Multichannel Convolutional Neural Network Model to Improve Compound Emotional Text Classification Performance

Aripin, *Member, IAENG*, Wisnu Agastya, and Solichul Huda

***Abstract*—Natural artificial facial expressions are necessary for the development of interactive virtual communication technologies. One effort that can be made to form compound facial expressions is a combination of several basic emotional classes. Identification of the dominant basic emotional classes in sentences can be done using a multilabel classification process. In this study, we combined the Multichannel Convolution Neural Network model and multilabel techniques to improve classification performance in identifying dominant basic emotional classes. Compound facial expressions are formed by combining these dominant emotional classes. The data used in the classification process is the text of social media posts that have been labeled by analysis from linguists into six categories of emotions: fear, anger, disgust, sadness, happiness, and surprise. The results of the labeling process were validated by psychologists. The experimental results for the 8:2 data portion showed that the CNN Multichannel model with kernel size = [4, 5, 6] and max-pooling layers produced the average value of the highest accuracy of 96% and F1-score of 86% compared to the use of the Simple bidirectional LSTM model with 32 LSTM units, CNN Single channel and other variations of CNN Multichannel. The experimental results show that our proposed method has better performance than some of the CNN models used in previous studies.**

***Index Terms*—Compound Emotional, Convolution Neural Network, Indonesian Text Classification, Multilabel, Multichannel**

I. Introduction

Interactive communication is the transfer of information between the communicator and the communicant directly. The communication includes verbal and non-verbal. Non-verbal communication is conveying messages using body language, one of which is facial expressions [1]. The technology of facial expressions is really needed in the animation industry, interactive games, and virtual technologies. Natural facial expressions can emphasize the information conveyed. In the formation of natural facial expressions, emotions play an important role as one of the components forming facial expressions [2]. Emotions can be determined based on the meaning of sentences communicated orally as well as in writing. The classification process of Indonesian texts can be used to determine the meaning of sentences.

In this study, Indonesian texts were used as datasets for the classification process. Indonesian is the official language used by the Indonesian people to exchange information. In general, the Indonesian sentence structure consists of a subject, predicate, object, and adverb [3]. There are two types of Indonesian sentences based on the number of clauses, namely, single sentences and compound sentences [4]. A single sentence is a sentence consisting of only one clause, while a compound sentence is a sentence consisting of more than one clause separated by a connecting word or punctuation mark. A compound sentence can have two or more clauses containing different information, so one sentence can potentially have more than two interconnected emotions.

Classification is the process of categorizing data into certain categories based on certain features. The classification process requires determining categories (labeling) first. In general, Classification is divided into three types, namely multi-label classification, multi-class classification, and binary classification. In binary classification, data is classified into one class of two existing classes. Whereas multi-class classification, data is classified into one of several classes (more than two classes). Meanwhile, multi-label classification is a data classification process that allows each data to have more than one class. The classification process was applied to Indonesian sentences into emotional classes. Each sentence can be classified into two or more classes of emotions. Therefore, the multi-label classification process is more precisely used to solve the problem.

Classification of text data requires text pre-processing to extract important features. This stage is an important stage that greatly affects the results of the modeling. In this study, text pre-processing included removing unused features and normalizing features. There are many models for the classification process, including Logistic Regression, K-Nearest Neighbors (K-NN), Naïve Bayes, Gradient Boosting, Support Vector Machine (SVM), Random Forest, and Neural Networks.

The Neural networks model is based on the process of weighting features that occur from layer to layer. In the neural network model, various kinds of feature-breaking layers (encoder) can process data features in order, such as Recurrent Neural Networks (RNN), Convolutional Neural Networks (CNN), Gated Recurrent Unit (GRU), Long Short-Term Memory (LSTM), and others. In some cases, text

Manuscript received September 19, 2022; revised June 3, 2023.

This study was funded by the Ministry of Research, Technology, and Higher Education through a competitive grant program of the Higher Education Excellence Applied Research scheme (Grant No. 10/061031/PB/SP2H//AK.04/2022).

Aripin is a PhD of the Biomedical Engineering Department of Dian Nuswantoro University, Indonesia. (Corresponding author, phone: +6281331866866, e-mail: arifin@dsn.dinus.ac.id).

Wisnu Agastya is an undergraduate student at the Informatics Engineering Department of Dian Nuswantoro University, Indonesia. (e-mail: wisnu.aga@gmail.com).

Solichul Huda is a PhD of the Informatics Engineering Department of Dian Nuswantoro University, Indonesia. (e-mail: solichul.huda@dsn.dinus.ac.id).

classification generally use LSTM as an encoder layer to recognize information by order and address the problem of vanishing gradients.

The Convolution Neural Network (CNN) is a popular architecture in addition to Recurrent Neural Networks (RNN) to handle Natural Language Processing (NLP) tasks. CNN performs superbly in sentence modeling [5], semantic decomposition [6], and text classification [7]. The CNN model is effectively used for Natural Language Processing (NLP), including performing text classification tasks [8], [9], [10]. NLP is a discipline related to methods for automatically analyzing and understanding natural language; that is, the language is written or spoken naturally by humans [11]. The CNN models typically used for NLP are one-dimensional CNN (1D CNN) and word embedding layers. This model can be expanded by using multichannel CNN with different kernel sizes to read source documents.

CNN architecture for text classification includes a word embedding layer, a max-pooling layer, a one-dimensional convolution layer, a fully connected layer (dense layer), and an output layer [12]. Broadly speaking, the text classification process using the CNN model is a) initial text processing and vectorization is used as a vector input in the embedding layer, which results in the embedding matrix, b) on the convolutional layer, the convolutional process is carried out for each column in the embedding matrix that produces feature maps, c) all feature maps are passed in the max-pooling layer to eliminate each feature in the maximum distribution, d) each feature map is passed in each neuron fully connected layer using ReLU (Rectified Linear Unit) activation function.

CNN's Multi-Channel model architecture is a refinement of the traditional CNN model [13]. CNN's Multi-Channel model has the advantage of maintaining spatial information on the features of the extracted and classified objects so that it can increase the level of accuracy and optimize the algorithm [14].

