal documents. Through the representative spatio-temporal information, it can be used for natural disaster detection and analysis of factors (events such as urban planning, building construction, traffic control, and store opening) influencing business district analysis.

Our main contributions are summarized as follows.

- We defined a novel problem of classifying representative spatio-temporal documents containing spatio-temporal information describing the core topic of a document.
- We developed 7400 learning data for representative spatio-temporal documents.
- We proposed a character-level CNN-based document classifier to classify representative spatio-temporal documents.
- The proposed RepSTDoc\_ConvNet outperforms traditional machine learning classifiers, achieving the F1 score of 61.2%.

The rest of the paper is organized as follows. Section 2 presents the literature review. In Section 3, we define the research problem. Section 4 is the proposed CNN-based document

classifier model. In Section 5, we provide the experimental results and discuss the detailed implications along with their results. Section 6 presents the conclusion.

## 2. Literature Reviews

### 2.1. Traditional Machine Learning-Based Document Classification

The study of classifying documents using machine learning rather than reading documents by humans and classifying them into a given class has been conducted using traditional machine learning. Among the various document classifications, the field of detecting whether or not spam is spam was treated as an initial document classification problem. The most common machine learning algorithms used to detect spam emails are Gaussian Naive Bayes, Support Vector Machines (SVMs), and Neural Networks. Gaussian Naive Bayes (GNB) is one of the earliest document classification algorithms applied to spam filtering because it has low false positives and simple processing [9,10]. GNB uses a conditional probability function combined with a simple bag-of-words feature to determine the overall probability of whether a given email is spam or not. First, stop words are deleted from the message, and the message is split into individual words. In all messages in the data set, the total frequency of occurrence for the entire list of words is calculated. A threshold is applied to delete the least frequent words and complete the unique vocabulary of the data. The spam or non-spam label is then used to calculate the probability of each word being included in the spam message. Finally, the probability that the message is spam is calculated by combining the spam probability of each word in the message. Mitra et al. [24] present a least-squares support vector machine (LS-SVM) that classifies noisy document titles into various predetermined categories. Random Forest (RF) classifiers are suitable for text classification on high-dimensional noise data. Islam et al. [25] proposed a dynamic ensemble selection method to improve the performance of a random forest classifier in text classification.

### 2.2. Deep Learning-Based Document Classification

Deep learning uses multi-layered artificial neural networks and learns useful features directly from data. Deep learning is changing the paradigm of machine learning research, showing remarkable performance gains in many areas of computer vision. Deep learning technology has been applied to computer vision since 1989, and Yann LeCun [26] proposed a Convolutional Neural Network that divides an image into several local regions and shares weights for character recognition in an automatic postal classification system. CNN learns features of input data using tensors as input, passes the data through a layer of neurons that classifies the data into multiple stages, and computes the weights to pass as input to the next layer. The main components that make CNN different from neural networks are three layers (convolutional layer, pooling layer, and fully connected layer). The convolutional layer convolves the multidimensional features of the input tensor and outputs a reduced vectorization to pass to the pooling layer. In the max-pooling layer, we extract the maxima from each neuron cluster in the previous layer, reducing the dimensionality while retaining important information from the convolution. The final fully connected layer connects the final node to each specified output class. Recently, in the field of computer vision, a Recurrent Neural Network (RNN) is being used for image and video description generation, handwriting recognition, and text or sound translation functions in images or videos [27].

Deep learning is being actively applied not only to computer vision but also to text classification which identifies what kind of category the input text belongs to. Word2Vec is used to transform the text into tensors or vectorized representations for processing in CNNs. CNN showed higher performance in spam classification than traditional machine learning methods. Huang [28] proposed a CNN (Convolutional Neural Network) model for Chinese SMS (Short Message Service) spam detection. This study also discusses the influence of hyper-parameters on CNN models and proposes optimal combinations of hyper-parameters. Liu et al. [29] proposed a modified deep CNN model for email sentiment classification. Mutabazi et al. [30] provided reviews of various medical text question-

answering systems using deep learning. Kim et al. [22] developed a document classification model related to infectious diseases using deep learning. A document classification model was constructed using two deep learning algorithms (ConvNet and BiLSTM) and two classification methods, DocClass and SenClass. Given a specific text extraction system, it was shown to be compatible with the classification performance of human experts. It has shown the potential of using deep learning to identify epidemic outbreaks.

Table 1 presents the summary of methods for text classification.

**Table 1.** Summary of methods for text classification.

Methods.	Technique
Gaussian Naive Bayes [9,10]	Gaussian Naive Bayes is used for text classification based on Bayes theorem under a normal distribution with sample mean and sample variance.
Linear SVM [11,12,24]	When a set of data belonging to one of two categories is given, SVMs are powerful machine learning supervised learning models that can be used for classification tasks.
Random Forest [13,14,25]	Random forest is an ensemble method for learning multiple decision trees. Random forests are being used for various problems such as detection, classification, and regression.
ConvNet [15]	CNN is a type of multi-layer feed-forward artificial neural network. It is a deep neural network technology that can process regional features of data by applying filtering techniques to artificial neural networks.

### 3. Problem

In this section, we first define several concepts as well as the problem of representative spatio-temporal documents.

