Proceedings of the 1st Joint SLTU and CCURL Workshop (SLTU-CCURL 2020), pages 185–193
Language Resources and Evaluation Conference (LREC 2020), Marseille, 11–16 May 2020

Speech-Emotion Detection in an Indonesian Movie

Fahmi, Meganingrum Arista Jiwanggi, Mirna Adriani
Faculty of Computer Science, Universitas Indonesia
Universitas Indonesia
fahmi51@ui.ac.id, meganingrum@cs.ui.ac.id, mirna@cs.ui.ac.id

Abstract

The growing demand to develop an automatic emotion recognition system for the Human-Computer Interaction field had pushed some research in speech emotion detection. Although it is growing, there is still little research about automatic speech emotion detection in Bahasa Indonesia. Another issue is the lack of standard corpus for this research area in Bahasa Indonesia. This study proposed several approaches to detect speech-emotion in the dialogs of an Indonesian movie by classifying them into 4 different emotion classes i.e. happiness, sadness, anger, and neutral. There are two different speech data representations used in this study i.e. statistical and temporal/sequence representations. This study used Artificial Neural Network (ANN), Recurrent Neural Network (RNN) with Long Short Term Memory (LSTM) variation, word embedding, and also the hybrid of three to perform the classification task. The best accuracies given by one-vs-rest scenario for each emotion class with speech-transcript pairs using hybrid of non-temporal and embedding approach are 1) happiness: 76.31%; 2) sadness: 86.46%; 3) anger: 82.14%; and 4) neutral: 68.51%. The multiclass classification resulted in 64.66% of precision, 66.79% of recall, and 64.83% of F1-score.

Keywords: Speech, Emotion Detection, Bahasa Indonesia, Artificial Neural Network (ANN), Recurrent Neural Network (RNN), Long Short Term Memory (LSTM), Word Embedding

1. Introduction

The growth of studies about speech emotion detection began to be applied in various fields of computing, especially in the field of Human-Computer Interaction (HCI). A good HCI system is said to not only be able to capture important information but also to detect emotions from users, so the system will give more appropriate responses (Wunarso and Soelistio, 2017). Some examples of the useful application of emotion detection in HCI systems are the automated call center system or personal assistant application in which such a system might provide the more natural interaction when emotions are involved to generate users' responses (Lubis et al., 2014).

The commonly used emotion grouping model is the valence-arousal model by Barrett and Russell (1999). Valence-arousal model states that emotions can be decomposed in 2 aspects i.e. the valence aspect which expresses the sentiment of emotions and the arousal/intensity aspect which states the intensity in expressing those emotions. Valence-arousal models are built-in 2-dimensional planes with the horizontal axis is valence and the vertical axis is intensity. Then each emotional class is measured in degrees of intensity and valence and mapped in the 2-dimensional plane. For example, anger can be seen as a class of emotions that have negative sentiments and are overflowed with a high enough intensity, then the valence-arousal model of angry emotions will be mapped to the high-intensity quadrant and negative valence. Valence-arousal models provide information that emotional classes that are in one quadrant have high similarity and are difficult to distinguish by humans (Barret and Russel, 1999). Generally, research on speech emotion considers the issue of choosing emotional classes.

One of the earliest studies about speech emotion is a study by Yu et al. (2001) that developed a speech classification model for 4 classes of emotion using pitch as its feature. For the following years, the leading studies in the field of speech emotion began to adapt Mel-scaled Frequency Cepstral Coefficient (MFCC) as the feature to improve the system performance, e.g. the research by Ko et al. (2017) that built a speech classification model for anger, sad, happiness, disgusted, shocked, scared, and neutral.

The research about speech in Bahasa Indonesia has also been beginning to grow in the past few years. The topics include speech-language identification (Safitri et al., 2016), speech synthesis (Vania and Adriani, 2011), and speech recognition (Wanagiri and Adriani, 2012). However, there are still a few numbers of research about speech emotion conducted in Bahasa Indonesia, whereas research on speech emotion has been done in other languages such as English (Livingstone and Russo, 2018), German (Burkhardt et al., 2005) and Language Persia (Keshtiari, 2015). The other issue is the lack of standard corpus in Bahasa Indonesia to be used for the experiments in automatic speech-emotion detection.

We also learn from the previous studies that although usually the speech data is transformed into statistical representations, the temporal speech data representation may address some issues on the previous well-known representation. However, the research that views speech data as temporal data was still hardly found. Also, studies of speech emotion were rarely utilizing textual information. To address the above problems, this research aims to develop the speech-emotion detection model for Bahasa Indonesia by transforming speech data into temporal representation as to the acoustic feature and also by adding textual information from speech as the lexical feature. Based on the valence-arousal model, we choose 4 emotion classes for this study i.e. happiness, sadness, anger, and neutral that comes from different quadrants in the model to improve the ability of the system to distinguish the emotion.

This paper structure is as follows: the Introduction section describes briefly about the background of this research. The second section of Related Works summaries the previous researches in a particular area. The Methodology section explains briefly about the steps run to do the research. The Results and Analysis sums up the result of our experiments and also the summary of our analysis behind the results. Lastly, the Conclusion wraps up the content of this paper and describes the possible future works.