Based on these studies, based on previous studies, this proposed study aims to improve the CNN model's performance by using the multilabel and multichannel layers of CNN to classify sentences of compound structure Indonesian to produce classes of compound emotions. The CNN model architecture used in this study is a *neural network* model with a *convolutional* layer architecture as an encoder layer.

II. EMOTIONS AND DATA REPRESENTATION

A. *Linguistic studies of Indonesian emotional words*

Emotion is a feeling experienced by a person when facing a certain situation. Semantically, the person who experiences that emotion is referred to as the perpetrator. The word emotion is part of an individual's understanding of his emotional experience. In colloquial language, the experience of emotions is simply interpreted as the status of feelings labeled with the words anger, hate, or happy [15]. Emotional vocabulary is verbal labels used to describe and express the emotional state experienced by an individual. Indonesian has a wide variety of emotional vocabulary.

Four components limit the meaning of the word emotion to distinguish between emotions and other words [16]: 1) The appearance of certain feelings in the perpetrator in the emotional center, which is a part of the body often used as a reference when experiencing emotions, 2) Assessment of emotions, 3) The existence of a trigger that causes the appearance of emotions in the perpetrator, 4) Reactions to emotions. The categorization of the word emotion is inseparable from how emotional status is categorized. Categorizations that are often used are categories based on basic human emotional concepts.

Fig. 1. The six basic emotions

B. *Emotions*

Basic human emotions appear in infancy, especially the age of the first 6 months [17]. Basic human emotions are categorized into six that is *anger, disgust, fear, happiness, sadness, and shock* [18] as presented in Fig. 1. Happy Emotions describe reactions to success or satisfaction. Sad emotions describe a reaction of failure or lethargy. The emotion of fear is an expression of concern for a threat. Angry emotions describe an upset or frustrating reaction. The emotion of surprise describes the spontaneous reaction to a situation that comes suddenly or suddenly. Whereas the emotion of disgust describes the reaction of rejection to something that causes discomfort.

A person's emotions can be recognized by verbal communication in both oral (voice) and written (text) forms. Communication conveyed through writing (text) requires analysis to group them into certain categories of emotions. In general, the message a person usually conveys in the process of communication is in the form of sentences with a compound structure. Classification results in compound structure sentences that tend to contain more than one category of emotions. Therefore, the CNN model with multilabel and multichannel layer classifications is expected to be able to recognize the dominant classes of emotions used as the basis for the formation of compound facial expressions.

Compound emotions are a combination of two or more basic emotions. There are 15 mixed emotions, including happy-surprise, sad-surprise, angry-surprise, fear-surprised, disgust-surprised, happy-sad, sad-disgusted, angry-disgusted, disgust-fear, sad-fear, angry-sad, angry-fear, hateful, and fascinated [19]. Each of these types of compound expressions is different from the others. For example, a happy surprise is very different from happy fear or pleasure disgust. In addition

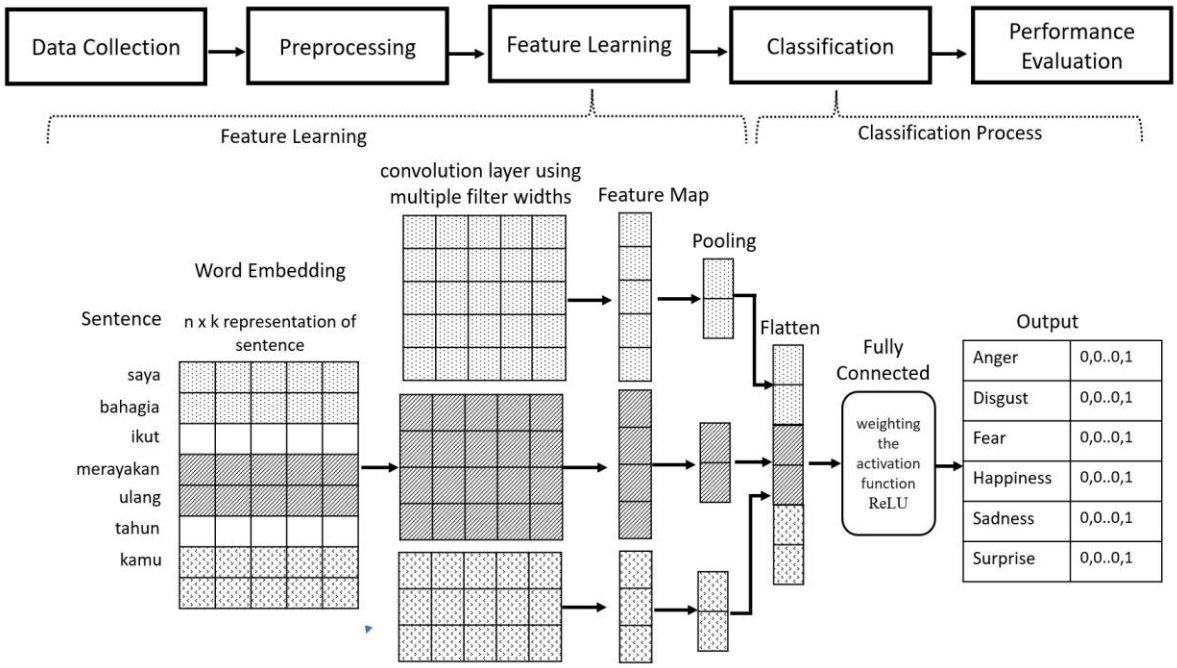


Fig. 2. The system overview of our method

to the above types of emotional expression, there are several secondary emotions [20]: disappointment and hatred. These emotional expressions are a combination of two or more basic emotions.

III. Related Work

There is research on the process of classification of scientific sentences in Indonesian using 2019 data in the form of single sentences obtained from computational scientific papers, which has been labeled into four classes: "Weak," "Comparison," "Point," and "Neutral." This study has conducted a simulation test using CNN Multichannel with the Adam optimization model, which has a learning rate of 0.001, resulting in an accuracy of 74.51% with a loss value of 0.82. In comparison, the CNN-Single channel model produces values of accuracy and loss of 70.76 and 1.73, respectively. [3].

