MULTIMODAL APPROACH FOR EMOTION RECOGNITION USING DEEP LEARNING

RESEARCH

Kenny

2340081090

Graduate Program in Computer Studies
STUDY PROGRAM MASTER OF INFORMATION TECHNOLOGY
UNIVERSITAS BINA NUSANTARA
JAKARTA
2022

MULTIMODAL APPROACH FOR EMOTION RECOGNITION USING DEEP LEARNING

RESEARCH

Kenny

2340081090

Thesis as Graduation Prerequisite To
Obtain Master Title
MASTER OF COMPUTER SCIENCE
On
Graduate Program
UNIVERSITAS BINA NUSANTARA

MULTIMODAL APPROACH FOR EMOTION RECOGNITION USING DEEP LEARNING

RESEARCH

Kenny

2340081090

Supervisor:
Ir. Andry Chowanda, Ph.D., MBCS.
Date: 3 Januari 2022

STATEMENT

PERNYATAAN

With this I,

Name : Kenny

Student ID : 2340081090

Thesis title : MULTIMODAL APPROACH FOR EMOTION RECOGNITION USING DEEP LEARNING

Hereby grant to my school, Bina Nusantara University, the non-exclusive right to archive, reproduce, and distribute my thesis, in whole or in part, whether in the form of printed and electronic formats.

I acknowledge that I retain exclusive rights of my thesis by using all or part of it in the future work or outputs, such as article, book, software, and information system.

Memberikan kepada Universitas Bina Nusantara hak non-eksklusif untuk menyimpan, memperbanyak, dan menyebarluaskan tesis karya saya, secara keseluruhan atau hanya sebagian atau hanya ringkasannya saja, dalam bentuk format tercetak dan atau elektronik.

Menyatakan bahwa saya, akan mempertahankan hak eksklusif saya, untuk menggunakan seluruh atau sebagian isi tesis saya, guna pengembangan karya di masa depan, misalnya bentuk artikel, buku, perangkat lunak, ataupun sistem informasi.

Jakarta, 3 Januari 2022

Kenny

STUDENTS STATEMENT

HALAMAN PERNYATAAN

I, Kenny, student id of 2340081090, truly acknowledge that my thesis titled "MULTIMODAL APPROACH FOR EMOTION RECOGNITION USING DEEP LEARNING" is my concept and project result with guidance from a supervisor.

I, also truly acknowledge that the content of this thesis is not copied and are not from other people's work, except citation from literature or written interview results which have already been written in the reference list.

I hereby conclude my acknowledgement which were made true and am willing to sanction if the contents are not true.

Saya, Kenny, dengan NIM 2340081090, menyatakan dengan sebenar benarnya bahwa tesis saya yang berjudul "PENDEKATAN MULTIMODAL UNTUK PENGENALAN EMOSI MENGGUNAKAN DEEP LEARNING" adalah konsep saya dan hasil dari projek yang dibuat dengan bimbingan dari seorang *supervisor*.

Saya juga menyatakan dengan sebenarnya bahwa isi tesis ini tidak merupakan jiplakan dan bukan pula hasil karya dari orang lain, terkecuali kutipan dan atau hasil wawancara yang digunakan sebagai acuan yang telah dituliskan didalam daftar pustaka.

Demikian pernyataan ini saya buat dengan sebenar benarnya dan saya bersedia menerima sanksi bila ternyata pernyataan saya ini tidak benar.

Jakarta, 3 Januari 2022

Student,

Kenny

2340081090

PREFACE

Thank to Almighty God who has given His blessing to the writer for this thesis entitled “MULTIMODAL APPROACH FOR EMOTION RECOGNITION USING DEEP LEARNING” can be finished. This thesis is written as one of the pre-requisites to finish the graduate program and achieve the title Master of Computer Science in Bina Nusantara University.

On the course of the writing of this thesis, the writer is supported by a number of people, be it from a knowledge, moral, financial, or social perspective. A big thank you is given to:

- Mr. Prof. Dr. Ir. Harjanto Prabowo, M.M. as Bina Nusantara University rector that lead and manage the whole organization in the university
- Mr. Dr. Sani Muhamad Isa, S.Si., M.Kom. and BGP team that manage the flow of the Master Track course, especially thesis creation schedule that ensure the thesis is finished in time
- Mr. Ir. Andry Chowanda, Ph.D., MBCS. as the thesis guidance counsellor that has answered all thesis related questions that the writer asked
- Lecturers of Master track course, as the source of knowledge which became the foundation of the thesis
- The writer’s families, that has supported the writer through all possible means from start until this thesis is finished written
- Friends of the same Master Track program, especially those that has helped give feedbacks and discuss about the writing of this thesis

The writing of this thesis is not free of limitations. Even though the writer has tried his best to write the thesis without errors, `he apologize if there is a mistake found in the thesis. Please feel free to contact the writer to give feedback which can be used to further improve the thesis.

Jakarta, 3 January 2022,

Student,

Kenny

2340081090

Thesis title: MULTIMODAL APPROACH FOR EMOTION RECOGNITION USING DEEP LEARNING

ABSTRACT

Emotion recognition has been a very challenging topic. Multimodality approach in emotion classification has been used in various research to improve the recognition performance. Nevertheless, there is a lack of understanding between how the multimodality affects the performance of the model. This paper use IEMOCAP as dataset and create several unimodal model and multimodal model resulted in combination of the top unimodal model for emotion recognition. The fusion method of the different modalities is based on feature level fusion with the help of concatenation layer. After evaluating the unimodal and multimodal models, this paper analyzes the importance of every unimodality involved and its implication to multimodality built. The top result of F1 score in unimodal models achieve 0.675 which is the BERT base model and the top result of F1 score in all multimodal models is 0.765 which is the combination between BERT base and proposed CNN model. This paper also talks about the possibility of further research by applying more dataset, more complex model, and more technique.

Keywords: Emotion Recognition, Multimodal Classification, Deep Learning, Text classification, Image classification, Audio classification

ABSTRAK

Pengenalan emosi telah menjadi topik yang sangat menantang. Pendekatan multimodalitas dalam klasifikasi emosi telah digunakan dalam berbagai penelitian untuk meningkatkan kinerja pengenalan. Namun demikian, ada kekurangan dalam pemahaman antara bagaimana multimodalitas mempengaruhi kinerja model. Makalah ini menggunakan IEMOCAP sebagai dataset dan membuat beberapa model unimodal dan model multimodal yang menghasilkan kombinasi model unimodal teratas untuk pengenalan emosi. Metode kombinasi dari beberapa modalitas yang berbeda didasarkan oleh metode penggabungan tingkat fitur dengan bantuan lapisan gabungan. Setelah mengevaluasi model, makalah ini menganalisis pentingnya setiap unimodal yang terlibat dan implikasinya terhadap multimodal yang dibangun. Hasil teratas skor F1 pada model unimodal mencapai 0.675 yang merupakan model BERT-base dan hasil teratas skor F1 pada semua model multimodal adalah 0.765 yang merupakan kombinasi antara basis BERT dan model CNN yang diusulkan. Makalah ini juga berbicara tentang kemungkinan penelitian lebih lanjut dengan menerapkan lebih banyak dataset, model yang lebih kompleks, dan lebih banyak teknik.

Keywords: Pengenalan Emosi, Klasifikasi Multimodal, *Deep Learning*, Klasifikasi teks, Klasifikasi gambar, Klasifikasi audio

Table of Contents

LIST OF TABLES

LIST OF FIGURES

LIST OF EQUATIONS

CHAPTER I
INTRODUCTION

1.1. Background

Emotion has always been an important part of daily life. It is often used to give meaning to events, so those events would be more meaningful. Having an emotion helps to coordinate interpersonal relationships between humans. Various research on emotion has been done over the years, many of which have attempted to classify emotions based on various factors. These research include William James's 4 basic emotions (1890), Paul Ekman's 6 basic emotions (1990), Richard and Lazarus's 15 emotions (1996), Berkeley's 27 emotions (2017), and many more.

Emotion can be expressed in many ways, such as gestures, facial expressions, texts, and voices. It is a very interesting subject and has been researched since the 19th century by experimental psychologists (Thanapattheerakul et al., 2018). With the breakthrough of Artificial Intelligence in past decades, it is possible to do an emotion classification task using the provided data such as facial expression, speech, body parts motions, and text to recognize the emotion.

Classification of emotion in Artificial Intelligence has been improving over the years, starting with the usage of Neuroimaging, Autonomic Nervous System (ANS), Facial Expression – Facial Action Coding System (FACS), and Speech Emotion Recognition (SER) (Thanapattheerakul et al., 2018). Most of the recent classification implements deep learning classification. There are several feature extraction and layer combinations in a model for images, textual, and speech

datasets. Although the usage of a single modality can create a prediction model with good accuracy, some approaches can be used to improve the overall performance of the model.