2. Related Works

2.1 Research on Detecting Emotions in Speech Data

There were several prior studies related to speech-emotion detection. The research group that will be reviewed first used statistical representations such as averages, standard deviations, medians, minimums, and maximums as features in speech data. Previously, the audio data features are generally in the form of sequences that depends on to the duration and the majority of classification algorithms cannot process data of different sizes. The statistical view is one of the solutions to the above problems and was adopted in research conducted by Yu et al. (2001), Lubis et al.(2014), and Wunarso and Soelistio (2017).

The first study about speech emotion using the statistical feature was conducted by Yu et al. (2001). This study aimed to build a model of speech-emotion classification in 4 classes (anger, sadness, happiness, and neutral) using the corpus built from the television series. The features used in this study focus on the use of pitch and rhythm such as smoothed pitch, derivative pitch, and speaking rate. The results of this process are 16 features. This study used Artificial Neural Network (ANN) algorithm, K-Nearest Neighbor (KNN), and Support Vector Machine (SVM). The experimental results show that SVM resulted in the most accurate predictions of up to 77% for anger, 65% for happiness, 83% for neutral, and 70% for sadness.

The second study on the statistical feature group is the speech emotion research by Wunarso and Soelistio (2017) that builds a model of speech emotion classification for anger, sadness, and happiness using the corpus built by collecting voice recordings from 38 volunteers and producing around 3420 speech data. The study only used 3 features i.e. the average amplitude, average wavelet, and duration of the speech. The classification algorithms are ANN and SVM where SVM resulted in 76% accuracy and ANN reached only 66% accuracy.

The third study is the speech emotion research by Lubis et al. (2014) that builds a model of speech emotional classification for anger, sadness, happiness, and neutral using the corpus built by collecting speech from the talk show. The study used various features i.e. Mel-scaled Frequency Cepstral Coefficient (MFCC), spectral features, energy, and pitch with SVM as the classification algorithm. The average accuracy obtained is around 80% using SVM.

The second group of researchers used temporal representations as to the speech data features. It means, the sequence of features directly feeds into the algorithms without worrying about dimensional differences. As explained, the different duration of audio data can be a problem because there are not many algorithms that are able to handle these problems, but the studies below adopt algorithms that can accept sequence inputs such as the Hidden Markov Model (HMM) in research by Ko et al. (2017) and Recurrent Neural Network (RNN) in a study by Basu et al. (2017).

The first study using the temporal feature is the study of speech emotion by Ko et al. (2017) that aimed to build a speech emotion classification model in anger, pleasure, sadness, neutral, disgust, surprise, and fear. They build a corpus from drama and films with around 454 speech data with a fairly even distribution for each class. The 39 features were collected and the classification was performed with the Hidden Markov Model (HMM) resulting in an average accuracy of around 78%.

Another study using the temporal feature performed by (Basu et al., 2017) build a speech emotion classification model for anger, pleasure, boredom, neutral, anxiety, disgust, and fear. The speech corpus was the same as the research by Burkhardt et al. (2005) and they used 13 channels from MFCC as the features. The classification model uses a combination of Convolutional Neural Network and Recurring Neural Network (CNN-RNN). The accuracy of the combination model was up to 80%.

2.2 Research on Detecting Emotions in Text Data

Unlike the studies in speech data, research on emotion detection in Bahasa Indonesia for the text data has been done by various methods. Several prior studies will be discussed in this section including the research by The et al. (2015) which focuses on emotional tweets, and research by Muljono et al. (2016) which focuses on fairytale scripts, and research by Savigny and Purwarianti (2017) which focuses on YouTube comments.

The research conducted by The et al. (2015) aims to build a classification model of emotions in the text using 5 classes i.e. happy, love, anger, fear, and sadness. The text corpus was crawled from Indonesian-language tweets. The feature extraction includes various types of features i.e. N-Gram features such as Bigram and Unigram, linguistic features such as Part-Of-Speech (POS) Tagging, semantic features such as sentiments in the Indonesian lexicon, and orthographic features such as punctuations. The experiments were held in two stages. In the first stage, the tweets with emotion and without emotion were separated, then in the next stage, the classification model only works on the tweets with emotions. This study used several classification algorithms such as Maximum Entropy (ME) and Support Vector Machine (SVM). The evaluation results found that ME performance is superior to SVM with 72% accuracy.

The second study was conducted by Muljono et al. (2016) with the aim to build a classification model of emotions into six classes i.e. anger, sadness, joy, surprise, disgust, and fear. The corpus was obtained from a collection of fairy tales that are labeled manually with about 1200 data in the corpus. The pre-processing steps were stemming, stopword removal, and normalizing the data. Then the data is mapped into the feature vector using one-hot encoding with TF-IDF weighting. The machine learning model used is Naïve Bayes, Decision Tree J48, Support Vector Machine, and K-Nearest Neighbor. The experimental results show that SVM provides the best performance with an average accuracy of 85%.