The CNN model is used to classify Indonesian language news [21]. The research stage consisted of pre-processing text, removal of stopwords, and stemming stages is very suitable for Indonesian. Experiments was conducted 472 text data of the Indonesian News from several sources with four categories: "*olahraga*" (sports), "*hiburan"* (entertainment), "*teknologi*" (technology), and "*berita utama*" (headlines). The 472 text data were divided into 2 parts with proportion of 80% training data and 20% test data. The experimental results with 10 epochs and 300 hidden layers are that CNN has the best accuracy percentage of 90.74% and a loss value of 29.05%.

Research on the classification of Indonesian sentences used the CNN model [12]. The process of categorizing Indonesian sentences was based on multilabel classification. The CNN model uses one channel on the CNN layer forwarded by the Max-Pooling layer as the encoder layer. The classification result is a probability value that shows the intensity of emotions from each class of emotions. The CNN Single-Channel model used for the multilabel classification of Indonesian sentences can produce a compound emotion class with an accuracy value of 94.5%. In a sentence, a compound structure can generate more than 1 (one) class of emotions that have a probability value with a dominant emotional intensity.

Proposed research to solve the problem of short text classification by using CNN Models and semantic extensions [22]. Some suggestions are 1) improving similarity for word vector table coverage in short text preprocessing, 2) using methods for the semantic expansion of short texts so that the CNN model can find related words in a short text, 3) Expansion of semantics, especially semantics at the sentence level and word level semantics, especially in short texts. The experimental result is a very significant increase in classification performance with an accuracy rate of 87.37%.

Based on some of the research above, the CNN model can be developed by exploring the CNN architecture. The exploration includes the use of multichannel, which can process text with varying lengths to improve its performance. CNN Multi-Channel was leveraged to acquire the most discriminatory features [23]. Therefore, the study is proposed to achieve such goals.

IV. Methodology

The stages of the study consisted of 1) data acquisition, 2) text preprocessing, 3) word embedding, and 4) the learning and classification process. Text preprocessing aims to reduce data noise. Word embedding is used to convert a corpus of unique words into word vectors by using the Word2vec CBOW model. The learning process uses CNN's 1D Multichannel to gain training weights for the Indonesian sentence classification process. The system overview of our method is presented in Fig. 2.

In the methodology, we experimented with different neural word embeddings that were learned differently in different settings. The word2vec-based neural word embeddings are used to map each word to a corresponding vector containing real values. The document is represented as several vectors inserted into the CNN layer.

A. Data Collection

The data used in this study is a collection of Indonesian sentences sourced from social media posts totaling 47,326 sentences. The data consists of sentences with single structures and compound structures annotated by linguistic experts based on the categories of emotions contained in each sentence. The distribution of annotated data is presented in Table I.

TABLE I
DISTRIBUTION OF DATASET

#	Anger	Disgust	Fear	Happiness	Sadness	Surprise
1	6708	255	726	4992	7090	316
0	40618	47071	46600	42334	40236	47010

In Table 1, it appears that each class has an unbalanced amount of data. This issue may result in the model classifying the data into false labels. This is because there are too many false labels when compared to true labels. Therefore, class weighting defends the features present on the true - label. The class weights are annotated in (1).

$$w_j = \frac{n\ samples}{n\ classes \cdot n\ samples} \quad (1)$$

Where w_j is the weight of class j, $n\ samples$ is the total sample data, and $n\ classes$ are the total class.

B. Text Preprocessing

Quality data greatly affects the performance of algorithms, such as some studies that analyze the influence of preprocessing data on predictability [24], [25]. Therefore, the data needs to be preprocessed so that data processing with algorithms becomes consistent. Text pre-processing is an important stage for generating the important features of each sentence.

The data used in this study is text data composed of words that include The text data used is composed of plural words, single words, special characters, and numbers. Therefore, data preprocessing consists of a transformation process that converts the original form into a word-based form, an elimination process that eliminates words considered unnecessary for interpretation, and the process of normalization and word embedding. The normalization process aims to improve the quality of the text so that it is suitable for processing by machines.

B.1. Text Cleaning and Case Folding

Text cleansing is the process of removing unimportant features, such as numbers, punctuation marks, and user names. Meanwhile, case folding is one of the simplest and most effective forms of text preprocessing. Case folding aims to convert all letters in a document to lowercase [25]. One way is to use lowercase. In Python programming, the function of cleaning numbers is ***re.sub(pattern, replace_string, string)***, and the function of cleaning numbers is *.translate()*. Meanwhile, the case folding operation can use the *string* and *regex* library.

B.2. Tokenizing and Filtering

Tokenizing functions to split sentences into words. The technique used is to cut the word in the space. The tokenizing process can directly use the function available on the NLTK, which is *word_tokenize()*.

On the other hand, filtering is the process of removing very common words (stopwords). The words included in the stopword, for example, which, and, in, that, with, for, not, from, in, will, on, this, also, me, as well as, is, that, other, you, etc.

B.3. Text Normalization

Text normalization is a preprocessing stage that aims to convert text into a canonical representation (by language rules). A series of text normalization tasks consists of 1) transforming non-standard or typos words into standard words based on a standard word list database, 2) converting an abbreviation into an equivalent of the word it stands for, 3) stemming, removing affixes (suffixed, prefixes, infixes, circumfixes) to get the base word [26]. The algorithm used in the steaming process is Sastrawi, which has better accuracy than the previous algorithm.

B.4. Word Embedding

The CNN model cannot accept input data in the form of strings, so each sentence must be converted into a numeric vector. Word2vec is a neural network used to process text into vector values [23]. This method works by taking the corpus of text as input by going through the preprocess and one hot-encoding stage, then generating a vector value representation of each word in the word corpus as output [27]. Word2vec has two types of modeling architectures that can be used to represent word vectors into an embedding matrix. These architectures include Continous bag-of-word (CBOW) and Skip-gram. The embedding matrix represents each feature in a higher vector dimension. Each feature in the embedding matrix is annotated in (2).

$$x_i \in R^k \quad (2)$$

$$x = [x_1, x_2, \dots, x_l] \quad (3)$$

Where x_i is the ith and $x \in R^{l \times k}$ vector feature with l as the vector length and k is the dimension of the matrix [19]. The resulting embedding matrix can be annotated in (3).