**Subject of the document.** Let  $D = \{d_1, \dots, d_n\}$  be a set of documents. Each document has a core subject, which is the message the author wants to convey to the reader. For example, consider a news article reporting the damage of a typhoon that occurred on Jeju Island, South Korea on September 7.  $d_i.subject = \{\text{'typhoon damage'}\}$  denotes the subject of  $d_i$  is about the damage caused by the typhoon that occurred on Jeju Island on September 7.

**Spatio-temporal word.**  $d_i = \{s_1, \dots, s_m\}$  is a sequence of sentences and  $s_i = \{w_1, \dots, w_l\}$  is a sequence of words. Among the words contained in a document, there are words for a specific time and place where an event occurred.  $w_i.time = \{\text{'September 7'}\}$  denotes that an event occurred on September 7.  $w_j.place = \{\text{'Jeju Island'}\}$  denotes that the place where an event occurred is Jeju Island.

**Representativeness of spatio-temporal word.** Several spatio-temporal words can exist in one document. Some of the spatio-temporal words are related to the subject of the document, and some are not. Among spatio-temporal words, we consider the words most relevant to the subject of a document as 'representative spatio-temporal words'. We denote a representative spatio-temporal word,  $w_i.representativeness = true$ .

**Representative spatio-temporal document.** We define a document containing both a representative spatial word and a representative temporal word among words included in one document as a representative spatio-temporal document.

## 4. Materials and Methods

### 4.1. Datasets

In this study, learning data for the classification of representative spatio-temporal documents were constructed using the published Korean corpus. The National Institute of Korean Language [23] discloses various data in Korean. In this study, a newspaper corpus provided by the National Institute of the Korean Language for research purposes was used to construct learning data for representative spatio-temporal documents. The newspaper corpus provided by the National Institute of the Korean Language is a collection of newspaper articles produced for 10 years from 2009 to 2018 with a total of 3,536,491 articles.

The corpus consists of a total of 363 files, with a total size of 15.6 GB. The original file is composed of JSON (UTF-8 encoding). Raw data contains article content in the document tag. One article consists of a metadata tag indicating the metadata of the article (title, article name, newspaper company, publication date, and subject) and a paragraph tag indicating the article body. In the paragraph, the article body is divided into paragraphs and composed of form tags.

#### 4.2. Data Preprocessing

We constructed representative spatio-temporal information learning data for 7400 articles out of 3,536,491 articles. Eight workers read the content of the news article and judge whether the article has representative spatio-temporal information. In order to improve the performance of artificial intelligence systems, the quality of training data is important. In order to maintain the consistency of data quality among workers, we cross-checked each other’s work results three times.

#### 4.3. Deep Learning Model

Determining whether or not a news article is a representative spatiotemporal document is a binary classification problem. We used a deep learning neural network model, a character-level convolutional neural network (CNN) called ConvNet [15]. In general, ConvNet divides sentences/paragraphs/documents into word unit tokens when text classification is performed. However, Zhang et al. [15] argue that by using the character (alphabetic) unit instead of the word unit token, a good enough performance for the Natural Language Processing (NLP) task can be achieved without using the word unit. An attempt to use tokening as a character-level unit was first presented in this paper. We also used an embedding matrix created by tokenizing the text in character units as shown in Figure 1.

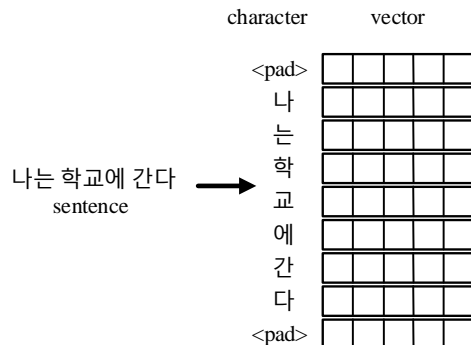


Figure 1. Character-level embedding. (‘나는 학교에 간다’ in Korean means ‘I go to school’ in English).

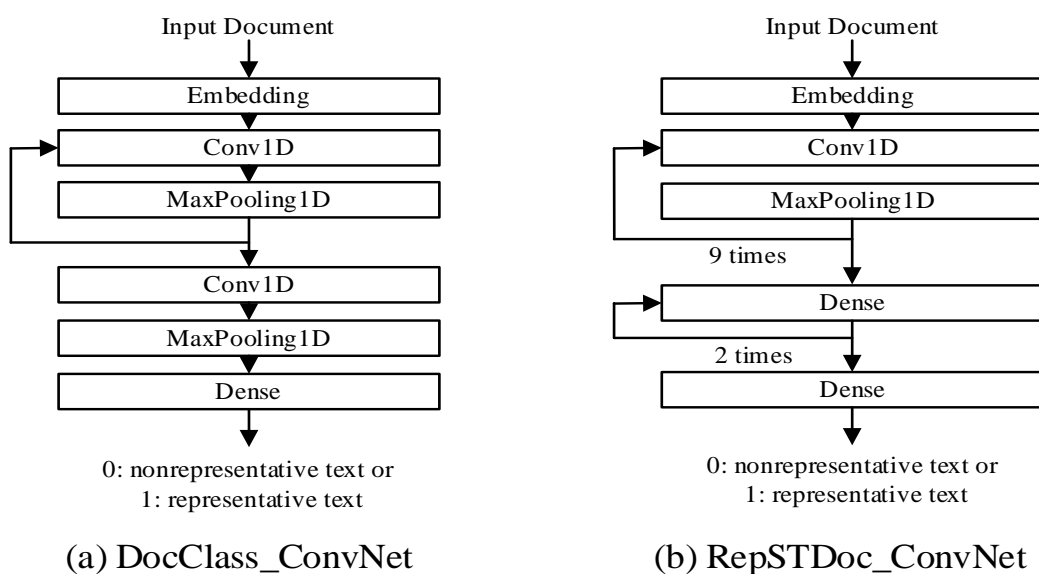
ConvNet treats each document as a series of characters and is passed to 6 convolutional and max-pooling layers and 3 fully connected layers to determine the probability that a document belongs to a positive class. Because this model does not require pre-trained embedded words, it learns quickly and with reasonable performance compared to word-level models.