The last study to be discussed was conducted by Savigny & Purwarianti (2017) with the aim of building a model of the classification of emotional emotions in 7 classes i.e anger, sadness, joy, surprise, disgust, fear and neutral. The text corpus was obtained from Indonesian-language comments on YouTube.The total data obtained was around 3000 comments with a relatively even distribution for each emotional class. The preprocessing methods were applied such as deleting duplicate characters, deleting numerical characters, and translating emoticons in the appropriate words. Next, each data in the corpus is changed in the vector representation using the word embedding algorithm with Continous Bag-Of-Words (CBOW) and Skip-Gram architecture. The classification methods used are vector averages, vector averages with Term Frequency - Inverse Document Frequency (TF-IDF) weighting, and Convolutional Neural Network (CNN). The results of the experiment show that CNN provides the best performance with an accuracy of around 73%.

3. Methodology

3.1 Data Collection

The dataset used for this study is collected from Indonesian widescreen movies because the dialogues in the movie mostly well-structured and rarely overlap. Moreover, the audio quality of the big screen movie is also very good given the clear voice and intonation of the actors in which the background noise is also rarely found.

The Indonesian movie selected in this study titled "Cek Toko Sebelah”. The reason for choosing only one movie to build our dataset is to limit the variations in our audio data such as the variation of speakers (actors), the variations of the speaker’s accents, and the variations caused by the difference of sound recording technique in the different movie. Ideally, the dataset is built with a limited number of variations with limited sentences as well, e.g. the research by Burkhardt et al. (2005) used a dataset using 5 actors and 5 actresses where each actor/actress can 10 different sentences in 7 emotional classes. However, the efforts to limit the number of films used have an impact on the amount of data that can be collected.

3.2 Initial Processing

The initial processing steps include audio and its respected transcript extraction, audio segmentation, and transcript tokenization. First, we did audio and transcript extraction from the film since the focus of the research was limited to speech and transcript data. The audio data has 1 hour and 44 minutes duration with a 48 kHz sampling rate along with the transcript data of 1776 sentences.

Before segmenting audio data, any particular segment of speech data must meet the set of standard eligibility conditions to be used in research i.e.

1. The sentences spoken in a segment of speech must be complete, uninterrupted in the middle or not an unfinished sentence;
2. Speech data must be properly spoken by 1 speaker, no other speaker may cut the sentence spoken in one segment of speech data
3. Speech data must be free from/minimally interrupted by background songs/noise that disturbs resulting in cannot be clearly heard.

Next, we performed segmentation by cutting the extracted audio data into pieces of speech data. At this stage, the transcript data is also used to help the segmentation process. Transcript data stores information about the start and end time of one particular speech that is accurate to milliseconds. By utilizing this information, the audio segmentation can be done automatically and accurately. The segmentation results are 1776 speech pieces and transcripts.

However, the errors may occur from the automatic segmentation process e.g. when the start or end time in the transcript data is inaccurate. Therefore, it needs to be checked manually. After a manual check, 965 pieces of speech are valid and ready to be labeled. Then, we tokenized the transcript of 965 speech data from the previous process. The transcript tokenization includes lowercasing all letters and removing all the special symbols such as periods, commas and numerical symbols.

3.3 Labeling

At this stage, 965 collected speech data and transcripts were labeled. The labeling process involves two annotators i.e. a 20-year-old man student and a 40-year-old woman who works as a tailor. Each annotator labeled the pairs of speech data and the respected transcript following some annotation guidelines as follows:

1. Play speech audio, focusing on aspects of actor/actress pronunciation such as intonation, rhythm, pauses, and speech emphasis.
2. Read the speech transcript, focus on the meaning of the sentence spoken by the speaker.
3. Determine the emotion class that is most appropriate for the pronunciation of the speech and the meaning of the sentence. For example, the sentence "This is very funny huh" which is spoken in high intonation indicates happiness.
4. Give labels according to the emotional class captured, the following is a list of labels used i.e. a) Label "0" is for happiness, b) Label "1" is for sadness, c) Label "2" is for anger, d) Label "3" is for neutral.

Then, both annotators did the annotation process for each pair of speech data and its transcript. We only include the data that is annotated with the same class by two annotators in our dataset while the remaining data will be discarded. At this stage, there are 775 data with the number of samples for each class shown in Table 1 below.

Happiness	Sadness	Anger	Neutral
186	181	138	270

Table 1: The Number of Samples per Emotion Class

3.4 Feature Extraction

We conducted two steps of feature extraction, i.e. feature extraction for speech data and feature extraction for transcript data. For speech data, the features used are Mel-scaled Frequency Cepstral Coefficient (MFCC), pitch, and Root Mean Squared Energy (RMSE). As for the transcript data, the feature used is the dense word vector representation with the word embedding approach. In addition, statistical computing is also performed on the MFCC, PITT, and RMSE features so that it can be used on non-temporal models.

3.4.1 Feature Extraction in Speech Data

The first feature taken from speech data is the MFCC feature. The MFCC maps sound signals on a Mel scale that is compatible with the human hearing system. The MFCC feature of the sound signal has a dimensionality between 13 to 26 dimensions for each frame, but in this study, a more general variation with 13 dimensions for each frame was used. The second feature taken from speech data is the RMSE feature. RMSE provides cumulative energy information on speech. RMSE is used to measure the loudness of a sound signal. The RMSE feature is a 1-dimensional vector for each frame. The last feature taken from speech data is the pitch feature. The pitch feature represents the characteristics of the sound from a frequency perspective so that it is good to answer questions such as "what is the average frequency of an angry man's voice?" or "what is the average frequency of a woman's voice that is sad?". Pitch features are 1-dimensional vectors for each frame.