C. Feature Learning

The learning process stage consists of convolution, pooling, concatenating, and flattening layers. The Convolutional layer serves as an encoder or feature solver of a layer embedding in matrix convolution operations [8]. Convolution is the operation of multiplying the dot product matrix between the input data and the kernel, which is used as a filter against the input data to produce a feature map. The convolutions are divided into channels with different filter sizes on each channel. The matrix convolution process takes place sequentially, using a kernel filter that generates a feature map. Convolution operations are annotated in (4).

However, each sentence of the dataset has a different size or length, so it is necessary to do a padding process to get a vector of the same size without adding or subtracting the features of each datum. Padding is the process of adding the zeros of each vector until the specified size has been met, as shown in Fig. 3.

$$m_j = f(w \cdot x_{i:i+s-1} + b) \quad (4)$$

with $w \in R^{s \times k}$, where s is the filter size, the notation $\cdot$ (point) is the dot matrix operation, b is a biased value, f is a non-linear function, and $x_{i:i+s-1}$ spelled out in (5).

$$x_{i:i+s-1} = x_i + x_{i+1} + x_{i+s-1} \quad (5)$$

Where $+$ notation is the concatenation of vector lines. So that a formulation is obtained to produce a feature map that is annotated in (6).

$$m = [m_1, m_2, \dots, m_{l-s+1}] \in R^{l-s+1} \quad (6)$$

Where m is one feature map. In one convolutional layer, there are several *n-filter* that produce an *n-feature map*, so the whole process in one convolutional layer produces a formulation in (7).

$$M = [m_1, m_2, \dots, m_n] \in R^{n \times (l-s+1)} \quad (7)$$

Where M is *n-feature maps* produced by one convolutional layer.

Furthermore, the feature map value M is entered into the ReLU activation function. The function is used to avoid negative values, as annotated in (8). ReLU activation serves to change the negative value to 0 to avoid negative values. The output of the ReLU is expressed as 0 (zero) if the input is negative. But if the input is positive, then the output will be equal to the input value of the activation function itself [8a].

$$ReLU(M): \begin{cases} M, if\ M \geq 0 \\ 0, \ if\ M < 0 \end{cases} \quad (8)$$

Feature maps generated from convolutional layers are passed on to the max-pooling or average-pooling layer. The max-pooling layer processes each feature map in the maximum distribution to produce the highest value of each feature. Whereas, the average-pooling layer takes the average value of each feature map.

The feature map generated from feature learning is in the form of a multidimensional array, while the input for a fully connected layer must be data in vector form. Therefore, a function is needed that can convert multidimensional data arrays into vector form. Flatten serves to reshape the feature map from a multidimensional array into a 1-dimensional vector [28]. This is necessary so that these values can be used as input on a fully connected layer.

D. Classification Process

Multilayer Perceptron (MLP) is a classification layer consisting of an input layer, an output layer, and several hidden layers. Each layer is constructed from neurons that are interconnected with each other. For the learning process using several methods such as Backpropagation. In this layer, some dropouts are fused to prevent overfitting during the learning process. The activation function commonly used in this layer is Softmax, which converts the output probability values for each class as annotated using (9). The softmax activation function is used to calculate the probability value of each target class against all target classes. The output probability range on softmax is a value from 0 to 1. Softmax uses the exponential value of a given input divided by the sum of the exponential values of all the values in the output.

$$Softmax(y_j) = \frac{e^{M^T w_j}}{\sum_{k=1}^{K} e^{M^T w_k}} \quad (9)$$

Where y_j is the j^{th} index output and K is the total number of outputs generated in the classification process.

V. Experimental Results

This section describes the experimental results using various types of CNN and LSTM models. Each experiment is analyzed and evaluated for its performance.

5.1 Types of CNN and LSTM Models for Experiments

The experiments in this study focused on the development of CNN models applied to the process of classification of emotional categories of Indonesian sentences. We determined eight types of CNN models and one LSTM model in the classification process as presented in Table II. CNN's eight models consist of single-channel and multichannel. Single channel on the CNN model shows that in the there is only one kernel filter size convolutional layer, while there is more than one kernel filter size in the multichannel. There is one LSTM model with 32 bidirectional LSTM units as an experimental comparison of the CNN model. Each model has its specifications so that the experimental results of each model can be analyzed for comparison of performance results.

The convolutional layer performs convolution operations to convert inputs into feature maps by performing dot operations between the input matrix and filters. A filter consists of several K -weight matrices (called kernels). In the 1D CNN layer, kernel sizes are set to various sizes. Determination of Kernel size refers to the number of words reads at any time. The smaller the kernel size, the more detailed the resulting result will be. The number of features produced is strongly influenced by the CNN architecture being trained. Therefore, the number of filters used determines the number of feature maps obtained. The number of kernels on the l^{th} layer, according to the number of feature maps on the $l - 1$ layer. The number of filters used on layer l determines the number of feature maps obtained, and the number corresponds to the number of kernels on layer $l + 1$.

In pooling layer, the number of features obtained from the convolution layer will be reduced again through the subsampling process. This process also utilizes convolutions, it's just that the purpose is somewhat different from the convolutions contained in the convolution layer. Convolution in the pooling layer aims to reduce the number of features that will be processed on the next layer by utilizing only the features that are considered the most representative. There are two types of pooling, namely max pooling and average pooling. We used a combination of max and average pooling.

TABLE II
THE CNN MODELS AND LSTM FOR EXPERIMENTS

Model#	Model	Description
#1	LSTM	Simple bidirectional LSTM with 32 LSTM units
#2	CNN v1.0.0	1D CNN Single channel with kernel size = 3 and max-pooling layer
#3	CNN v1.0.1	1D CNN Single channel with kernel size = 4 and max-pooling layer
#4	CNN v1.0.2	1D CNN Single channel with kernel size = 5 and max-pooling layer
#5	CNN v1.0.3	1D CNN Single channel with kernel size = 6 and max-pooling layer
#6	CNN v1.0.4	1D CNN Multichannel with kernel size = [3, 4, 5] and max-pooling layer
#7	CNN v1.0.5	1D CNN Multichannel with kernel size = [3, 4, 5] and average-pooling layer
#8	CNN v1.0.6	1D CNN Multichannel with kernel size = [4, 5, 6] and max-pooling layer
#9	CNN v1.0.7	1D CNN Multichannel with kernel size = [4, 5, 6] and average-pooling layer

The CNN model has high accuracy because it has the number of feature extractions produced by convolutions and the number of neurons, as well as the merging of each neuron by using updated weights on a given iteration. Many parameters of the CNN model can be optimized to obtain high precision and accuracy values and a short time. We proposed a combination of various kernel sizes with CNN's single channel and CNN's multichannel architecture to obtain an optimal level of accuracy.