With the rapid advancement of computational power, multi-modality is introduced. Multi-modality is an approach that makes use of multiple inputs of different types instead of one modality. Some scenarios utilize multi-modality to improve the robustness of a model, such as detecting a person's emotion in recorded/real-time, their facial expressions, gestures, speeches, and the textual information is spoken can be observed. Many research has proven that the implementation of a multi-modality approach increases the overall performance of the model such as (Thang Duong et al., 2017), (Chaparro et al., 2018), and (Yoon et al., 2019). Although multi-modal approach increases the model's performance, there is still a lack of knowledge of the best combination of modality and each modality roles in the improvement.

This research aims to study the importance of each modality and the best combination that can provide the best accuracy for the model. This research uses Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM), and Bidirectional Long Short-Term Memory (Bi-LSTM) model for the motion capture modality. For the text modality, this paper uses Bidirectional Encoder Representations from Transformers (BERT) base, robustly optimized BERT approach (RoBERTa), and DistilBERT. CNN, Bi-LSTM, and a mix of CNN and Bi-LSTM architecture are used for Speech modality. In the late stage of the research, multi-modality models are presented by a combination of features from

the best model of each modality. A classification dense layer will then be applied to every model..

The main contributions of the thesis are as follows: (1) discover the value of each modality used in multi-modal and the combination. (2) Propose combinations of feature-fusion models that are created by the combination of 3 different modalities (facial, text, and speech).

1.2. Problem Formulation

Based on the introduction, the problems appeared in this research are represented using these questions:

1. Is there any new characteristic findings in multimodal model with the newly state-of-the-art unimodal models?
2. What is the modalities combination that gives best performance multimodality?

1.3. Goals and Benefits

The goals of this research are as follows:

1. Develop and explore newly State of the art unimodality and multimodality models which can predict emotions with better performance
2. Evaluate the multimodal result and analyze the importance of every modality

The benefits of this research are as follows:

1. Produces several multi-modal models with applied state-of-the-art unimodal model

2. Understand the characteristic of multimodality model and unimodalities used based on the results

1.4. Scope of Work

Scope of work that is needed to be done for the research are as follows:

1. The dataset used in this research are IEMOCAP with only 4 emotions as classes (Happy, Angry, Sad, and Neutral)
2. Test and validate the proposed model on the IEMOCAP speech dataset, which is an acted dataset. Hence, the result may differ ever so slightly when compared with the usage of real dataset.
3. The models used in the research is only the one proposed in the thesis

CHAPTER II
LITERATURE REVIEW

2.1. Emotion Classification

Emotion classification with text has been a challenging problem for a long time. There have been various research aiming for more accurate emotion detection from many languages. There have been various ways to tackle emotion classification problems, most of them use machine learning and deep learning to predict the emotion. Many different theories classify emotion as 4, 5, or 6 different emotions. The recently famous emotion classification is by Paul Ekman's six basic emotions. Classification of Paul Ekman's six emotions are happy, sad, fear, anger, surprise, and disgust. There is various research that used Ekman's 6 emotions for the emotion classification, such as (Kirange D. K*, 2012) and (Becker et al., 2017). **Figure 2.1** show the example of each emotion in Paul Ekman's 6 emotions.

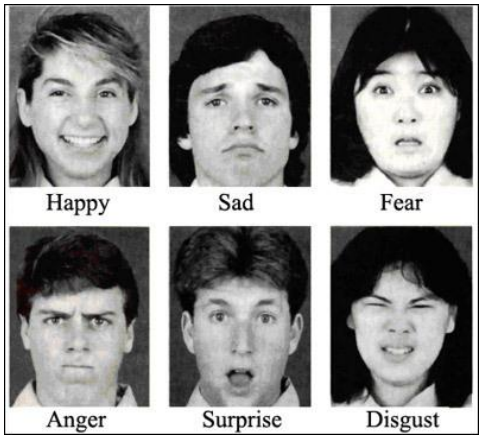

Figure 2.1 Facial Expression
(Nitsch & Popp, 2014)

2.2. Machine Learning

In Artificial Intelligence, there is a concept where machines start to learn from data without being programmed. This concept is called Machine Learning. Machine Learning can be broadly defined as computational methods using experience to improve performance or to make accurate predictions (Mohri et al., 2013). There are many applications of machine learning, such as image recognition, speech recognition, emotion recognition, stock prediction, and many else. Machine learning, by technique, can be further divided into two categories, which are supervised learning and unsupervised learning.

Supervised learning builds a model based on existing data and labels and uses it to predict a label for new input, while Unsupervised learning is used to find hidden patterns in a set of data. Supervised learning applications are classification and regression, where both predict something, while one of the common unsupervised learning applications is clustering, where we find a specific pattern/group within a collection of data. The differences between supervised learning and unsupervised learning can be seen **Figure 2.2**.

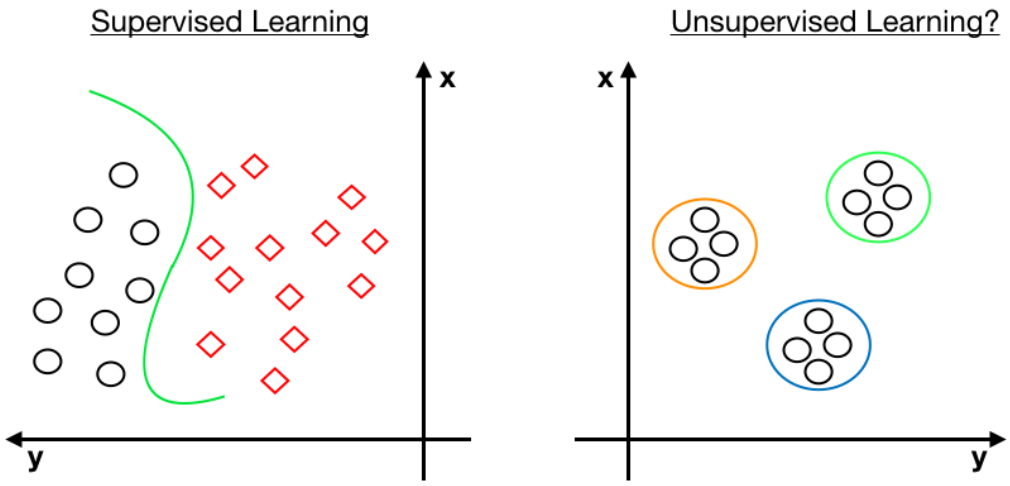

Figure 2.2 Difference of supervised and unsupervised
(Odaibo, 2019)

2.3. Deep Learning

Deep Learning allows computational models that are composed of multiple processing layers to learn representations of data with multiple levels of abstraction (Lecun et al., 2015). There are differences between Deep Learning and Machine Learning, such as amounts of layers and the existence of a feature. Machine Learning creates a model based on supervised data whereas Deep Learning structured the network to imitate a brain that can learn patterns and extract features from the big amount of data to create the optimal model.

Deep Learning combines advances in computing power and neural networks with many layers to learn complicated patterns in large amounts of data. It is an extension of a classical neural network and uses more hidden layers so that the algorithms can handle complex data with various structures (Muniasamy & Alasiry, 2020).

2.4. Optimizer algorithm

Optimizer algorithms are used by neural networks to control the weight change and learning rate. There are several optimizer algorithms, such as Gradient Descent, Momentum, Adam (Adaptive Moment Estimation), etc.

2.4.1. Gradient Descent

Gradient Descent is the most basic and most used optimized algorithm. Gradient Descent algorithm calculates the weight direction based on the error to get the cost function to reach the maximum/minimum value. Gradient Descent has a few variations, such as Batch Gradient Descent, Mini Batch Gradient Descent, and Stochastic Gradient Descent.

2.4.1.1. Batch Gradient Descent

Batch Gradient Descent is called vanilla gradient descent. Batch Gradient Descent updates the model parameters after all training data has been evaluated. Batch Gradient Descent has an advantage in stable error gradient and stable convergence. The disadvantage of Batch Gradient Descent is that most likely stuck at the convergence which is not the best that the model can achieve.

$$w_{t+1} = w_t - a.\nabla_w J(w) \quad (2.1)$$

In **equation (2.1)**, the new value of weight (w) is obtained by reducing the last weight with gradient value, which is obtained by derivate value of weight multiplied by the learning rate.

2.4.1.2. Stochastic Gradient Descent

Stochastic Gradient Descent updates the model parameters for every training example. Stochastic Gradient Descent advantage is that it cost less time to the convergence and requires less memory because it does not need to store the loss function's value. The disadvantage is that the change on the parameters have high variance and have a high computational cost.

$$w_{t+1} = w \quad - a.\nabla_w J\left(x^i, y^i; w\right) \tag{2.2}$$

In **equation (2.2)**, the value of new weight is obtained by reducing the last weight with a derivative of weight in every data multiplied by the learning rate.