3.4.2 Feature Extraction in Transcript Data

Transcript data is represented in the dense word vector using the word embedding algorithm Continuous Bag-Of-Words (CBOW) with Word2Vec. To convert sentences into word vectors, it is necessary to develop a word embedding model based on data with similar characteristics. Therefore, 5,572 additional transcripts from other Indonesian big-screen films were also collected to build the Word2Vec model. Then, we will extract the vector representation for each word in the transcript sentence. Each word vector in one sentence will be calculated on average or we may call it the average word vector. The calculation of averages also considers the TF-IDF scores for each word.

3.4.3 Statistics Features on Speech Data

Features in speech data need to be represented in statistical measures so that they can be used by non-temporal models. Therefore, it is necessary to transform the speech data features which were originally in the form of sequences into a numerical value following certain statistical measures. In this study, the statistical measures that will be used are the average and standard deviations.

3.5 Model Development

The models used include non-temporal and temporal models for speech data and embedding models for transcript data. Also, a joint model will be discussed that can receive speech data input as well as transcript data.

3.5.1 Development of Non-Temporal Models

The algorithm used in the model is the Artificial Neural Network (ANN). ANN is composed of one layer for input and one activation layer in the form of a sigmoid function for output. The model will accept statistical representations (averages and standard deviations) of each feature (MFCC, pitch, and RMSE). In this study, hyperparameter tuning will be performed on the number of neurons in the input layer. We also performed feature selection by finding the best combination of the features above. The architectural detail used can be seen in Figure 1. The dense_25 represents the input layer that accepts a vector with dimension 30 and returns a vector with dimension 64 and dense_26 represents the sigmoid layer that receives a vector with dimension 64 and returns the numeric value in the form of the predicted emotional class probability.

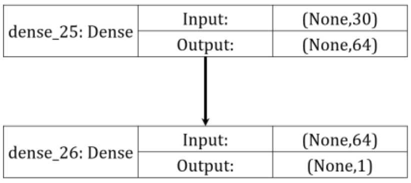

Figure 1: The Architecture of The Non-Temporal Model

3.5.2 Development of the Temporal Model

The algorithm used in the model is the Recurrent Neural Network (RNN) variation of Long Short Term Memory (LSTM). The model for the temporal model has a hidden layer as the addition to the input and activation layer. So, the model is composed of one LSTM layer for input, one hidden layer, and one activation layer in the form of a sigmoid function for output. The model will receive a temporal representation (sequence) of each feature. In this study, hyperparameter tuning will be carried out on the number of neurons in the input layer and hidden layer. We also performed feature selection by finding the best combination of the features above.

Figure 2 shows the detailed architecture. The lstm_50 is the LSTM input layer that accepts sequences of 300 vectors with each vector of dimension 15 and returns vectors with dimensions 128. The dense_99 is a hidden layer that accepts input vectors with dimensions 128 and returns vectors with dimensions 64. The dense_100 is a sigmoid layer that accepts a vector with dimension 64 and returns the probability of the predicted emotion class.

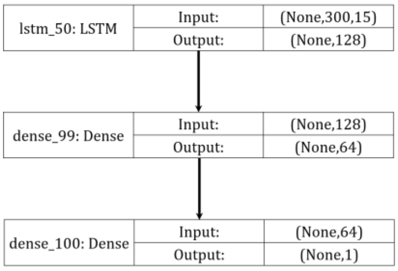

Figure 2: The Architecture of The Temporal Model

3.5.3 Embedding Model Development

The third model built is an embedding model for transcript data. The algorithm used in the model is ANN. ANN accepts average word vector input by weighting TF-IDF from the previous stage. ANN is composed of one input layer and an activation layer in the form of a sigmoid function as the output. In this study, hyperparameter tuning will be performed on the number of neurons in ANN. The detailed architecture used can be seen in Figure 3. The dense_5 represents the input layer which accepts a vector with a dimension of 150 and returns a vector with a dimension of 64. The dense_6 represents the sigmoid layer that receives a vector with dimension 64 and returns the probability of the predicted emotional class.

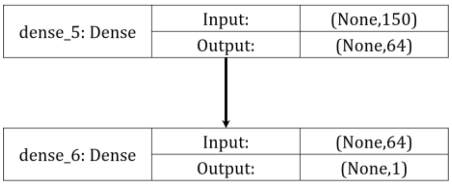

Figure 3: The Architecture of The Embedding Model

3.5.4 Building a Combined Model

The last model was built as a joint model. The purpose of building this model is to utilize both the features of the speech and the features of the transcript in one model. The combined model was built in 2 variations, namely: (1) a combination of non-temporal models and embedding models and (2) a combination of temporal models and embedding models.