We developed a character-level CNN-based document classifier to classify representative spatio-temporal documents, RepSTDoc\_ConvNet using the entire document as input. We used the layers of the CNN model, DocClass\_ConvNet, in [22] as our baseline. Figure 2 shows a comparison of the two models.

ConvNet has both 9 layers deep with 6 convolutional layers and 3 fully-connected layers. DocClass\_ConvNet has both 6 layers deep with 4 convolutional layers and 2 fully-connected layers. RepSTDoc\_ConvNet has both 12 layers deep with 9 convolutional layers and 3 fully-connected layers.

In order to train a ConvNet model, we need to keep documents of various lengths constant. Considering the hardware memory constraint and the length distribution of the training data, the number of characters in the document was set to 4700 in ConvNet. Long text is truncated and short text is padded.





**Figure 2.** A comparison of DocClass\_ConvNet [22] and RepSTDoc ConvNet.

## 5. Result and Discussion

In this section, we present comprehensive experimental results of the deep learning model. The purpose of this paper is to develop a classifier for representative spatio-temporal documents based on deep learning. To evaluate the performance of a proposed deep learning-based classifier, we first evaluated the performance of three traditional machine learning algorithms: Gaussian Naïve Bayes, Linear SVM, and Random Forest. For performance comparison with our CNN model (RepSTDoc\_ConvNet), we also evaluated the performance of DocClass\_ConvNet, an existing CNN-based document binary classifier, and DocClass\_ConvNet\_Mod, which adjusted hyper-parameters in the DocClass\_ConvNet model to fit our dataset.

To confirm that our CNN model works properly, we pre-tested the performance of binary classification using the benchmark spam dataset from the UCI Repository [31]. The spam dataset contained 5572 messages in English. This spam dataset was fed to our proposed CNN model and the experimental results were as follows: accuracy (0.982), precision (0.962), recall (0.916), and F1-score (0.938). This result is not significantly different from that of the recently published CNN model [32].

All experiments were carried out on conducted on a GeForce RTX 2080 Ti 11GB GPU and an Intel(R) Xeon CPU with 64 GB memory.

### 5.1. Performance Evaluation

For the experiment, we divided the collected data into training (60%), validation (20%), and test data (20%) as shown in Table 2. Target data were distributed to each data about 25.23%. The training data was used to train the model, the validation data was used to select the best performing model in the training process, and the test set was used to evaluate the performance of the finally selected model.

**Table 2.** Statistics of training, validation, and test data.

Split Data	Count	Non RepSTDoc	RepSTDoc	Ratio
Training	4440	3319	1121	25.25%
Validation	1480	1107	373	25.20%
Test	1480	1107	373	25.20%
Total	7400	5533	1867	25.23%

### 5.2. Hyper-Parameter Tuning

CNN consists of several hyper-parameters such as kernel size, batch size, dropout rate, learning rate, pooling window size, pooling type, activation function, number of neurons in a density layer, and optimization function, etc. We found the most suitable parameter values for the proposed model by manually adjusting the values of each parameter. We found the optimal parameter values by using the learning curves for accuracy and loss of training data and validation data for every experiment. We set up the experimental environment with various parameters, the parameters used in the experiment are summarized in Table 3, and the parameter values with the highest performance are shown in bold. During the training process of the CNN model, we trained our CNN model with up to 1000 epochs and early stopping patience = 220.

**Table 3.** Hyper-parameters for the experiments.

Hyper-Parameter	Values
Kernel size	2, 3, 4, 5, 6, 7
Feature maps	32, 64, 128, <b>256</b> , 512
Pooling window size	3, 4, 5
Pooling type	Max pooling
Activation function	ReLu
Dense layer neurons	<b>100</b> , 300
Dropout rate	0.3, 0.4, 0.5, <b>0.6</b> , 0.7, 0.8
Batch size	16, 32, <b>64</b> , 128, 256
Learning rate	0.1, 0.01, 0.001, 0.0001, <b>0.00001</b> , 0.000001
Optimizer	<b>Adam</b> , RMSprop

Overfitting deep learning models makes it difficult to trust their predictive performance on new data. Therefore, training should be stopped when the loss in the validation data is no longer reduced during the training phase. Early stopping is one of the regularization techniques that makes neural networks avoid overfitting [33]. We can use the EarlyStopping callback to terminate the model early when the performance index of the model does not improve during the set epoch. Through a combination of EarlyStopping and ModelCheckpoint callbacks, it is possible to trigger an early shutdown for non-improving training and resume training by reloading the best model from ModelCheckpoint. Both training loss and validation loss decrease until overfitting occur, but when overfitting occurs, training loss decreases while validation loss increases. Thus, we set the monitor option of EarlyStopping callback to stop training when the validation loss increases.