2.4.1.3. Mini Batch Gradient Descent

Mini Batch Gradient Descent is a combination of the advantage of Stochastic Gradient Descent and Batch Gradient Descent. It divided the training data into a different batch and update the model parameter after every batch is done.

$$w_{t+1} = w_t - a.\nabla_w J\left(x^{\{i:i+b\}}, y^{\{i:i+b\}}; w\right) \tag{2.3}$$

In **equation (2.3)**, the mini-batch gradient descent obtained a new weight from the last weight minus a derivative of weight calculated with the current batch of data multiplied by the learning rate.

2.4.2. Momentum

Momentum was invented to fix the Stochastic Gradient scent disadvantage in high variance and soften the convergence. In its equation, an extra variable 'γ'

called momentum when updating the weight. In applying momentum, there is a variable called velocity (V) that decides the growth speed of a weight.

$$V_{t+1} = \gamma * V_t + \ a.\nabla_w J\left(x^i, y^i; w\right) \quad (2.4)$$

$$w_{t+1} = \ w_t - V_{t+1} \quad (2.5)$$

In equation (2.4), the value of velocity is obtained by the momentum variable value multiplied by the direction and added by the derivative of the weight in every data. Equation (2.5) show the new weight is the result of the last weight minus the velocity value.

2.4.3. Adam

Adam Optimizer algorithm is an algorithm proposed aimed at efficient stochastic optimization that only requires first-order gradients with little memory requirement (Kingma & Ba, 2015). The idea behind Adam is to decrease the velocity so that it does not go pass through over minimum/maximum value. Adam Optimizer algorithm computes 2 Moments M(t) and V(t) which are called Mean and uncentered variance, respectively. Equation 2.6 shows the formula for mean and equation 2.7 show the formula for uncentered variance.

$$m_t = \ \beta_1 * \ m_{t-1} + (1 - \beta_1) * \ \nabla_w J(w) \quad (2.6)$$

$$v_t = \ \beta_2 * \ v_{t-1} + (1 - \beta_2) * \ \nabla_w J(w)^2 \quad (2.7)$$

$$w_t = w_{t-1} - \ \frac{\alpha * m_t}{\sqrt{v_t} + \varepsilon} \quad (2.8)$$

With Adam, the learning rate of the optimizer is changing during training which is called learning rate decay. The decay value is utilized in the mean and uncentered variance with the symbol 'β'. With this feature, it became many favorite choices aside from traditional SGD, which maintains a single learning rate for all weight updates. The formula to update weight in Adam can be seen in **equation 2.8**.

2.5. Image Classification

Classification of the image in artificial intelligence has many techniques, with the most recent technique Convolutional Neural Network (CNN). The main idea of Image classification is that given a set of images with a label, we need to create a model that can predict the tested image and measure the accuracy of the prediction.

Convolutional neural networks have emerged as the master algorithm in computer vision in recent years and developing recipes for designing them has been a subject of considerable attention (Chollet, 2017).

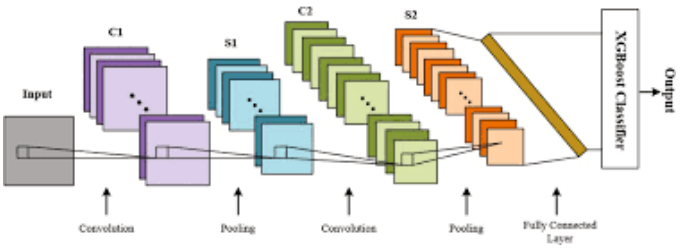

Figure 2.3 Example Model of CNN
(Ren et al., 2017)

There have been many models created based on the Convolution layer such as Convolutional Neural Network, VGG-16 (Very Deep Convolutional Networks for Large-Scale Image Recognition), Inception, ResNet, and EfficientNet.

Convolutional Neural Network consists of a combination of convolutional layer, pooling layer, fully connected layer, and several activation functions like ReLU (Rectified Linear Unit) and Softmax (Sony et al., 2021). One of the model examples that can be built with CNN can be seen in **Figure 2.3**.

2.5.1. Convolutional Layer

A convolutional layer is a layer that is mainly used for feature extraction. This layer performs an action called convolution. Convolutions perform multiplication between input data and a 2D array called kernel or filter. The intention of this process is to produce a collection of features with a smaller size compared to the input data called a feature map. The example of the convolution process can be seen in **Figure 2.4**. At the end of the process, the feature map content will be passed to the activation function ReLU (can be seen in **equation 2.9**) to reduce diversity (usually black/darker value). The ReLU function results can be seen in **Figure 2.5**.

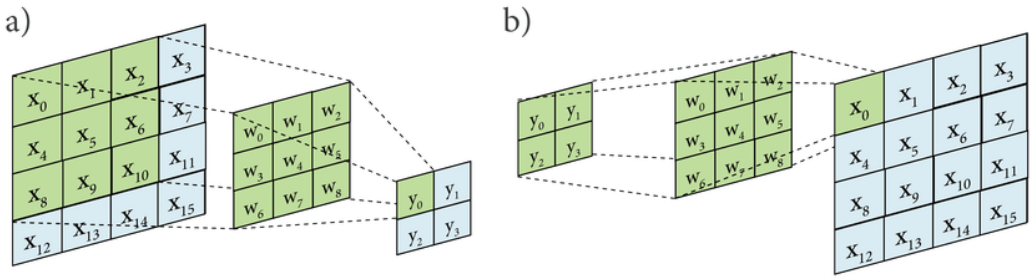

Figure 2.4 Convolution Example
(Mosser et al., 2018)

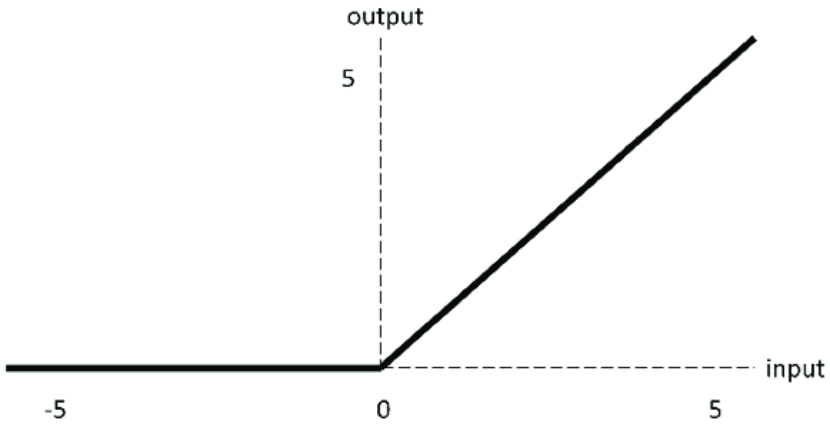

Figure 2.5 ReLU Activation Function
(Sultan et al., 2019)

$$f(x) = \max(0, x) \tag{2.9}$$

2.5.2. Pooling layer

The pooling layer usually appears to follow up after the convolution layer. The main purpose of the pooling layer is to reduce the computation cost and redundancy by filtering/reducing the size of the input. There are two common pooling operations used in the pooling layer which are Max Pooling and Average Pooling. Like its name, max pooling takes the maximum value in the selected patch in the feature map, while Average Pooling calculates the average of all values in the selected patch. The example of the pooling process can be seen in **Figure 2.6**.

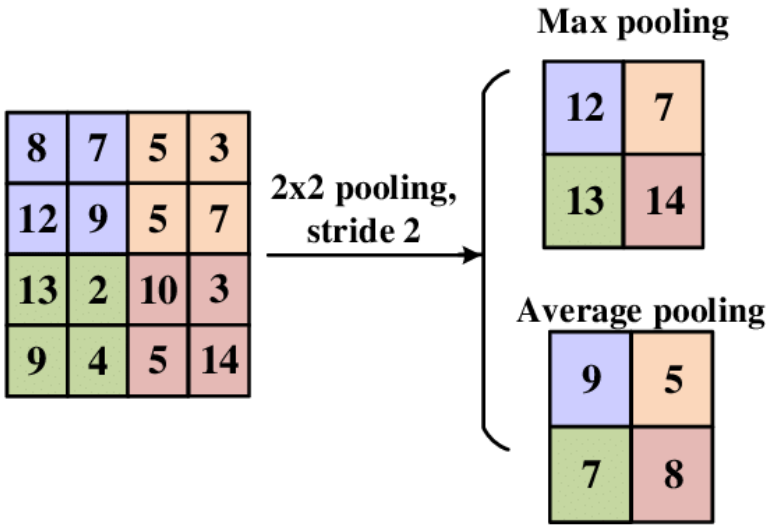

Figure 2.6 Pooling Layer Example
(Yingge et al., 2020)

2.5.3. Fully Connected Layer

A fully Connected Layer is used as an input in the process of classifying the value. The input in this layer needed to be a one-dimensional vector. This layer is a transitioning process between two-dimensional to one flattened dimensional input. The classifying process occurs at a dense layer with an activation function softmax (calculation value can be seen in **Figure 2.8**, which takes the features and calculate the probability of each possible class (Sony et al., 2021). Then, the layer produces the highest probability as the output. The process of classifying can be seen in **Figure 2.7**.