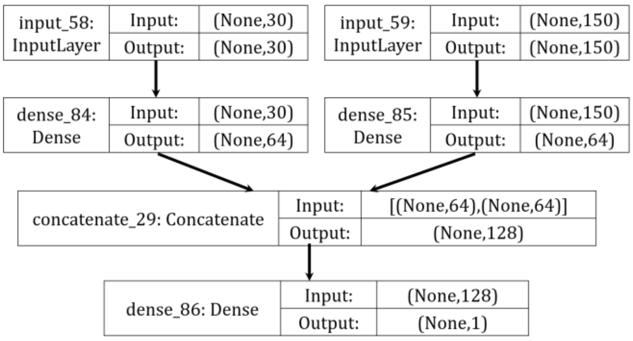

Figure 4: The Architecture of The Combined Model I (Non-Temporal and Embedding Model)

Figure 4 shows the detail of the combined model of non-temporal and embedding. input_58 states the input layer that receives a vector of speech data with a dimension of 30. input_59 expresses an input layer that accepts a vector of transcript data with a dimension of 150. dense_84 is the hidden layer that receives the inputs from the input_58 layer and returns a vector with dimension 64. dense_85 is a hidden layer that receives a vector from the input_59 layer and returns a vector with dimension 64. concatenate_29 accepts two output vectors from dense_84 and dense_85 and returns a concatenated vector with a dimension of 128. dense_86 is a sigmoid layer that receives a vector with dimensions 128 and returns the probability of the predicted emotion class.

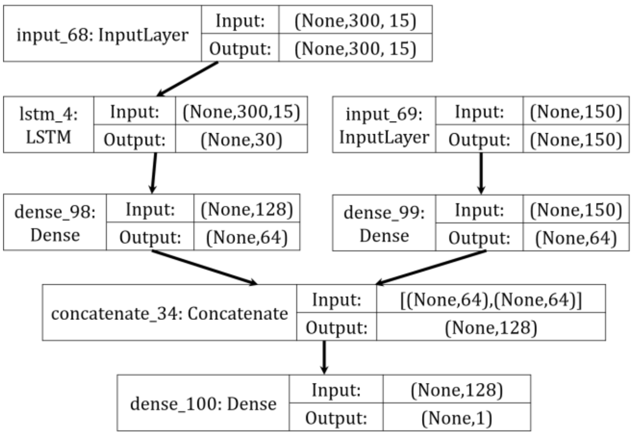

Figure 5: The Architecture of The Combined Model II (Temporal and Embedding Model)

Figure 5 describes the detail for the architectural model of the combination of temporal and embedding. input_68 states the input layer which receives a sequence of 300 vectors, each of which has dimension 15 of speech data. input_69 states the input layer that receives a vector with the dimension of 150 from the transcript data. lstm_4 is an LSTM layer that receives input sequences from the input_68 layer and returns a vector with a dimension of 128. dense_98 is the hidden layer that receives vector from layer lstm_4 and returns a vector with a dimension of 64. dense_99 is the hidden layer that receives vector from input_69 layer and returns a vector with the dimension of 64. concatenate_34 accepts two output vectors from dense_98 and dense_99 and returns a vector with a dimension of 128. Finally, dense_100 represents the sigmoid layer that receives a vector with a dimension of 128 and returns the probability of the predicted emotion class.

3.6 Design of Training Scenarios

According to Yu et al. (2001), it would be difficult to build a good model of each emotional class with only 200 data, thus the recommended alternative is to use a one-vs-rest scenario. The one-vs-rest scenario will divide the dataset into 2 classes, i.e. the class you want to study and the other class. For example to build a model to recognize anger, 130 data from anger classes and 130 data from other classes (labeled as "others") are prepared. With the one-vs rest scenario, the model can learn the characteristics of each emotional class deeper and the model is expected to provide better results. In addition, training parameters such as loss and activation functions need to be defined. The study will use the Binary Cross-entropy function for loss function and Root Mean Square Propagation (RMSProp) function for the activation function.

3.7 Design of Testing Scenarios

The research will involve two testing scenarios i.e. the testing scenario for each one-vs-rest model and the testing scenario for classification which directly recognizes the four classes of emotions. For the testing scenario of each model, the metric used to measure performance is accuracy. Accuracy provides information related to the proportion of correct predictions by the model of the testing data. For testing scenarios that can directly recognize the four emotions following the following mechanism, each test data will be run on the four one-vs-rest models and the emotional class is chosen which gives the highest probability as a label of the test data. The one-vs-rest model chosen for each emotion class is the best variation according to the one-vs-rest model testing scenario that was run before. The metrics used were precision, recall, and F1-score.

4. Results and Analysis

We describe the result of our experiments into two different sections i.e: non-temporal models and temporal models. For each section, we will explain the result of experiments in feature selection and hyperparameter tuning.

4.1 Non-Temporal Models

4.1.1 Feature Selection

The development of non-temporal models begins with feature selection. First, we trained the model for each possible feature subset of the 3 main research features i.e. MFCC, pitch, and RMSE. Then each model will be compared in performance through the accuracy metric. The subset of features of the model with the best accuracy will be taken and used as model candidates for tuning. The results of feature selection experiments can be seen in Figure 6. The accuracy shown is the accuracy of the test set.

According to Figure 6, the combination of the three features (MFCC, pitch, and RMSE) has the best accuracy for each model. The combination of the three features provides more information so that the model is able to classify more accurately. Furthermore, we can see that the MFCC feature alone gives a fairly high accuracy when compared to the pitch or RMSE features due to the MFCC feature consisting of 13 channels compared to the other two features which only consist of 1 channel. The combination of features with the best accuracy proceed to the tuning stage.