5.2 Performance Evaluation

In this section, we conducted a series of tests ranging from comparing CNN-Multichannel and CNN-Singlechannel models, comparing multiple optimization models with various kernel size variations and combinations of max-pooling and average-pooling, and testing system classification models from the results of accuracy, precision, recall, and *f1-score*.

$$Recall = \frac{TP}{TP + FN} \tag{10}$$

$$Precision = \frac{TP}{TP + FP} \tag{11}$$

$$F1 - Score = 2 * \frac{precision * recall}{precision + recall} \tag{12}$$

$$Accuracy = \frac{TP + TN}{TP + FN + FP + TN} \tag{13}$$

Where TP (abbreviation of True Positive) is positive data that is predicted correctly, FP (abbreviation of False Positive) is negative data predicted as positive data, FN (abbreviation of False Negative) is positive data predicted as negative data, and TN (abbreviation of True Negative) is negative data that is predicted correctly. Accuracy is the degree of closeness between the actual value and the predicted value. Precision is the level of accuracy between the actual value and the predicted value. Recall is the ratio of true positive predictions compared to the overall true positive data, and F1-Score is the average value of precision and recall.

The experimental result of each model is a probability value converted into binary labels of each category using optimal thresholds. In this experiment, we used the optimal value of the thresholds calculated using (12).

$$t_i = F1\ score(y_{true}, t(y_{prob})) \tag{12}$$

With $0 < t_i < 1$

Where y_{true} is the actual label, $t(y_{prob}))$ is a function of converting probability values to y_{pred} (provision: if the probability value is ≥ then it returns a TRUE value, but if the probability value is < then it returns a FALSE value), and $F1 - score(y_{true}, t(y_{prob}))$ is the F1-score value of each threshold? The *F1-Score* value is used as a reference for determining the optimal value of thresholds based on the amount of unbalanced data. The *F1-Score* measurement is a collaboration of the aggressiveness and the sensitivity of the engine which is formulated as (13)

$$t_{optimal} = \max(t_1, t_2, t_3, .. t_n) \tag{13}$$

Where max is the maximum distribution and is the i^{th} threshold.

We have conducted experiments based on the CNN and LSTM models specified in Table II and the equations formulated above. Description and analysis of the results of the experiment are presented in Table III to Table VII.

TABLE III
RECALL CALCULATION RESULTS FOR EACH MODEL AND EMOTION CLASS

Model#	Anger	Disgust	Fear	Happiness	Sadness	Surprise
#1	0.73	0.82	0.87	0.69	0.71	0.88
#2	0.67	0.87	0.91	0.76	0.75	0.84
#3	0.76	0.87	0.87	0.81	0.79	0.83
#4	0.78	0.87	0.82	0.82	0.72	0.83
#5	0.76	0.88	0.79	0.81	0.83	0.87
#6	0.72	0.92	0.82	0.81	0.87	0.82
#7	0.87	0.84	0.91	0.73	0.80	0.84
#8	0.79	0.92	0.85	0.80	0.85	0.88
#9	0.78	0.88	0.83	0.84	0.80	0.88

Table III presents the results of the Recall calculation experiment for each model and emotion class. The average Recall value for the entire model is 0.82 (82%). This value shows the ratio of data success predicted for each emotion class compared to the overall actual data for the emotion class. The average Recall value increased higher occurred on model#6 (CNN v1.0.4), model#7 (CNN v1.0.5), model#8 (CNN v1.0.6), and model#9 (CNN v1.0.7).

TABLE IV
PRECISION CALCULATION RESULTS FOR EACH MODEL AND EMOTION CLASS

Model#	Anger	Disgust	Fear	Happiness	Sadness	Surprise
#1	0.69	0.65	0.65	0.67	0.73	0.66
#2	0.73	0.65	0.72	0.82	0.85	0.89
#3	0.82	0.8	0.77	0.84	0.79	0.76
#4	0.82	0.77	0.79	0.85	0.83	0.81
#5	0.77	0.96	0.83	0.79	0.78	0.78
#6	0.77	0.79	0.83	0.84	0.86	0.84
#7	0.76	0.93	0.83	0.84	0.81	0.84
#8	0.82	0.89	0.88	0.87	0.85	0.87
#9	0.84	0.88	0.83	0.81	0.84	0.79

Table IV presents the results of precision calculation experiments for each model and class of emotions. The average Precision value for the entire model is 0.80 (80%). This means that the ratio of the accuracy of the actual data compared to the predicted data for each class of emotions is well. On model#1 (LSTM), the average value of Precision shows the lowest value of 68%, while in model#8 (CNN v1.0.6), the average value of Precision shows the highest value of 86%, which indicates a good accuracy ratio.