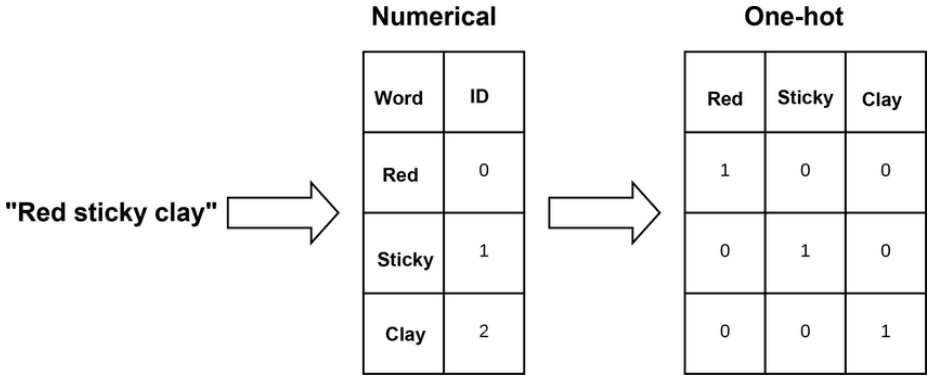

Figure 2.7 One Hot Encoding
(Padarian & Fuentes, 2019)

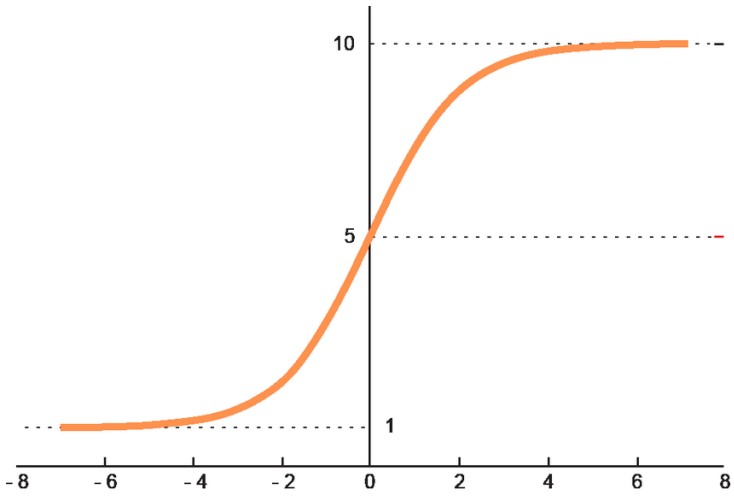

Figure 2.8 Softmax Activation Function
(Es-Sabery et al., 2021)

In (Wu & Chen, 2016), the paper proposes a deep learning method to recognize handwriting. The paper proposes 2 mainstream algorithms of deep learning, which are Convolutional Neural Network (CNN) and Deep Belief Network (DBN). The paper collects datasets from MNIST Database and the real world's handwritten character database. The paper achieved an accuracy of 99.28% and 98.12% from CNN and DBN for the MNIST database and accuracy of 92.91% and 91.66% for the real world's handwritten database.

In (Affonso et al., 2017), Affonso and his fellow researcher classify the quality of wood and compare the deep learning methods, such as Convolutional Neural Network (CNN) against traditional machine learning with Texture descriptor technique for feature extraction. The research concludes that texture descriptor with machine learning classification method is very competitive to Convolution Neural Network.

2.6. Text Classification

In NLP (Nature language processing), there are several ways to classify an emotion based on text. The technique goes from machine learning, Recurrent Neural Network Deep learning (Rius et al., 1998), Long Short-Term Memory (LSTM) (Hochreiter & Urgen Schmidhuber, 1997), Bidirectional Long Short-Term Memory (BLSTM) (Graves & Schmidhuber, 2005), and lastly the state-of-the-art Bidirectional Encoder Representations from Transformers (BERT**)** (Devlin, Chang, Lee, & Toutanova, 2018). BERT was proposed by researchers at Google Research in 2018. The main idea of text classification is that given a set of text and labels, we need to create a model that can predict the prepared validation's text and measure the accuracy of the prediction.

2.6.1. Feature Extraction

There are many techniques to extract features from the text, such as TF-IDF (term frequency-inverse document frequency), Bag of Words, Skip-Gram Model, GloVe (global vectors for word representation), FastText, and WordPiece.

2.6.1.1. TF-IDF

TF-IDF highlights a specific word that might not have many occurrences but have great importance in the documents. TF-IDF is a combination of two different words i.e. Term Frequency and Inverse Document Frequency (Qaiser & Ali, 2018). Term Frequency increase the importance of the words based on the counts the occurrences of the word in the documents and the Document Frequency decrease the importance based on the existence of the words across all the documents. An example of TF-IDF calculation can be seen in **Table *2.1***.

Table 2.1 TF-IDF Calculation Example

			TF			TF-IDF		
Term	**DF**	**IDF**	**d1**	**d2**	**d3**	**d1**	**d2**	**d3**
car	18,16	1.6	27	4	24	44.5	6.6	39.6
autol	6,72	2.0	3	33	0	6.2	68.6	0
insur.	19,24	1.6	0	33	29	0	53.4	46.9
best	25,23	1.5	14	0	17	21	0	25.5

2.6.1.2 Bag of Words

Bag of Words is a feature extraction technique that produces vector space that represents each document in the corpus. The key idea of the Bag of Words model is to quantize each extracted key point into one of the visual words, and then represent each image by a histogram of the visual words (Zhang et al., 2010). There are different vector formats depending on Bag of Words modifiers such as One Hot Encoding, TF-IDF, and Frequency Vectors. The most used is Frequency Vectors, in which the model will fill the vectors with the count of each word that appears in the document. An example of frequency vector counting can be seen in **Table 2.2**.

Table 2.2 Bag of Words Frequency Vectors Example
(Ye et al., 2016)

Article ID	biolog	biopsi	biolab	biotin	almost	cancer-surviv	cancer-stage	Article Class
00001	12	1	2	10	0	1	4	breast-cancer
00002	10	1	0	3	0	6	1	breast-cancer
00014	4	1	1	1	0	28	0	breast-cancer
00063	4	0	0	0	0	18	7	breast-cancer
00319	0	1	0	9	0	20	1	breast-cancer
00847	7	2	0	14	0	11	5	breast-cancer
03042	3	1	3	1	0	19	8	lung-cancer
05267	4	4	2	6	0	14	11	lung-cancer
05970	8	0	4	9	0	9	17	lung-cancer
30261	1	0	0	11	0	21	1	prostate-cancer
41191	9	0	5	14	0	11	1	prostate-cancer
52038	6	1	1	17	0	19	0	prostate-cancer
73851	1	1	8	17	0	17	3	prostate-cancer

doi:10.1371/journal.pone.0162721.t001

2.6.1.3. N – Gram

N-Gram models are used for a variety of things in NLP, such as sentence auto-complete, capturing hidden meaning in a sentence, and auto-check for grammar. N-Gram can capture context on the word, based on its before and afterword. bi-gram stands for 2-Gram which creates a 2-word sequence of a word, while Trigram stands for 3-Gram which creates 3-word sequence for every word in the sentences. An example of N-Gram can be seen in **Figure *2.9*.**

Table 1: Illustration of some texts and their sentiments.

ID	Sentiment	Text
$Text_1$	negative	This film is **not good**.
$Text_2$	positive	This film is **good**.
$Text_3$	negative	This film is **bad**.
$Text_4$	positive	This film is **good**, I give **7/10** to it.
$Text_5$	positive	Patrick Swayze's acting is **perfect**.

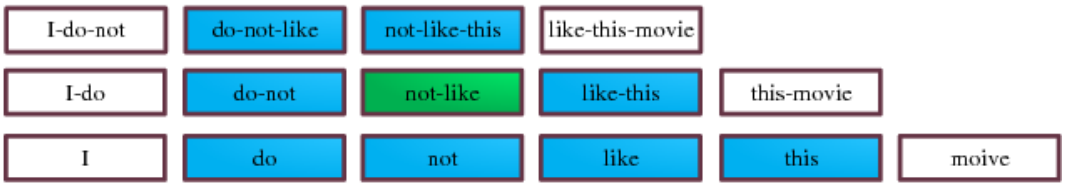

Figure 2.9 N-Gram Variety
(Li et al., 2017)

2.6.1.4. GloVe

The gloVe is an unsupervised learning algorithm for obtaining vector representations for words. Training is performed on aggregated global word-word co-occurrence statistics from a corpus, and the resulting representations showcase interesting linear substructures of the word vector space (Pennington et al., 2014). GloVe uses Euclidean Distance between the two words to measure the semantics similarities between the corresponding words. The end result of the calculated probability can be represented in **Figure *2.10*.**