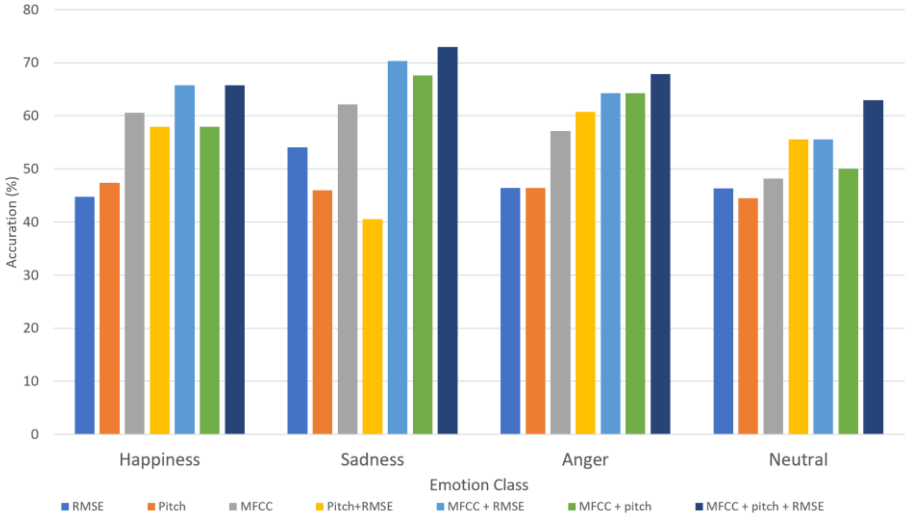

Figure 6: The Experiment Results on the Feature Selection of Non-Temporal Models

4.1.2 Hyperparameter Tuning

We performed tuning to the number of neurons. The numbers of neurons used in this experiment are 32, 64, 128, 256, and 512. After conducting several experiments, the best number of neurons for models in happiness is 64, sadness is 128, anger is 256, and neutral is 64. For neurons 512, the accuracy of each model variation tends to decrease. Goodfellow et al. (2016) say that a model with a high capacity (number of neurons) will be able to solve more complex problems, but a higher capacity than the problem needs can lead to overfitting. The best accuracy falls on the sadness model with an accuracy of 72.97% in the test set.

4.2 Temporal Models

4.2.1 Feature Selection

The feature selection method is the same as in the non-temporal model which testing each feature subset of 3 main features (MFCC, pitch, and RMSE) using the one-vs-rest model for each emotional class. The results of feature selection experiments can be seen in Figure 7.

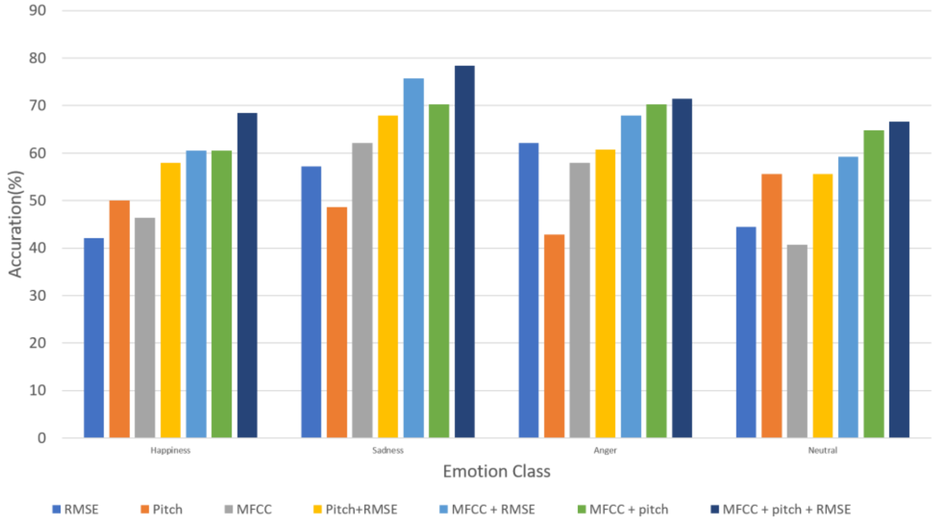

Figure 7: The Experiment Results on the Feature Selection of Temporal Models

The results of the experiments are also not much different from the non-temporal model where the combination of all three features provides the best performance on all variations of the model. Each model also only needs around

12 to 20 iterations to be converged according to the early stopping mechanism, meaning that in 12 to 20 iterations, the accuracy of the model in the validation set does not change significantly and the model starts showing signs of overfitting so the training process is stopped. Then, the most accurate model variations enter the hyperparameter tuning step by tuning of the number of neurons.

4.2.2 Hyperparameter Tuning

The most accurate model variations enter the hyperparameter tuning step by tuning of the number of neurons. Variations in the number of neurons used are 32, 64, 128, 256, and 512. The best number of neurons for the happiness is 64, sadness is 128, anger is 128, and neutral is 64. The highest accuracy is achieved by sadness class which is around 81.08% for test set accuracy.