### 5.3. Experimental Results

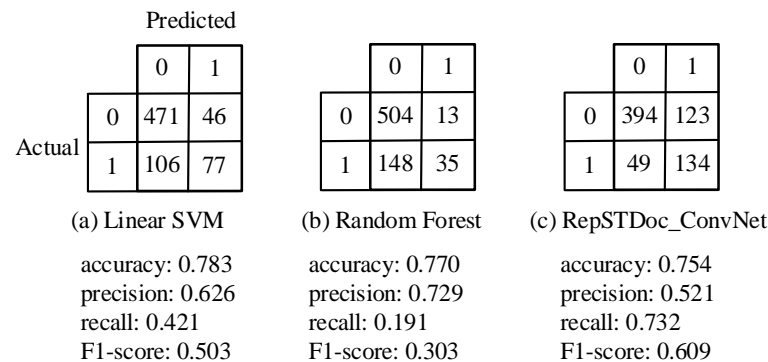
We compared the RepSTDoc\_ConvNet with three baseline machine learning classifiers (Gaussian naïve Bayes, linear SVM, and random forest) and three deep learning models (ConvNet, DocClass\_ConvNet, and DocClass\_ConvNet\_Mod). DocClass\_ConvNet is a model in which the CNN layer and hyper-parameters presented in the study are identical. DocClass\_ConvNet\_Mod is a model that optimizes the hyper-parameter values according to the experimental data while maintaining the same CNN layer of DocClass\_ConvNet. Deep learning includes the process of randomly setting weight values during model training. Therefore, to compensate for such randomness, the average performance was measured after performing each experiment 10 times. The experimental results are presented in Table 4.

The accuracy of machine learning algorithms to classify representative spatio-temporal documents was derived from a minimum of 0.74 to a maximum of 0.79. This accuracy is far below the performance of machine learning that deals with general document classification problems. The CNN layer used in this paper derives relatively high performance in the spam classification problem. From these results, it can be seen that classifying representative spatio-temporal documents is a difficult problem.

**Table 4.** Comparison of evaluation based on the precision, recall, F1 score, and accuracy.

Machine Learning	Precision	Recall	F1 score	Accuracy
Gaussian Naïve Bayes	0.562	0.224	0.320	0.751
Linear SVM	0.626	0.421	0.503	0.783
Random Forest	<b>0.729</b>	0.191	0.303	0.770
ConvNet	0.525	0.591	0.556	0.762
DocClass_ConvNet	0.511	0.453	0.480	0.744
DocClass_ConvNet_Mod	0.614	0.496	0.548	<b>0.794</b>
RepSTDoc_ConvNet	0.552	<b>0.673</b>	<b>0.612</b>	0.785

Random Forest showed the highest precision with 0.729 and DocClass\_ConvNet\_Mod showed the highest accuracy with 0.794. RepSTDoc\_ConvNet showed the highest recall and F1-score with 0.673 and 0.612, respectively. In terms of accuracy, DocClass\_ConvNet\_Mod seems to have the highest performance with 0.794. However, considering the confusion matrix, it does not seem appropriate to evaluate the performance of machine learning only with accuracy in the problem of classifying representative spatio-temporal documents. Figure 3 shows three confusion matrixes of Linear SVM, Random Forest, and RepSTDoc\_ConvNet.



**Figure 3.** Comparison of confusion matrixes (a) Linear SVM, (b) Random Forest, and (c) RepSTDoc\_ConvNet. ‘0’ means the nonrepresentative spatio-temporal document and ‘1’ means the representative spatio-temporal document.

In the validation data used to evaluate the proposed CNN model, the proportion of representative spatio-temporal documents (RepSTDoc) is only 25.20%. Therefore, even when the model is not trained at all, the accuracy is 74.80%. In this case, high accuracy is maintained even if the number of documents predicted by the model with RepSTDoc is small. In Figure 3a, Linear SVM classified 123 documents (46 false positives, 77 true positive) as RepSTDoc. Even if the model training is not done properly, the high true negative value (471) results in high accuracy. A random forest with the second-highest accuracy is also similar to Linear SVM. In the random forest, the accuracy is 0.770 even though there are few documents classified by RepSTDoc (48) because the model is hardly trained. The fact that the number of documents predicted as RepSTDoc is small because the model is not trained can be confirmed by the small recall value (0.191). In Figure 3c, RepSTDoc\_ConvNet classified 257 documents (123 false positives, 134 true positive) as RepSTDoc. In RepSTDoc\_ConvNet, as the value of true positive increased, the value of false-positive also increased. The fact that the model classified many documents as RepSTDoc can be seen from the high value of recall (0.609). This phenomenon occurs because the number of positive and false documents in the data is imbalanced. Therefore, in order to accurately evaluate the performance of the model, the F1-score, which considers both precision and recall, should be used as a measure. In terms of the F1-score, RepSTDoc\_ConvNet yields the highest performance with 0.609.