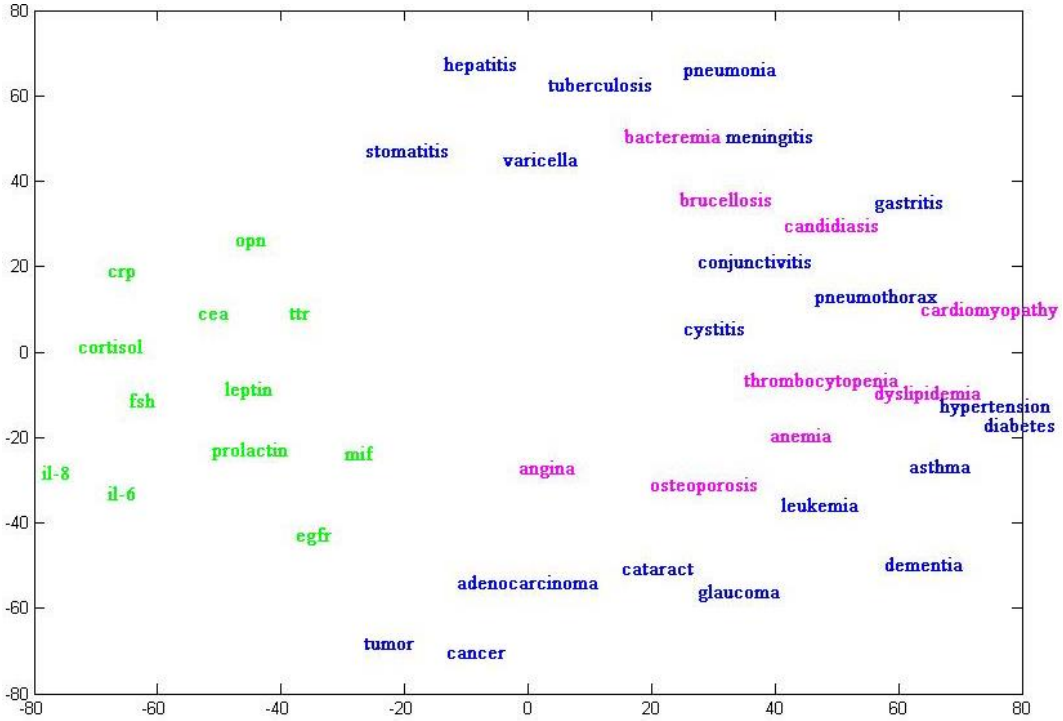

Figure 2.10 Word Relationship in GloVe
(Youn et al., 2016)

2.6.1.5. FastText

FastText embeddings exploit subword information to construct word embeddings. Representations are learned of character -grams, and words are represented as the sum of the -gram vectors (Bojanowski et al., 2017). The main idea of FastText is that FastText split the words into subwords and apply an N-gram algorithm to understand the complexity of the subwords.

2.6.1.6. WordPiece

WordPiece is a subword segmentation algorithm used in natural language processing. The vocabulary is initialized with individual characters in the language, then the most frequent combinations of symbols in the vocabulary are iteratively added to the vocabulary model is a word embedding model (Schuster & Nakajima, 2012).

WordPiece is a similar version of FastText because both of them split the words into subwords. WordPiece does not split every character but only split the unknown word in WordPiece's vocabulary into several parts, with every character or combination of 2 characters already having its token id. Splitting examples of word can be seen in **Figure 2.11**.

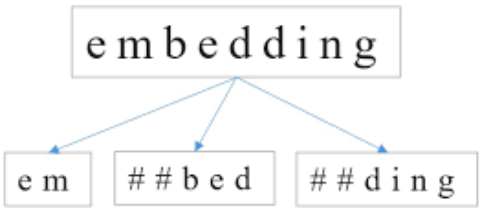

Figure 2.11 Word Piece Split Result
(Ly et al., 2020)

2.6.2. BERT

Most of the text classification processes implement the NLP algorithm. BERT, a state-of-the-art NLP technique, uses a bidirectional transformer encoder to pre-train the language model. BERT utilizes WordPiece for tokenization and uses the transformer's encoders to decide the value of every token which can be used for word embedding.

BERT language model implements the 'Masked LM' and Next Sentence Prediction (NSP) technique to train the new token that is inserted into the model. Masked LM technique randomly replaces 15% word with [MASK] token and tries to predict the value of the mask token with softmax function for each word in the vocabulary. The next Sentence Prediction technique trains the language model with 50% correct next sentence and 50% random next sentence to distinguish the correct next sentences. Every sentence is inserted with [CLS] token at the beginning of the

sentences and [SEP] token is inserted at the end of each sentence, which can be seen in **Figure 2.12**. Masked LM and Next Sentence purpose is to provide the best value of the token for each word in word embedding.

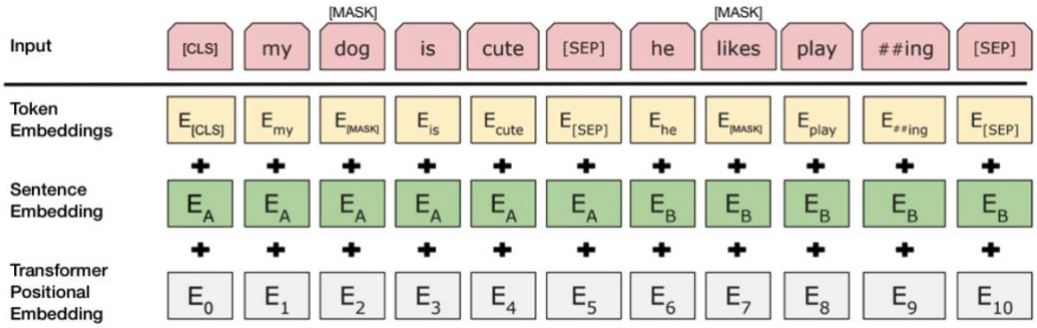

Figure 2.12 Example of NSP Input in BERT
(Paul & Saha, 2020)

There is various research that compares text classification algorithms and models. In (Kumar & Ojha, 2019), the research did the comparison of offensive or hate sentences using BERT and SVM. The research compares the result of the F1 score in three different language datasets, is English, Hindi, and Bangla text. This paper concludes that BERT generally does better than SVM in most cases.

In (González-Carvajal & Garrido-Merchán, 2020), Gonzales and Garrido compared BERT (Bidirectional Encoder Representations from Transformers) with NLP machine learning in 4 different NLP scenario from different sources. This paper proves that BERT outperformed classical NLP models like Linear SVC, Ridge Classifier, etc. The researcher also wrote that implementing BERT is far less complicated than traditional methods.

In (Baruah et al., 2020), the research used English-BERT, RoBERTa, DistilBERT, and SVM classifier for English language and multilingual BERT (M-BERT), XLM-RoBERTa, and SVM classifiers for Hindi and Bangla text. The

research purpose is to compare BERT and SVM classifiers. The output of the experiment is a weighted F1 score. The conclusion of the research is the superior performance of the SVM classifier was achieved mainly because of its better prediction of the majority class. BERT based classifiers were found to predict the minority classes better.

2.7. Multimodal feature fusion

many phases, which are in data level fusion, feature level fusion, or decision level fusion. In (Mondal & Kaur, 2016) and (Gunatilaka & Baertlein, 2001), it is proven that feature level fusion is more efficient than decision level fusion. In feature fusion, the different modality's features are extracted separately. With the help of the dense layer, the feature can achieve the same dimension and with the concatenate layer, the feature can combine into a one-dimension vector (Bae, Park, Lee, Lee, & Lim, 2020). Then, the combined features will pass through the dense softmax layer for the classification. **Figure 2.13** show the example of feature level fusion.

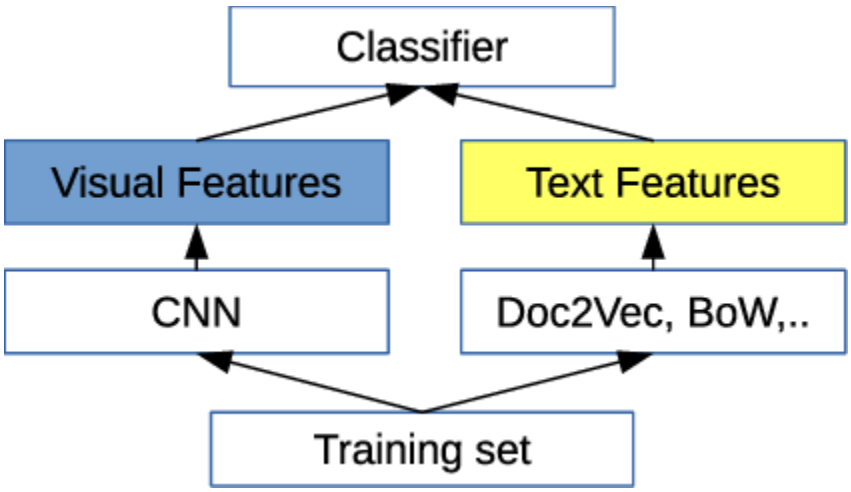

Figure 2.13 Example of Multimodal Feature Level Fusion Model Structure
(Gallo et al., 2017)

There is various research that compares the Unimodal and Multimodal features. In (Thang Duong et al., 2017), the research proposed a simple model that combines multiple modalities to analyze social media. The research purpose is to classify with better accuracy for multimodal. The paper experiments use 2 fusions, which are common space fusion and joint fusion. The proposed fusion is robust in that it can achieve high accuracy while missing some modalities.