TABLE V
F1-SCORE CALCULATION RESULTS FOR EACH MODEL AND EMOTION CLASS

Model#	Anger	Disgust	Fear	Happiness	Sadness	Surprise
#1	0.71	0.73	0.74	0.68	0.72	0.75
#2	0.70	0.74	0.80	0.79	0.80	0.86
#3	0.79	0.83	0.82	0.82	0.79	0.79
#4	0.80	0.82	0.80	0.83	0.77	0.82
#5	0.76	0.92	0.81	0.80	0.80	0.83
#6	0.74	0.85	0.82	0.82	0.86	0.83
#7	0.81	0.88	0.87	0.78	0.80	0.84
#8	0.80	0.90	0.86	0.83	0.85	0.87
#9	0.81	0.88	0.83	0.82	0.82	0.83

TABLE VI
ACCURACY CALCULATION RESULTS FOR EACH MODEL AND EMOTION CLASS

Model#	Anger	Disgust	Fear	Happiness	Sadness	Surprise
#1	0.89	0.92	0.89	0.91	0.83	0.90
#2	0.91	0.92	0.89	0.96	0.85	0.90
#3	0.92	0.91	0.89	0.94	0.86	0.91
#4	0.92	0.91	0.93	0.87	0.92	0.91
#5	0.94	0.92	0.96	0.95	0.84	0.96
#6	0.94	0.92	0.97	0.96	0.95	0.95
#7	0.92	0.93	0.97	0.94	0.97	0.94
#8	0.95	0.91	0.98	0.97	0.95	0.97
#9	0.95	0.90	0.98	0.95	0.95	0.99

Table V presents the experimental results of F1-Score calculations for each model and class of emotions. The average F1-Score value for all models is 0.81 (81%), which indicates that the average ratio of precision and recall weighted is high This value indicates a fairly good accuracy ratio. The highest F1-Score value is on model#8 (CNN v1.0.6).

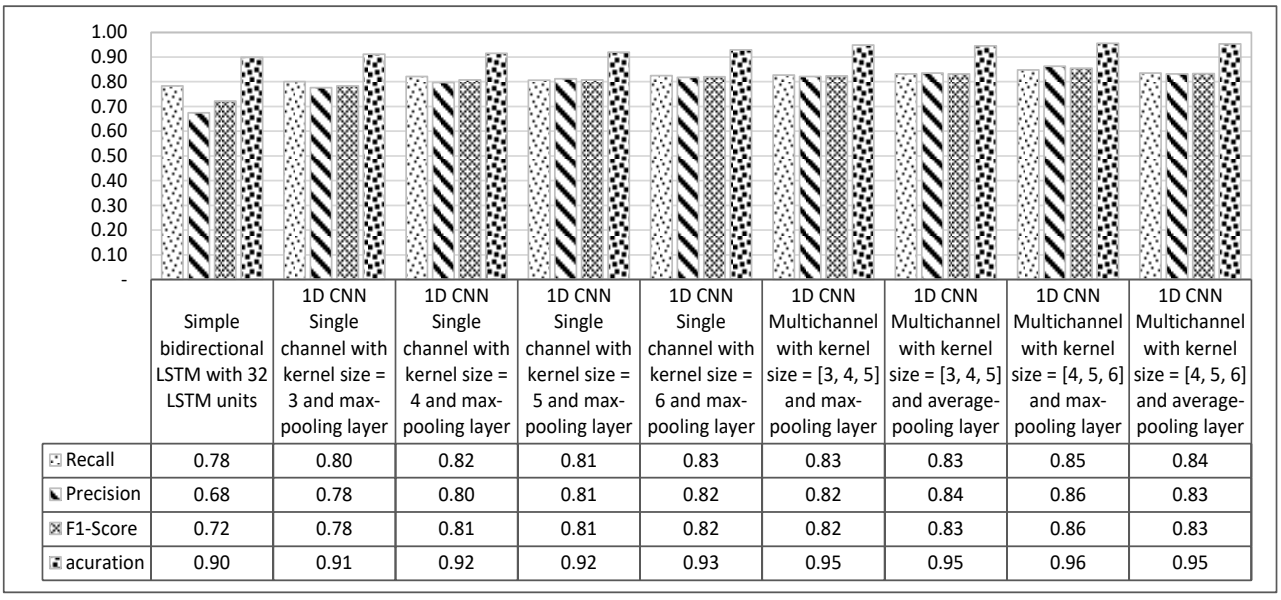
TABLE VII
THE AVERAGE PERFORMANCE OF EACH MODEL

Model#	Recall	Precision	F1-Score	Accuracy
#1	0.78	0.68	0.72	0.90
#2	0.80	0.78	0.78	0.91
#3	0.82	0.80	0.81	0.92
#4	0.81	0.81	0.81	0.92
#5	0.83	0.82	0.82	0.93
#6	0.83	0.82	0.82	0.95
#7	0.83	0.84	0.83	0.95
#8	0.85	0.86	0.86	0.96
#9	0.84	0.83	0.83	0.95

Based on the model performance results above, the highest recall generated in the CNN v1.0.6 model is 0.85. High recall results mean that the model can predict more true labels than other models. Then the highest precision value generated in the CNN v1.0.6 model is 0.86, which means that the model produces the least FP. Finally, the highest F1-Score and accuracy were produced by model#8 (CNN v1.0.6) model, namely 0.86 and 0.96 respectively.

Model #8 (CNN v1.0.6) is a convolution neural network model using multiple channels. Channels refer to the number of input layers in the convolution process stage. Each channel uses three different kernel sizes, namely: 4, 5, 6. Kernel size 4 means that every time do a convolution, the 4x4 matrix in the data input multiplied by 4x4 is filtered.

The accuracy value of each model that has been tested in the experiment above produces a good value (above 0.91). This is due to the large number of false-label values, so to determine the best model based on the experimental results in



	Simple bidirectional LSTM with 32 LSTM units	1D CNN Single channel with kernel size = 3 and max-pooling layer	1D CNN Single channel with kernel size = 4 and max-pooling layer	1D CNN Single channel with kernel size = 5 and max-pooling layer	1D CNN Single channel with kernel size = 6 and max-pooling layer	1D CNN Multichannel with kernel size = [3, 4, 5] and max-pooling layer	1D CNN Multichannel with kernel size = [3, 4, 5] and average-pooling layer	1D CNN Multichannel with kernel size = [4, 5, 6] and max-pooling layer	1D CNN Multichannel with kernel size = [4, 5, 6] and average-pooling layer
Recall	0.78	0.80	0.82	0.81	0.83	0.83	0.83	0.85	0.84
Precision	0.68	0.78	0.80	0.81	0.82	0.82	0.84	0.86	0.83
F1-Score	0.72	0.78	0.81	0.81	0.82	0.82	0.83	0.86	0.83
acuration	0.90	0.91	0.92	0.92	0.93	0.95	0.95	0.96	0.95

Fig. 3. Performance comparison of proposed models

the study, it is added using the F1-Score value as a determinant. Statistics of F1-Score value and accuracy for each proposed model is presented in Fig. 4.