We measured the classification accuracy of human workers on 1400 learning data to verify the challenge of the representative spatio-temporal document classification prob-



lem. The 1400 learning data consists of 359 representative spatio-temporal documents and 1041 non-representative spatio-temporal documents. Four workers who participated in building learning data classified representative spatio-temporal documents for 1400 learning data. For each learning data, the number of workers who judged actual representative spatio-temporal documents as representative spatio-temporal documents (True Positive: TP) and the number of workers who judged non-representative spatio-temporal documents (False Negative: FN) were calculated.

For one actual representative spatio-temporal document, the ratio was calculated by dividing the number of all four people judged as TP, the number of three or more judged as TP, the number of two or more judged as TP, and the number of one or more judged as TP in Table 5. For each of the 359 representative spatiotemporal documents, the number of documents judged as TP by all 4 people was 189 (52.64%), the number of documents judged as TP by 3 or more people 251 (69.92%), and the number of documents judged as TP by 2 or more people was 310 (89.35%), the number of documents judged as TP by 1 or more people was 332 (92.48%).

**Table 5.** The ratio and count of actual representative spatio-temporal documents to be judged as representative spatio-temporal documents according to workers.

	4	>3	>2	>1
ratio	52.64%	69.91%	86.35%	92.47%
count	189	251	310	332

For one actual nonrepresentative spatio-temporal document, the ratio was also calculated by dividing the number of all 4 people judged as FN, the number of 3 or more people judged as FN, the number of 2 or more people judged as FN, and the number of 1 or more people judged as FN in Table 6. For each of the 1041 nonrepresentative spatio-temporal documents, the number of documents judged as FN by all 4 people was 5 (0.48%), the number of documents judged as FN by 3 or more people was 24 (2.31%), and the number of documents judge as FN by 2 or more people (6.34%), and the number of documents judged as FN by more than one person was 135 (12.97%).

**Table 6.** The ratio and count of actual nonrepresentative spatio-temporal documents to be judged as nonrepresentative spatio-temporal documents according to workers.

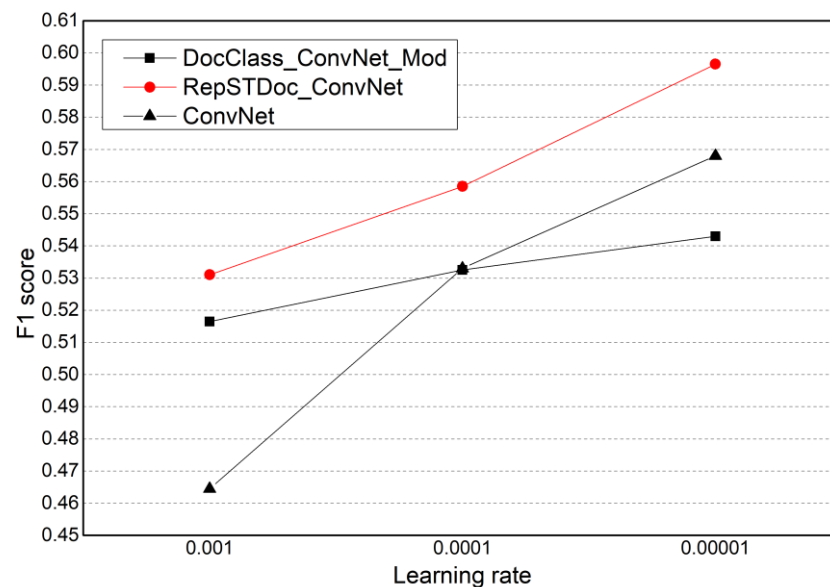
	4	>3	>2	>1
ratio	0.48%	2.30%	6.34%	12.96%
count	5	24	66	135

First of all, we describe the challenge of the representative spatio-temporal document classification problem through the ratio of documents in which at least three people, more than half of the judges, judged the actual representative spatio-temporal document as the representative spatio-temporal document. About 70% of the three or more people judged the actual representative spatio-temporal document as TP, and the ratio of all four people who judged it as TP was only about 53%, confirming that it is difficult for humans to classify representative spatio-temporal documents from large documents.

#### 5.4. Effect of Learning Rate

The learning rate refers to the amount by which the weights are updated during model training and determines how quickly the model adapts to the problem. Larger learning rates converge more quickly to suboptimal solutions, while lower learning rates can result in early intervening learning. One of the important hyper-parameters that must be appropriately selected in deep learning neural network model training is the learning rate. We experimented with the effect of learning rate [0.1, 0.01, 0.001, 0.0001, 0.00001, 0.000001] on performance.

Figure 4 shows the effect of the learning rate for ConvNet, DocClass\_ConvNet\_Mod, and RepSTDoc\_ConvNet. The learning rate at which no training was performed in each model was not shown on the graph (learning rate: 0.1, 0.01, and 0.000001). In the section where the model is trained, the F1-score tends to increase as the learning rate decreases. There is a large difference in performance according to the learning rate in each model. In the representative spatio-temporal learning data used in this study, the learning rate shows the highest performance at 0.00001.



**Figure 4.** The effect of learning rate.

### 5.5. Effect of Batch Size

Most of the training of deep learning models is based on mini-batch stochastic gradient descent (SGD). At this time, the batch size is one of the important hyper-parameters when training the actual model. Various studies are being conducted regarding the effect of the batch size on model training. Although it has not been clearly identified yet, it is experimentally observed in several studies that the use of a small batch size has a positive effect on generalization performance. We experimented with the effect of learning rate [16, 32, 64, 128, and 256] on performance.