In (Chaparro et al., 2018), Chaparro and his fellow researcher implement their first approach on multimodal classification fusion of both Electroencephalography and facial micro-expressions. The accuracy result obtained improved by 12% per emotion compared to unimodal.

In (Huang et al., 2017), Yongrui and his team proposed 2 modal fusions of the brain (EEG) and peripheral signal and facial recognition. In their research, the multimodal fusion method outperforms the unimodal of facial and unimodal EEG.

In (Cambria et al., 2018), the paper compares the use of unimodal sentiment analysis, bimodal sentiment analysis, and trimodal sentiment analysis. They use 3 modalities such as visual, text, and audio. They combine on feature level fusion which textual feature by Convolutional Neural Network (CNN), audio feature extracted by openSMILE, and the visual obtained from the video which features extracted by CNN. The paper arrived in conclusion that multimodal outperforms the state of the art of both tasks.

In (Huang et al., 2019), the paper proposed a novel multimodal of text and image model, called A Deep Multimodal Attentive Fusion (DMAF). The research's first step is to separate the unimodal attention model and train them to classify the

emotion, respectively. Then, they exploit the correlation between the two unimodal models to an intermediate fusion model. Then, they do a final late fusion to the output of the three models to arrive at the final decision. The paper concludes that the new model outperforms the state-of-the-art baselines which are SVM, STM, Early Fusion, Late Fusion, etc. on four different real-time datasets.

In (Zheng et al., 2021), the paper proposed three models, which is speech emotion recognition, motion emotion recognition, and text emotion recognition. SER model use LLD features to highlight the local and global information. The focus on the audio modality can improve the accuracy by 9% from the state of the art. This paper also combined the models into a multimodal model that achieve 75.1% of accuracy.

In (Guggenmos et al., 2020), the paper applies multimodal technique to a model with purpose of detecting alcohol dependence classification. This research proposed a model that takes several neuroimages as input and with an accuracy of 79.3%, it outperforms the strongest individual modality by 2.7%.

The summarize of the recent works in proposed unimodal and multimodal research can be seen in **Table 2.3**.

Table 2.3 Recent Work

No	Publication	Problem	Solution	Result
1	(Affonso et al., 2017)	Wood classification	Convolution Neural Network and Texture descriptor method	Accuracy - CNN: 77.69% - MLP: 81.26% - KNN: 82.11%

No	Publication	Problem	Solution	Result
				- SVM: 80.73% - DT: 81.28%
2	(Guggenmos et al., 2020)	Diagnostic classification for alcohol dependence	Multimodal classification of neuroimages	- 79,3% accuracy of multimodal - 76.6% of unimodal
3	(Thang Duong et al., 2017)	Emotion classification	Multimodal approach of Image and Text	Reddit Dataset Accuracy: - Image: 77.48% - Text: 78.4% - Early fusion: 84.11% - Late fusion: 83.33%
4	(Chaparro et al., 2018)	Emotion classification	Multimodal approach of Electroencephalography and facial expression	MANHOB-HCI database - EEG: 93.9% - Face: 91.3% - Multimodal: 97.7%

No	Publication	Problem	Solution	Result
5	(González-Carvajal & Garrido-Merchán, 2020)	Text and sentiment analysis	BERT (Bidirectional Encoder Representations from Transformers) and the best classifier before BERT	IMDB Dataset - BERT: 0.9387 - Linear SVC: 0.8989 - Logistic Regression: 0.8949 - Multinomial-NB: 0.8771 - Voting Classifier: 0.9007 - Ridge Classifier: 0.8990 - Passive-Aggressive-Classifier: 0.8931
6	(Cambria et al., 2018)	Sentiment analysis and emotion recognition	Unimodal and Multimodal of Text, Audio, and Image feature	MOSI dataset - Text + Audio + Video: 71.59%

No	Publication	Problem	Solution	Result
				- Text + Audio: 70.79% - Text + Video: 68.55% - Audio + Video: 52.15% - Audio: 51.52% - Video: 41.79% - Text: 65.13%
7	(Baruah et al., 2020)	Aggression Identification from 3 different language (English, Hindi, and Bangla)	BERT (Bidirectional Encoder Representations from Transformers), RoBERTa, and SVM (Support Vector Machine)	Accuracy English: - En-BERT: 0.9467 - SVM: 0.9390 - DistilRoBERTa: 0.9289 Hindi: - XLM-RoBERTa: 0.7207 - SVM: 0.7192 - M-BERT: 0.6871

No	Publication	Problem	Solution	Result
				Bangla: - XLM-RoBERTa: 0.9039 - SVM: 0.8851 - M-BERT: 0.8966
8	(Kumar & Ojha, 2019)	Hate Speech and Offensive Content Detection in Indo-European Language	BERT (Bidirectional Encoder Representations from Transformers) and SVM (Support Vector Machine)	BERT perform high recall yet low precision even with a very little training sample. In this aspect, BERT outperform SVM. BERT also able to generalize better for the given task than SVM, even in the case of unbalanced dataset
9	(Zheng et al., 2021)	Multimodal classification with SER, FER, and MER	Create an improvised model of SER, FER, MER, and combine all of them to improve accuracy	- Multimodal accuracy of 75.1%

No	Publication	Problem	Solution	Result
10	(Huang et al., 2019)	Sentiment Analysis	Multimodal model DMAF (Deep Multimodal Attentive Fusion) from image and text modalities	DMAF model outperform state of the art baseline (T-LSTM-E, CCR, CCR+V, and TFN)

CHAPTER III

METHODOLOGY

3.1. Research Methodology

Emotion classification is an important aspect that can be used in many fields. This technology can be implemented in many fields of work e.g., detects user satisfaction about a product, detecting user's emotion in social media, and many more. The research of emotion classification has been around for a long time. There are many existing models to detect emotions. This research purpose is to produce a better model than some existing models.

The proposed model uses deep learning neural network with a combination of feature level extraction from the different dataset because the proposed method has a possibility to outperform the performance of unimodality text or images. The research outline can be seen in **Figure 3.1**.

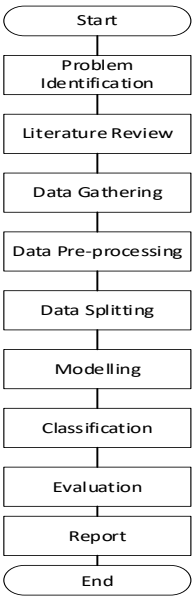

Figure 3.1 Research Methodology

3.2. Dataset Gathering and Preprocessing

Interactive Emotional Dyadic Motion Capture (IEMOCAP) (Busso et al., 2008) fulfill the requirement as the dataset for our research. IEMOCAP contains 12 hours of audio-visual data, consisting of video, speech, motion capture of face and hands, and text transcriptions with a categorical label such as angry, happiness, sadness, and neutrals that are annotated by multiple annotators split into 5 sessions. Data used in this research are the speech audio file, motion capture features (head, hand, and rotated), and text transcription of speech.

IEMOCAP dataset consists of 5 Sessions which every session represents improvised scenarios. The recording session is then divided into utterances that are annotated by multiple annotators. The utterances are evaluated based on 10 possible emotions (angry, happy, sad, neutral, frustrated, excited, fearful, surprised, disgusted, and other), valence (positive or negatives), activation (calm or excited), and dominance (passive or aggressive). The available data contained in each utterance are the timing of the utterance, the wave file, motion gestures (head, hand, and rotated), and the words were spoken per utterance.

Table 3.1 Emotion data information

Emotion	Count	Duration in seconds
Happy	1,627	7,560
Sad	1,084	5,958
Neutral	1,708	6,664
Angry	1,102	4,977
Total	5,521	25,160

This research first processes the available information then reconfigure it into a table with column start time, end time, wave file name, and annotated emotions. After that, the data are further filtered by specific emotions (angry, happy, sad, neutral, and excited) in consideration of the data's balance following the method proposed by (Mittal et al., 2020). Then, the data with excited emotion are replaced into happy to further increase the data's balance. The result of preprocessing is 5,521 total utterances with 1,102 angry, 1,627 happy, 1,084 sad, and 1,708 neutral utterances which can be seen in **Table *3.1***. For the audio part, this research enforces the length of every audio utterance into 17 seconds, which is the average length of all audios. Audio with less than 17 seconds is being filled with silence tone in the end and audio with more than 17 seconds is being taken only 17 seconds from the start of the audio.