4.3 Embedding Models

The next experiments are to build the embedding model. Embedding model experiments only involve the hyperparameter tuning process considering the features used are only one type i.e. average word vector with TF-IDF weighting. Variations in the number of neurons are 32, 64, 128, 256, and 512. The best number of neurons for happiness is 128, sadness is 256, anger is 256, and neutral is 64. The best accuracy is obtained for anger with the accuracy of 60, 71% in the test set.

4.4 Combined Models

The last experiments in model development are to build a combined model that only involves the tuning process and does not involve the selection of features considering the results of non-temporal and temporal experiments show that optimal results are obtained using all three features and the embedding model has only one feature representation. Tuning is performed on the number of neurons in both sub-models (non-temporal, temporal, and embedding). The best accuracy in the combined model I (non-temporal & embedding model) for the happiness is 76.31%, for the sadness is 78.37%, for the anger is 75.00%, and for the neutral is 64.81%. For the combined model II (temporal & embedding), the best accuracy is 71.05% for happiness, 86.46% for sadness, 82.14% for anger, and 68.51% for neutral.

4.5 Summary of Results and Analysis

Model	**Accuracy (%)**			
	Happiness	Sadness	Anger	Neutral
Non-temporal	65.78	72.97	71.42	62.96
Temporal	68.42	81.08	78.57	66.66
Embedding	60.52	56.75	60.71	48.14
Combined model I (non-temporal + embedding)	76.31	78.37	75.00	64.81
Combined model II (temporal + embedding)	71.05	86.46	82.14	68.51

Table 2: Summary of Experimental Results

The summary of our experimental results from the 5 variations of the model can be seen in Table 2. There is also another testing scenario mentioned in Section 3.7 to measure the accuracy of the classification which directly recognizes the four classes of emotions. This scenario obtained a precision of 64.66%, a recall of 66.79%, and an F1-score of 64.83%. The confusion matrix in Table 3 shows the detail evaluation of this direct classification scenario.

Prediction Actual	Happiness	Sadness	Anger	Neutral
Happiness	15	4	3	2
Sadness	1	11	0	3
Anger	2	1	14	1
Neutral	4	5	4	15

Table 3: Confusion Matrix of The Direct Classification

From the results of our experiments, we formulate the following analysis:

- The accuracy of each emotional class in the temporal model is always higher than the non-temporal model. This shows that the sequence representation is better than using statistical measures to represent the audio. Furthermore, since RNN can capture temporal relations, the model does not only classify each frame independently but also learn the relationship between frames in the sequence.
- The accuracy of embedding models is lower compared to temporal or non-temporal models. This shows that emotions are easier to detect in speech than in transcripts. Linking it to the valence-arousal model (Barrett and Russell, 1999), emotions in speech are more easily mapped on by analyzing the intensity of the speaker's speech. While on the transcript data, emotions can only be captured by understanding the correlation of words with the emotional class and the context of the transcripts.
- The accuracy of the combined model is relatively higher compared to a single model (non-temporal, temporal, or embedding only). This phenomenon proves that the combined information from transcript data and speech data synergizes with each other to provide information about the emotion.

- In the happiness model, misclassification generally falls on the class of anger and sadness. It was found that misclassification which fell on the anger class was generally due to the delivery of high intensity as in one of the speeches with the transcript "Hooray! we won! Our shop wins!" which is blurted out. Another mistake arises from the use of words with a negative sentiment as in the word "dies" from the transcript "Cockroach, you see! When it dies, it is upside down, haha". Even though the main context of the transcript is intended as a joke. The misclassification that falls on sad emotions is due to the pronunciation with the intensity that is too low as in one of the speech data with the transcript "Hey, long time no see, where are you going? Let's go with me" said in a friendly tone so that the model mistakenly captures its emotion as sadness.

- In the sadness model, misclassification most often falls on neutral emotions. Most of the data that was incorrectly classified as having a unique pattern of sadness is emphasized on the semantics of the transcript by using flat-pitched pronunciation. However, from the experimental results, the model tends to utilize information from speech as the main consideration because it provides the best overall performance so that the model gives a higher "weight" to a flat pronunciation rather than the semantics of the transcript. For example in the transcript "But that's because of the last memories of my mom ... which I can hold ..." with a relatively flat delivery, the model considers the data to have neutrality emotions even though the real meaning of the transcript is expressing sadness.
- In the anger emotions model, misclassification tends to fall on happiness emotions. The inspection of the misclassified data shows that the speech is delivered with an intensity that is not as high as other speech data showing anger emotions. Besides, the data does not use groups of words that generally appear on anger emotions such as swear words or words with other negative connotations. For example in the speech data with the transcript "You are fired!" with a delivery that is not loud enough has a resemblance in terms of pronunciation with our previous example that shows happiness ("We won! Our store won!"). This expression also does not contain words with negative connotations so the model assumes that the data contains happiness.
- In the neutral emotion model, misclassification most commonly occurs in either one-vs-rest scenarios or direct classification scenarios. Through listening to the speech data and reading the transcript data from the neutral class, it was found that a lot of data in the neutral class is still ambiguous. For example, there are speech data with a neutral emotion but at the beginning of the speech spoken in a high tone so that it is classified as anger or speech data with a low intensity so that it is classified as sad emotions. In terms of transcripts, the data in the neutral class do not have certain words that imply neutrality. It is in fact in contrast to the happiness class that usually is expressed using the words of praise or gratitude, or in the anger class that is closely related to swearing words.