Figure 5 shows the effect of batch size for ConvNet, DocClass\_ConvNet\_Mod, and RepSTDoc\_ConvNet. In the representative spatio-temporal learning data used in this study, there was no consistent performance variability across models. RepSTDoc\_ConvNet shows a tendency to improve performance as the batch size increases in the model training section [32, 64, 128, and 256]. However, in DocClass\_ConvNet\_Mod, the variation of performance according to the batch size was not consistent. Although this result cannot be generalized, the batch size may not affect the performance of the model depending on the complexity of the CNN layer and the characteristics of the data.

### 5.6. Time Efficiency

The numbers of weights are 1,410,609, 1,446,261, and 5,083,129 in DocClass\_ConvNet\_Mod, ConvNet and RepSTDoc\_ConvNet respectively. The overall algorithm time is affected by the complexity of the neural network. This is because the amount of computation increases as the number of weights in the network increases. Table 7 shows the time efficiencies for the three algorithms.

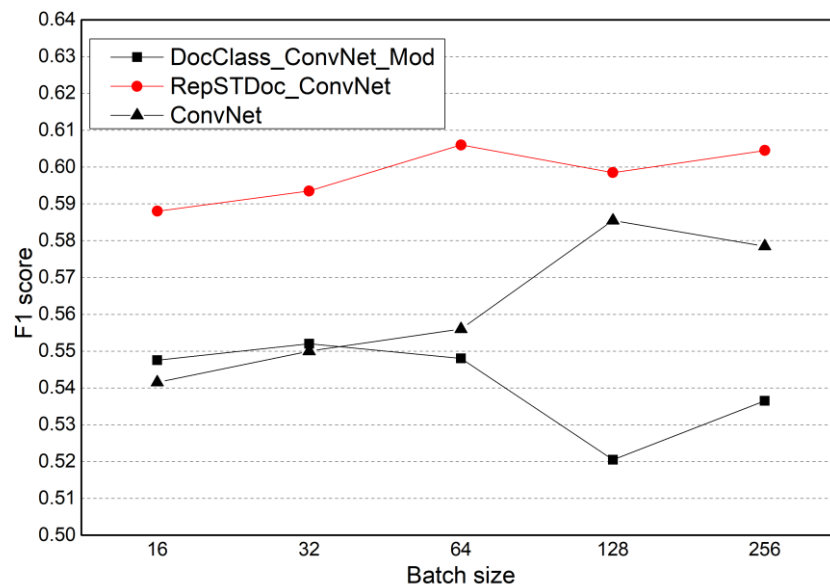


Figure 5. The effect of batch size.

Table 7. The comparison of the time.

Methods	Avg. Epoch	Avg. Time (s)	Avg. Time per Epoch (s)
ConvNet	405	387	0.912
DocClass_ConvNet_Mod	398	332	0.849
RepSTDoc_ConvNet	687	903	1.313

5.7. Data Distribution Rate

We also investigated the performance difference according to the change in the distribution ratio of training, validation, and test data. The ratio of training data was set while keeping the ratio of validation data and test data the same. The distribution ratio used in the experiment is as follows: training, validation, and test data are 4:3:3, 6:2:2, and 8:1:1 respectively. Figure 6 shows the highest performance with a 6:2:2 distribution ratio. There is not much difference in the performance of each model according to the distribution ratio.

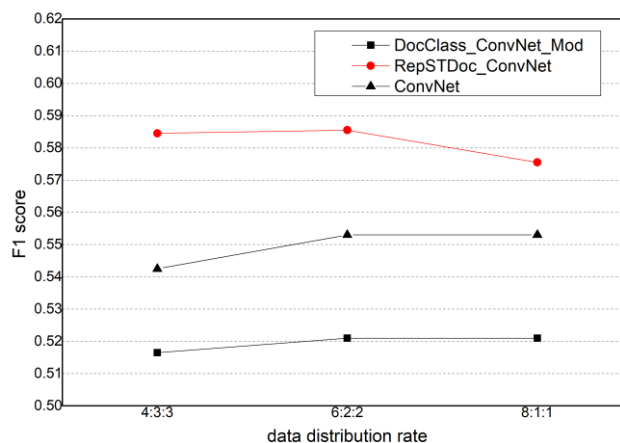
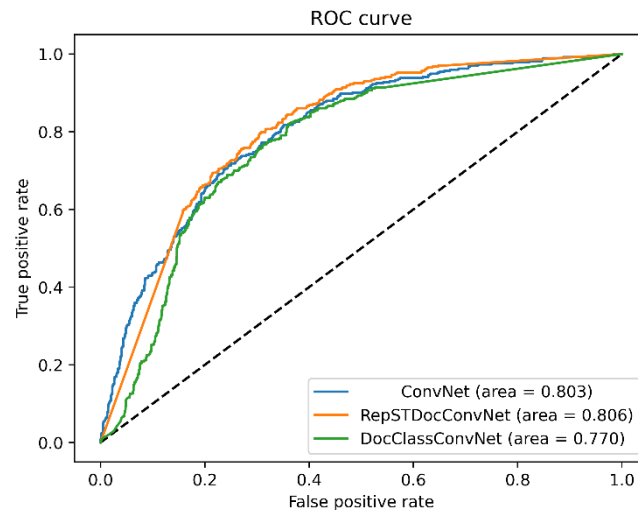


Figure 6. The effect of distribution rate.