This research also did data augmentation to further normalize the data as a possible solution to reduce the overfitting. With the success increase in accuracy proven by (Salamon & Bello, 2016)and (Nanni et al., 2019), the augmentation done to the audio file are following the standard application of the librosa library at python. The implemented augmentations are as such:

- addition of Gaussian noise of a range of 0.001 – 0.015 amplitude
- time-stretch between 0.8 to 1.2 multiplier
- shift of Pitch from -4 semitones to 4 semitones
- shift of fraction between -0.5 to 0.5.

3.3. Feature Extraction

After preprocessing, this research iterates for every utterance to collect features prepared in different directories. In the case of motion capture, the head, hand, and rotated, all have different columns' sizes, where the head has 6 features, the hand has 18 features, and the rotated facial expressions contain 165 features. This research handles the motion capture features following the approach used in (Tripathi et al., 2018). In every utterance, there are many different frames with an average interval of 0.008 seconds. To simplify the features and lessen the memory required, this research divides all the available frames by 200 blocks, then average the features in every block e.g., the dimension of the head feature is (5521, 200, 6) where the 5,521 are the count of the row, 200 are the block, and 6 are the dimension length of head's feature. The example of coordinates collected from the facial expression can be seen in ***Figure* 3.2**.

Figure 3.2 IEMOCAP Head Movement and Head Angle Feature
(Busso et al., 2008)

In the case of the text features, this research utilizes token embeddings and mask embeddings with a maximum length of 128 from the words of every utterance. Token embeddings and mask embeddings are the input required for the BERT

model's unique way to process data. In the BERT model, token embeddings are the actual value of the word and mask embeddings serve as the indicator of whether the token is an actual token or padded token. Although the word is the same, the value of the token embeddings can be different following the pre-trained BERT model's dataset. The output dimensions of the tokenizer are (5,521, 128) for the token embeddings and mask embeddings. The example of tokenized text can be seen in ***Table* 3.2**.

Table 3.2 BERT-base Tokenized Example

Words	Token (max length 18)	Mask (max length 18)
It'll be good. Wow, that's great.	[101 1,135 112 1,325 1,129 1,363 119 11,750 117 1,115 112 188 1,632 119 102 0 0 0]	[1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0]
I knew they would, your mother anyway.	[101 146 1,450 1,152 1,156 117 1,240 1,534 4,050 119 102 0 0 0 0 0 0 0 0]	[1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0]

In the case of speech features, this research utilizes Mel-spectrograms, a Spectrograms with the Mel Scale serves as y-axis and time as x-axis for the features. This research extracts the Mel-spectrograms feature with librosa library from every audio. The examples of Mel-spectrograms feature can be seen in ***Figure* 3.3**.

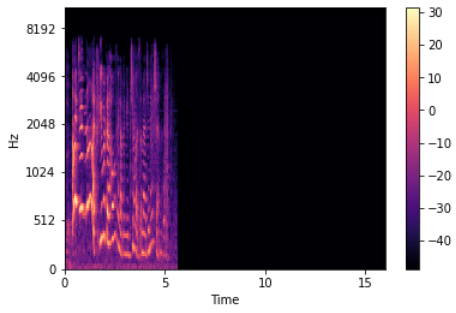

Figure 3.3 Padded Mel-spectrogram's Feature

3.4. Data Splitting

The dataset used by the model is separated into data splitting sets. This research follows the traditional method of splitting the dataset into train and test purposes. The first step of data splitting is randomizing the dataset to create more evenly distributed data. After that, we split the train and test by an 8:2 ratio, 8 for the train and 2 for the validation to avoid overfitting or underfitting. The 8:2 ratio is applied to all proposed models.

3.5. Modelling Unimodal Model

In this research, unimodal models are used to train every modality. Every model in this research consists of feature extraction layers then followed by a 4-unit dense layer with SoftMax activation function that acts as a classifier layer. All the model applies the same parameters batch size of 128, 50 training epoch, and usage of optimizer Adam. The modality used in this research are motion capture, text, and speech with several models for each modality is introduced.

3.5.1. Motion Capture

The motion captures modality uses the combined feature of the head, hand, and rotated facials feature. This research uses a simple LSTM model, Dense model, CNN model, with the addition of a simple bi-LSTM model as possible improvement as feature extractor, which is utilized in (Tripathi & Beigi, 2018). LSTM model consists of an LSTM layer with 256 units followed by a 128-unit Dense layer with activation function Rectifier Linear Unit (ReLU) which can be seen in ***Figure* 3.4**. The Dense model consists of a Dense layer with 256 units followed by a 128-unit Dense layer with ReLU activation, which can be seen in ***Figure* 3.5**. The CNN model consists of 3 Convolutional Layers with kernel size 3, Stride 2, and the filter

32, 64, and 64 respectively. Every Convolutional layer is followed with ReLU activation and a dropout of 0.2. The output of the last convolutional layer is followed by a 128-unit Dense layer with ReLU activation. The model layers in CNN can be seen in ***Figure* 3.6**. The bi-LSTM model consists of a 256-unit of LSTM layer with bidirectional applied followed by a 128-unit Dense layer with ReLU activation, which can be seen in ***Figure* 3.7**. All the model's input layers are configured following the dimension of the input e.g., the input in the LSTM layer for the hand is (200, 18), the head is (200, 6), the rotated facials is (200, 165), and the combined is (200, 189). A total of 4 model are created for the purpose of evaluating motion capture head, hand, rotated, and combined which can be seen in **Table *3.3***.