5. Conclusion and Future Works

This research has aimed to develop the speech emotion detection model in Bahasa Indonesia. Our main objectives are to understand how to represent speech data into features to obtain the best model and use speech transcripts to improve model performance. Thus, we experimented with our dataset using five model variations for the 4 emotional classes i.e. happiness, sadness, anger, and neutral.

The best experimental results for each emotion are described as follows: (1) the best accuracy for the class of happiness is 68.42% using speech data only with a temporal approach and 76.31% using speech data and transcripts with a combined non-temporal and embedding approach; (2) the best accuracy for the class of sadness is 81.08% using speech data only with the temporal approach and 86.46% using speech data and transcripts with a combined temporal and embedding approach; (3) the best accuracy for the class of anger is 78.57% using speech data only with the temporal approach and 82.14% using speech data and transcripts with a combined temporal and embedding approach; (4) the best accuracy for the class of neutral emotion classes 66.66% using speech data only with the temporal approach and 68.51% use speech data and transcripts with a combined temporal and embedding approach. In addition, experiments were also carried out for direct classification in four classes of emotions and the precision results were 64.66%, recall of 66.79%, and the f1-score was 64.83%.

The conclusion we may draw from our results is that the temporal representations provide better accuracy than statistical representations. The experimental results also exhibit the impact of the combined model using speech data as well as transcript data as a feature to improve the model performance because the model gets information from two different perspectives.

There is still a lot of potential for the development of the use of RNN such as by trying variations of the Gated Recurrence Unit (GRU) or LSTM bidirectional variations. Other suggestions are aimed at the stage of building models that can be explored more deeply. Furthermore, the use of additional layers such as the dropout layer for regularization can be applied to models to avoid overfitting and obtain models with better generalization capabilities.

6. Bibliographical References

Barrett, L. F., & Russell, J. A. (1999). The Structure of Current Affect: Controversies and Emerging Consensus. Current Directions in Psychological Science. SAGE Publishing.

Basu ,S., Chakraborty, J., & Aftabuddin, M. (2017). Emotion recognition from speech using convolutional neural network with recurrent neural network architecture. 2nd International Conference on Communication and Electronics Systems (ICCES) (hal. 333-336). IEEE.

Burkhardt, F., Paeschke, A., Rolfes, M.A., Sendlmeier, W.F., & Weiss, B. (2005). A database of German emotional speech. INTERSPEECH. Semantic Scholar

Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press.

Keshtiari, Niloofar. (2015). Recognizing emotional speech in Persian: a validated database of Persian emotional speech (Persian ESD). Behavior Research Method. Springer.

Ko, Y., Hong, I., Shin, H., & Kim, Y. (2017). Construction of a database of emotional speech using emotion sounds from movies and dramas. International Conference on Information and Communications (ICIC) (page 266-267). IEEE.

Livingstone, S. R. & Russo, F. A. (2018). The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS): A dynamic, multimodal set of facial and vocal expressions in North American English. Public Library of Science (PLOS).

Lubis, N., Lestari, D., Purwarianti, A., Sakti, S., & Nakamura, S. (2014). Emotion recognition on Indonesian television talk shows. IEEE Spoken

Language Technology Workshop (SLT) (page 466-471). IEEE.

Muljono, Winarsih, N. A. S., & Supriyanto, C. (2016). Evaluation of classification methods for Indonesian text emotion detection. International Seminar on Application for Technology of Information and Communication (ISemantic) (hal. 130-133). IEEE.

Safitri, N. E., Zahra, A., & Adriani, M. (2016). Spoken Language Indentification Using Phonotactic Methods on Minangkabau, Sundanese, and Javanese. 5th Workshop on Spoken Language Technologies for Under-resourced Languages (SLTU) (page 182-187). Science Direct.

Savigny, J., & Purwarianti, A. (2017). Emotion classification on youtube comments using word embedding. International Conference on Advanced Informatics, Concepts, Theory, and Applications (ICAICTA) (hal. 1-5). IEEE.

The, J. E., Wicaksono, A. F., & Adriani, M. (2015). A two-stage emotion detection on Indonesian tweets. International Conference on Advanced Computer Science and Information Systems (ICACSIS) (hal. 143-146). IEEE.

Vania, C. & Adriani, M. (2011). The effect of syllable and word stress on the quality of Indonesian HMM-based speech synthesis system. International Conference on Advanced Computer Science and Information Systems (ICASIS) (hal. 413-418). IEEE.

Wanagiri, M. Z. & Adriani, M. (2012). Developing and Analyzing ASR System for Accented Indonesian Speech. The 15th Oriental COCOSDA Conference. IEEE.

Wunarso, N. B., & Soelistio, Y. E. (2017). Towards Indonesian speech-emotion automatic recognition (I-SpEAR). 4th International Conference on New Media Studies (CONMEDIA) (page 98-101). IEEE

Yu, F., Chang, E., Xu, Y. Q., & Shum, H. Y. (2001). Emotion Detection from Speech to Enrich Multimedia Content. Lecture Notes in Computer Science (LNCS) (hal. 550-557). Springer.