5.8. Receiver Operating Characteristic

The Receiver Operating Characteristic (ROC) curve shows the performance of the binary classifier for various thresholds. Figure 7 shows the corresponding ROC curves when using ConvNet, DocClass\_ConvNet\_Mod, and RepSTDoc\_ConvNet. ConvNet outperformed the other models in the lower-left corner. However, in the section where the

false positive rate is greater than 0.2, RepSTDoc\_ConvNet was superior to other models. RepSTDoc\_ConvNet was found to have the best performance for classifying representative spatiotemporal documents.



**Figure 7.** Receiver operating characteristic (ROC) curves of models classifying representative spatio-temporal documents using ConvNet, DocClass\_ConvNet\_Mod and RepSTDoc\_ConvNet.

## 6. Conclusions

The purpose of this paper is to develop a CNN-based representative spatio-temporal document classification model. Because the representative spatio-temporal document is a novel concept, we defined a representative spatio-temporal document as documents containing spatio-temporal information describing the core topic of a document. We built 7400 learning data to train a CNN-based representative spatio-temporal document classifier and developed a character-level CNN-based document classifier to classify representative spatio-temporal documents. To evaluate the performance of RepSTDoc\_ConvNet, we evaluated the performance of three traditional machine learning algorithms: Gaussian Naïve Bayes, Linear SVM, and Random Forest. For performance comparison with our RepSTDoc\_ConvNet, we also evaluated the performance of ConvNet, DocClass\_ConvNet, and DocClass\_ConvNet\_Mod. The experimental results show that RepSTDoc\_ConvNet outperforms traditional machine learning classifiers and existing CNN-based classifiers.

A limitation of the work is that RepSTDoc\_ConvNet still has lower performance compared to general document classifiers. It is necessary to diversify the features of the input data as it shows that classifying representative spatio-temporal documents is a difficult problem. In order to further improve the performance of the representative spatio-temporal document classifier, it is necessary to find a way to lower the false positive value by finding the characteristic that distinguishes the general spatio-temporal document from the representative spatio-temporal document.

**Author Contributions:** Conceptualization, H.-J.J. and B.K.; methodology, B.K.; software, B.K.; validation, H.-J.J.; investigation, Y.Y.; data curation, Y.Y. and B.K.; writing—original draft preparation, H.-J.J. and B.K.; writing—review and editing, J.S.P.; visualization, J.S.P.; supervision, B.K.; project administration, B.K.; funding acquisition, B.K. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Korean Government (MSIT) (No. 2021R1F1A1049387) and by industry-academic Cooperation R&D program funded by LX Spatial Information Research Institute (LXSIRI, Republic of Korea) [Project Name: A Study on the Establishment of Service Pipe Database for Safety Management of Underground Space/Project Number: 2021-502]. This result was