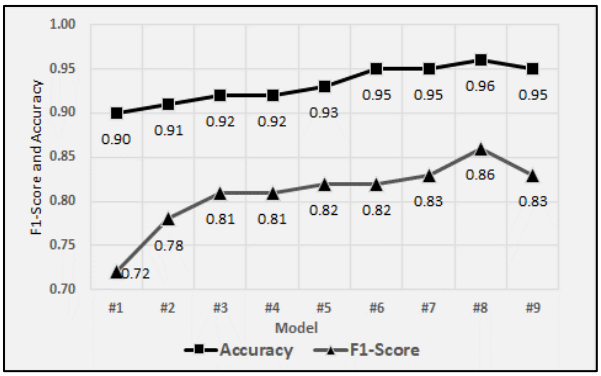


Fig. 4. The level of F1-Score and accuracy of each model

Fig. 4 shows the experimental results for the F1-Score and accuracy that the highest F1-Score and accuracy were produced by model#8 (CNN v1.0.6) model, namely 0.86 and 0.96 respectively. F1-Score is the average value between precision and recall. F1-Score will produce a high value, if both recall and precision values are high. F1-Score will produce a low value if both recall and precision values are low. F1-Score will produce a medium value if one of the recall or precision value is high and the other is low. The F1-Score resulting from the experiments in this study is high which indicates that the classification model has high precision and recall values.

Performance measurement using the F1-Score is performed when the composition of the number of datasets is unbalanced so that the calculation is focused on False Positive and False Negative values. While performance measurement using accuracy is performed when the composition of the number of datasets is balanced so that the calculation is focused on True Positive and True Negative values. In this study, we use both evaluation methods to measure the performance of the classification process. The F1-Score evaluation method is used to ensure that the method used to change dataset conditions from an unbalanced composition to a balanced dataset composition still produces good performance.

VI. Conclusion and Future Works

Based on the results of experiments and model evaluations, this study produced a model that can categorize the emotional classes of Indonesian sentences with a single and compound structure. The experimental results show that model #8 (CNN v1.0.6) has the highest F1-Score and accuracy, namely 0.86 and 0.96 respectively. In a series of tests, the model produces the best configuration of the CNN-Multichannel model with kernel size = [4, 5, 6] (using three different kernel sizes for each channel, namely: 4,5,6) and max-pooling layer. The results of machine predictions that produce categories of emotions from each sentence in Indonesian can be used as a determinant of artificial facial expressions.

Although this study has produced a CNN model that can effectively categorize emotional classes, it is necessary to conduct a further study for its implementation in the formation of artificial facial expressions so that interactive communication can be produced that looks more natural like humans.

Based on the results of experiments, the CNN model can outperform a simple LSTM model by 83%. with an average F1-Score value produced. These results also do not escape the use of an optimal threshold that can adjust the amount of weighting of each model.

CNN model performance can be improved by adding sentence semantic information and adding highly discriminatory features. The use of transfer learning techniques has been shown to improve CNN model performance and significantly reduce training time on image datasets. Several types of transfer learning models, namely VGG19, VGG16, MobileNet, DenseNet201, and ResNet50. Other factors that will be added in the implementation of the CNN model are the development of hidden layer variations, the use of other activation functions (besides sigmoid), and the use of other embedding representations. In future study, some of these factors will be applied to the CNN model, especially for text classification. Thus, the CNN model performance is expected to be significantly improved.

Acknowledgment

We thank the leaders of Dian Nuswantoro University who have provided support, spirit, facilities, and infrastructure so that this study can be completed.

References

[1] Luca Del Giacco, M. Teresa Anguera, and Silvia Salcuni, "The Action of Verbal in the Therapeutic Alliance Construction: A Mixed Methods Approach to Assess the Initial Interactions With Depressed Patients," *frontiers in Psychology*, Volume 11, Article 234. February 2020, pp. 1–26, https://doi.org/10.3389/fpsyg.2020.00234, 2020.

[2] Jacob Israelashvili and Agneta Fischer, "Recognition of Emotion from Verbal and Nonverbal Expressions and Its Relation to Effective Communication: A Preliminary Evidence of a Positive Link," *Journal of Intelligence* 2023, 11(1), 6, pp. 2–8, 28 December 2022, https://doi.org/10.3390/jintelligence11010006, 2022.

[3] Emad UI Haq Qazi, Abdulrazaq Almorjan, and Tanveer Zia, "A One-Dimensional Convolution Neural Network (1D-CNN) Based Deep Learning System for Network Intrusion Detection," *Applied Aciences 2022, 12, 7986, pp. 2-14, 10 August 2022*, https://doi.org/10.3390/app12167986, *2022.*

[4] Leon Gilberto Medellin Lopez, "The Indonesian Language and Its Potensial to Become an International Language," *Proceeding of the Third International Conference of Arts, Language and Culture (ICALC 2018),* Advances in Social Science, Education and Humanities Research (ASSEHR), volume 279, pp. 278-280, February 2019, https://doi.org/10.2991/icalc-18.2019.40, 2019.

[5] Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom, "A convolutional neural network for modeling sentences," *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, Baltimore, Maryland, USA, June 23-25 2014, pp. 655–665, DOI:10.3115/v1/P14-1062, 2014.

[6] Wen-tau Yih, Xiaodong He, and Christopher Meek, "Semantic parsing for single-relation question answering," In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), volume 2, pages 643–648, 2014.

[7] Lujuan Deng, Qingxia Ge, Jiaxue Zhang, Zuhe Li, Zeqi Yu, Tiantian Yin, and Hanxue Zhu, "News Text Classification Method Based on the GRU_CNN Model," *Hindawi: International Transactions on Electrical Energy Systems*, Volume 2022, Article ID 1197534, Published: 31 Aug 2022, pp. 1-11, https://doi.org/10.1155/2022/1197534, 2022.

[8] Y. Kim, "Convolutional Neural Networks for Sentence Classification," in Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2014, pp. 1746–1751, DOI: 10.1109/CLEI.2017.8226381, 2014.

[9] N. Kalchbrenner, E. Grefenstette, and P. Blunsom, "A Convolutional Neural Network for Modelling Sentences," Proc. ACL, pp. 655–665, April 2014, DOI:10.3115/v1/P14-1062, 2014.
[10] Y. Zhang and B. Wallace, "A Sensitivity Analysis of (and Practitioners' Guide to) Convolutional Neural Networks for Sentence Classification," Proceedings of the 8th International Joint Conference on Natural Language Processing, pp. 253–263, Taipei, Taiwan, November 27 – December 1, 2017.
[11] M. Bonzanini, "Mastering Social Media Mining with Python; Acquire and analyze data from all corners of the social web with Python," Copyright © 2016 Packt Publishing, ISBN 978-1-78355-201-6, First published: July 2016.
[12] Aripin, S. Sumpeno, Muljono, and M. Hariadi, "A model of Indonesian dynamic visemes from facial motion capture database using a clustering-based approach," IAENG International Journal Computer Science, vol. 44, no. 1, pp. 41–51, 2017.
[13] Raymond Van Dinter, Cagatay Catal, Bedir Tekinerdogan, "A Multi-Channel Convolution Neural Network approach to automate the citation screening process," Applied Soft Computing 112 (2021) 107765, ELSEVIER, pp. 1-14, DOI:https://doi.org/ 10.1016/j.asoc.2021. 107765, 2021.
[14] Yue Feng and Yan Cheng, "Short Text Sentiment Analysis Based on Multi-Channel CNN with Multi-Head Attention Mechanism," IEEE Access, Volume 9, DOI: 10.1109/ACCESS.2021.3054521, 2021.
[15] I Wayan Suryasa, I Nengah Sudipa, Ida Ayu Made Puspani, and I Made Netra, "Translation Procedure of Happy Emotion of English into Indonesian," *Journal of Language Teaching and Research*, Vol. 10, No. 4, pp. 738-746, July 2019, DOI: http://dx.doi.org/ 10.17507/jltr.1004.08, 2019.
[16] Robert F. Cavanagh, Yuko Asano- Cavanagh, and William P Fisher, Jr., "Natural Semantic Metalanguage as an approach to measuring meaning," *Journal of Physics: Conference Series*, 1379:012019, pp. 1-6, November 2019, DOI:10.1088/1742-6596/1379/1/012019, 2019.
[17] Dacher Keltner, Disa Sauter, Jessica Tracy and Alan Cowen, "Emotional Expression : Advances in Basic Emotion Theory," Journal of Nonverbal Behavior, DOI:10.1007/s10919-019-00293-3, pp. 1–48, 2019.
[18] P. Ekman and D. Cordaro, "What is Meant by Calling Emotions Basic," *Emot. Rev.*, vol. 3, no. 4, pp. 364–370, Oct. 2011, doi: 10.1177/1754073911410740, 2011.
[19] J. Zhang, Y. Li, J. Tian, and T. Li, "LSTM-CNN Hybrid Model for Text Classification," in *Proceedings of the 2nd International Conference on Electronics, Communication and Aerospace Technology, ICECA 2018*, no. Iaeac, pp. 450–458, DOI: 10.1109/ICECA.2018.8474920, 2018.
[20] S. Du, Y. Tao, and A. M. Martinez, "Compound Facial Expressions of Emotion", In: *Proceedings of the National Academy of Sciences*, pp. E1454-E1462, 2014.
[21] Muhammad Ali Ramdhani, Dian Saiadillah Maylawati, Teddy Mantoro, "Indonesian News Classification using Convolutional Neural Network," Indonesian Journal of Electrical Engineering and Computer Science, Vol. 19, No. 2, August 2020, pp. 1000-1009, DOI: 10.11591/ijeecs.v19.i2, pp. 1000-1009, 2020.
[22] Haitao Wang, Keke Tian, Zhengjiang Wu, Lei Wang, "A Short Text Classification Method Based on Convolution Neural Network and Semantic Extension," International Journal of Computational Intelligence Systems, Vol. 14(1), pp. 367-375, DOI: https://doi.org/10.2991/ijcis.d.201207.001, 2021.
[23] Muhammad Ali Ibrahim, Muhammad Usman Ghani Khan, Faiza Mehmood, Muhammad Nabeel Asim, Waqar Mahmood, "GHS-NET a generic hybridized shallow neural network for multi-label biomedical text classification," Journal of Biomedical Informatics, ELSEVIER, Available online 15 February 2021, pp. 1-16, DOI: https://doi.org/10.1016/j.jbi.2021.103699, 2021.
[24] HoSung Woo, JaMee Kim, and WonGyu Lee, "Validation of Text Data Preprocessing Using a Neural Network Model," Hindawi Mathematical Problems in Engineering, Volume 2020, Article ID 1958149, Published 14 May 2020, pp. 1-9, DOI: https://doi.org /10.1155/2020/1958149, 2020.
[25] R. D. Sharma, S. Tripathi, S. K. Sahu, S. Mittal, and A. Anand, "Predicting Online Doctor Ratings from User Reviews Using Convolutional Neural Networks," International Journal of Machine Learning and Computing, vol. 6, no. 2, pp. 149–154, 2016.
[26] K. Abainia, S. Ouamour, and H. Sayoud, "A novel robust Arabic light stemmer," Journal of Experimental & Theoretical Artificial Intelligence, no. 1–17, 2016.
[27] Ghoziyah Haitan Rachman, Masayu Leylia Khodra, Dwi Hendratmo Widyantoro, "Word Embedding for Rhetorical Sentence Categorization on Scientific Articles," Journal of ICT Research and Applications, Vol. 12, No. 2, pp. 168-184, DOI: https://doi.org /10.5614/itbj.ict.res.appl.2018.12.2.5, 2018.
[28] Chieng, H. H., Wahid, N., and Ong, P., "Parametric Flatten-T Swish: An Adaptive Non-linear Activation for Deep Learning," Journal of Information and Communication Technology (Journal of ICT), Vol. 20, No. 1, January 2021, pp. 21-39, DOI: https://doi.org/10.32890/ jict.20.1.2021.9267, 2021.