supported by the “Regional Innovation Strategy (RIS)” through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (MOE) (1345341782).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Written informed consent has been obtained from the patient(s) to publish this paper.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Chew, A.M.K.; Gunasekeran, D.V. Social Media Big Data: The Good, The Bad, and the Ugly (Un)truths. *Front. Big Data* **2021**, *4*, 6. [CrossRef] [PubMed]
2. Nurdin, N. Research in Online Space: The Use of Social Media for Research Setting. *J. Inf. Syst.* **2017**, *13*, 67–77. [CrossRef]
3. Kim, M.; Newth, D.; Christen, P. Trends of news diffusion in social media based on crowd phenomena. In Proceedings of the Companion Publication of the 23rd International Conference on World Wide Web Companion, Seoul, Korea, 7–14 April 2014; International World Wide Web Conference Steering Committee: Geneva, Switzerland, 2014; pp. 753–758. [CrossRef]
4. Naughton, M.; Stokes, N.; Carthy, J. Sentence-level event classification in unstructured texts. *Inf. Retr.* **2010**, *13*, 132–156. [CrossRef]
5. Lan, R.; Adelfio, M.D.; Samet, H. Spatio-temporal disease tracking using news articles. In Proceedings of the HealthGIS'14: 3rd ACM SIGSPATIAL International Workshop on the Use of GIS in Public Health, Dallas, TX, USA, 4–7 November 2014; pp. 31–38. [CrossRef]
6. Badia, A.; Ravishankar, J.; Muezzinoglu, T. Text Extraction of Spatial and Temporal Information. In Proceedings of the 2007 IEEE Intelligence and Security Informatics, New Brunswick, NJ, USA, 23–24 May 2007. [CrossRef]
7. Lim, C.-G.; Jeong, Y.-S.; Choi, H.-J. Survey of Temporal Information Extraction. *J. Inf. Processing Syst.* **2019**, *15*, 931–956. [CrossRef]
8. Ferial, A.; Kholadi, M.-K. Automatic Extraction of Spatio-Temporal Information from Arabic Text Documents. *Int. J. Comput. Sci. Inf. Technol.* **2015**, *7*, 97–107. [CrossRef]
9. Chen, J.; Huang, H.; Tian, S.; Qu, Y. Feature selection for text classification with Naïve Bayes. *Expert Syst. Appl.* **2009**, *36*, 5432–5435. [CrossRef]
10. Pavel, H. How to Build and Apply Naive Bayes Classification for Spam Filtering. *Medium, Towards Data Science*, 31 January 2020.
11. Bedi, G. Simple Guide to Text Classification (NLP) Using SVM and Naive Bayes with Python. *Medium*, 13 July 2020.
12. Ray, S. SVM: Support Vector Machine Algorithm in Machine Learning. *Analytics Vidhya*, 23 December 2020.
13. Liparas, D.; HaCohen-Kerner, Y.; Moutzidou, A.; Vrochidis, S.; Kompatsiaris, I. News Articles Classification Using Random Forests and Weighted Multimodal Features. In *Multidisciplinary Information Retrieval*; Springer: Cham, Switzerland, 2014; pp. 63–75. [CrossRef]
14. Sharma, S.K.; Sharma, N.K.; Potter, P.P. Fusion Approach for Document Classification using Random Forest and SVM. In Proceedings of the 9th International Conference System Modeling and Advancement in Research Trends (SMART), Moradabad, India, 4–5 December 2020. [CrossRef]
15. Zhang, X.; Zhao, J.; Yan, L.C. Character-Level Convolutional Networks for Text Classification. *arXiv* **2015**, arXiv:1509.01626.
16. Bibi, S.; Abbasi, A.; Haq, I.U.; Baik, S.W.; Ullah, A. Digital Image Forgery Detection Using Deep Autoencoder and CNN Features. *Hum. Cent. Comput. Inf. Sci.* **2021**, *11*, 1–17.
17. Song, W.; Zhang, L.; Tian, Y.; Fong, S.; Liu, J.; Gozho, A. CNN-based 3D object classification using Hough space of LiDAR point clouds. *Hum. Cent. Comput. Inf. Sci.* **2020**, *10*, 1–14. [CrossRef]
18. Song, W.; Liu, Z.; Tian, Y.; Fong, S. Pointwise CNN for 3D Object Classification on Point Cloud. *J. Inf. Proc. Syst.* **2021**, *17*, 787–800. [CrossRef]
19. Zeng, Y.; Zhang, R.; Yang, L.; Song, S. Cross-Domain Text Sentiment Classification Method Based on the CNN-BiLSTM-TE Model. *J. Inf. Proc. Syst.* **2021**, *17*, 818–833. [CrossRef]
20. Li, S.; Hu, J.; Cui, Y.; Hu, J. DeepPatent: Patent classification with convolutional neural networks and word embedding. *Scientometrics* **2018**, *117*, 721–744. [CrossRef]
21. Chen, Y.; Dai, H.; Yu, X.; Hu, W.; Xie, Z.; Tan, C. Improving Ponzi Scheme Contract Detection Using Multi-Channel TextCNN and Transformer. *Sensors* **2021**, *21*, 6417. [CrossRef] [PubMed]
22. Kim, M.; Chae, K.; Lee, S.; Jang, H.-J.; Kim, S. Automated Classification of Online Sources for Infectious Disease Occurrences Using Machine-Learning-Based Natural Language Processing Approaches. *Int. J. Environ. Res. Public Health* **2020**, *17*, 9467. [CrossRef] [PubMed]
23. National Institute of Korean Language [Internet]. Available online: <https://www.korean.go.kr> (accessed on 6 April 2022).
24. Mitra, V.; Wang, C.-J.; Banerjee, S. Text classification: A least square support vector machine approach. *Appl. Soft Comput.* **2007**, *7*, 908–914. [CrossRef]
25. Islam, M.Z.; Liu, J.; Li, J.; Liu, L.; Kang, W. A Semantics Aware Random Forest for Text Classification. In Proceedings of the 28th ACM International Conference on Information and Knowledge Management, CIKM'19, Beijing, China, 3–7 November 2019; pp. 1061–1070. [CrossRef]



26. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [[CrossRef](#)] [[PubMed](#)]
27. Zhong, Z.; Gao, Y.; Zheng, Y.; Zheng, B. Efficient Spatio-Temporal Recurrent Neural Network for Video Deblurring. In *Computer Vision—ECCV 2020*; Vedaldi, A., Bischof, H., Brox, T., Frahm, J.M., Eds.; ECCV 2020. Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2020; Volume 12351. [[CrossRef](#)]
28. Huang, T. A CNN Model for SMS Spam Detection. In Proceedings of the 4th International Conference on Mechanical, Control and Computer Engineering (ICMCCE), Hohhot, China, 25–27 October 2019.
29. Liu, S.; Lee, I. Sequence encoding incorporated CNN model for Email document sentiment classification. *Appl. Soft Comput. J.* **2021**, *102*, 107104. [[CrossRef](#)]
30. Mutabazi, E.; Ni, J.; Tang, G.; Cao, W. A Review on Medical Textual Question Answering Systems Based on Deep Learning Approaches. *Appl. Sci.* **2021**, *11*, 5456. [[CrossRef](#)]
31. Almeida, T.A.; Hidalgo, J.M.G.; Yamakami, A. Contributions to the study of sms spam filtering: New collection and results. In Proceedings of the 11th ACM Symposium on Document Engineering, Mountain View, CA, USA, 19–22 September 2011; pp. 259–262.
32. Roy, P.K.; Singh, J.P.; Banerjee, S. Deep learning to filter SMS Spam. *Future Gener. Comp. Syst.* **2019**, *102*, 524–533. [[CrossRef](#)]
33. Goodfellow, I.; Yoshua, B.; Courville, A. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2016.