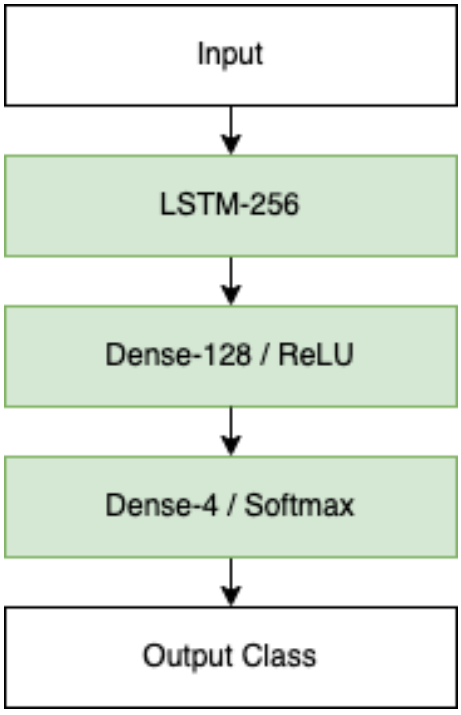

Figure 3.4 LSTM model

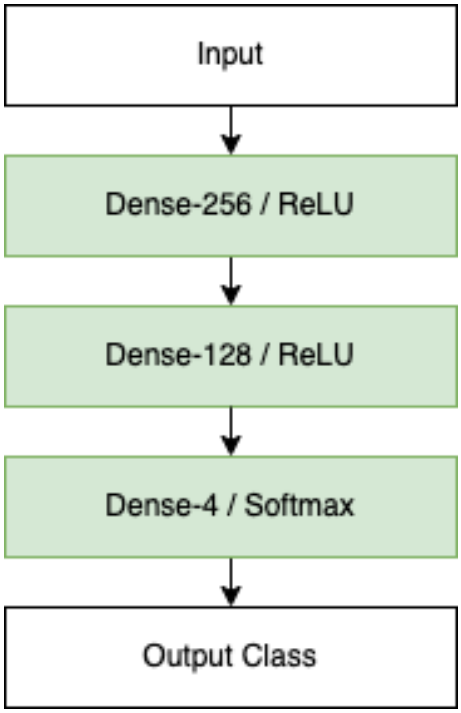

Figure 3.5 Dense model

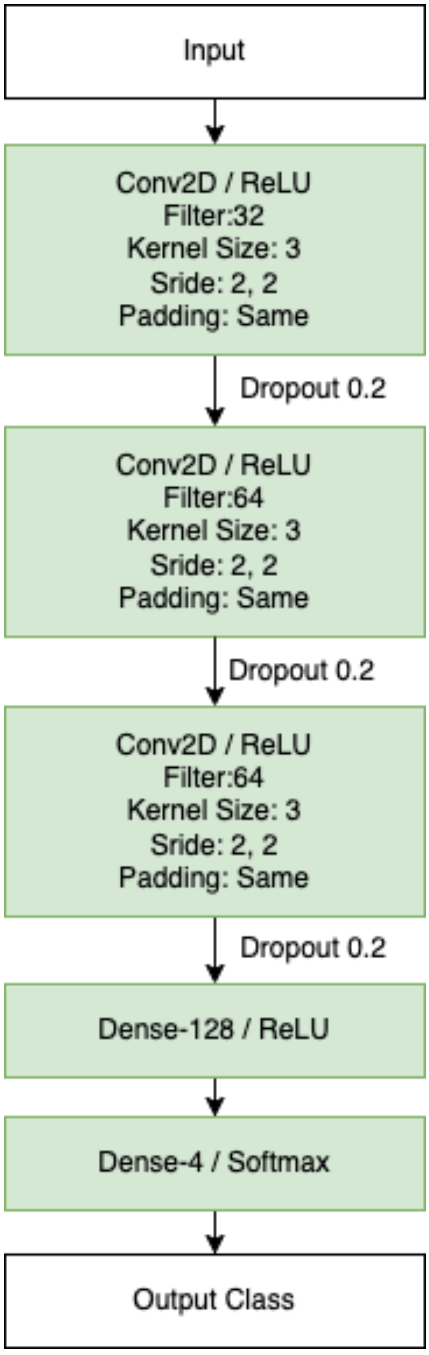

Figure 3.6 CNN model

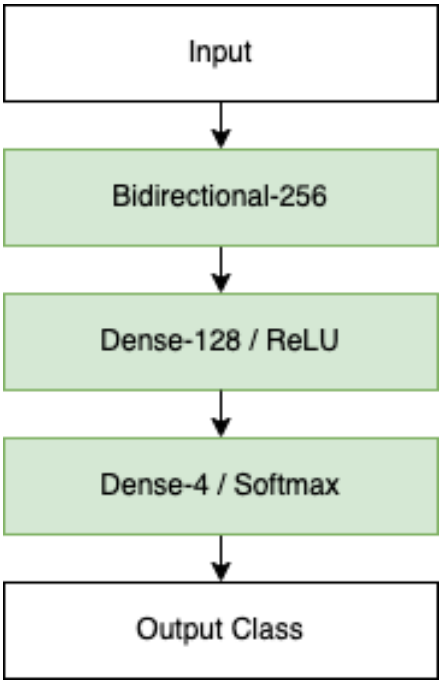

Figure 3.7 Bidirectional LSTM model

Table 3.3 Motion Capture unimodal models

No	Model Name	Motion Capture Segments	Feature Count	Total Parameter
1	CNN	Head	6	261,828
2	Dense	Head	6	340,868
3	LSTM	Head	6	302,724
4	bi-LSTM	Head	6	604,804
5	CNN	Hand	18	671,684
6	Dense	Hand	18	955,268
7	LSTM	Hand	18	315,012
8	bi-LSTM	Hand	18	629,380
9	CNN	Rotated	165	4,357,828

No	Model Name	Motion Capture Segments	Feature Count	Total Parameter
10	Dense	Rotated	165	8,481,668
11	LSTM	Rotated	165	465,540
12	bi-LSTM	Rotated	165	930,436
13	CNN	Combined	189	4,972,228
14	Dense	Combined	189	9,710,468
15	LSTM	Combined	189	490,116
16	bi-LSTM	Combined	189	979,588

3.5.2. Text

For the text modality, this paper proposes the utilization of 3 different BERT models, which are BERT, roBERTa, and DistillBERT which are referenced from (Devlin, Chang, Lee, Google, et al., 2018), (Liu et al., 2019), and (Sanh et al., 2019) respectively. All BERT models start with an input layer, which accepts token embeddings and mask embeddings. After the input layer, the next layer consists of pre-trained transformers model from each BERT version, with the trainable of every layer disabled. The transformers model output was extracted with Tensorflow Slicing Lambda which was then applied to a Batch Normalization layer and followed by a Dense layer with 128 units. The overall model's structure can be seen in **Figure 3.8**. The text unimodal model used in this research can be seen in ***Table* 3.4**.

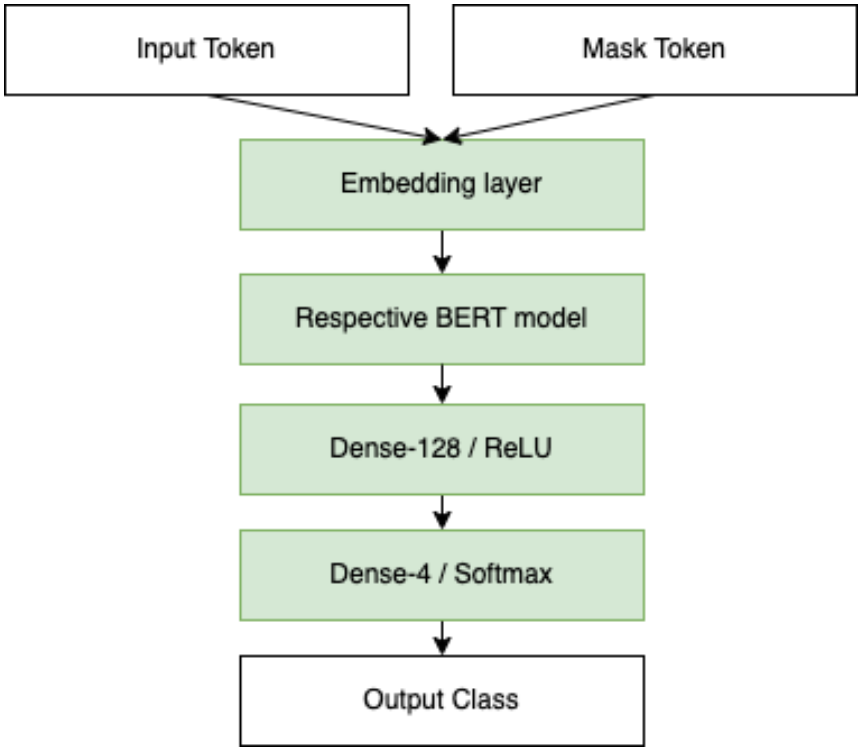

Figure 3.8 BERT-based Plotted Model

Table 3.4 Text unimodal models

No	Model Name	Total Parameter
1	BERT base	108,412,292
2	RoBERTa	124,747,652
3	DistilBERT	65,292,932

3.5.3. Speech

The speech modality in this research proposes 3 models which are a simple bi-LSTM model, CNN model, and mixed CNN and bi-LSTM model. The bi-LSTM model consists of a 256-unit of LSTM with bidirectional layer applied, followed by 128-unit Dense layer with activation function Rectifier Linear Unit (ReLU), which can be seen in ***Figure* 3.7**. The CNN model consists of 5 Convolutional Layers with

kernel size 3, Stride 2, and the filter 32, 64, and 64 respectively. Every Convolutional layer is followed with ReLU activation and a dropout of 0.2 and the output of the last convolutional layer is followed by a 128-unit Dense layer with ReLU activation, which can be seen in ***Figure* 3.6**. The mixed CNN bi-LSTM model is a model that consists of 3 convolutional layers, followed by a 512-unit bi-LSTM layer. The output of the bi-LSTM layer is then followed by a Dense layer with 128 units. Overall, the structure of CNN-bi-LSTM model can be seen in ***Figure* 3.9**. The models used for unimodal model audio classification can be seen in ***Table* 3.5**.

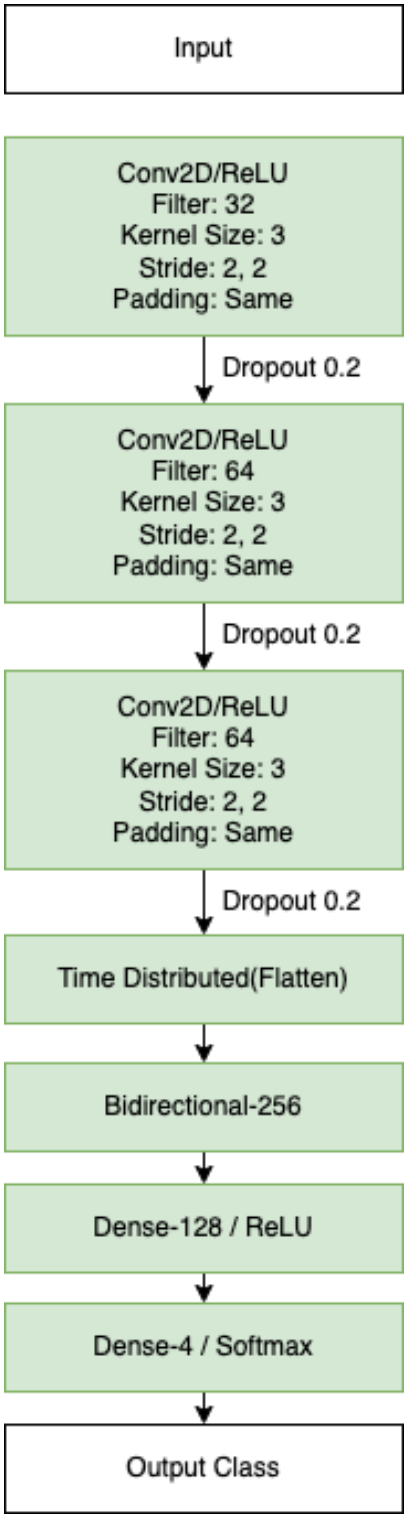

Figure 3.9 CNN-biLSTM model

Table 3.5 Audio unimodal models

No	Model Name	Total Parameter
1	CNN	11,460,292
2	Bi-LSTM	2,005,636
3	CNN-Bi-LSTM	12,052,164

3.5. Modelling Multimodal Models

The multimodal models are built from the best performance model from each modality. The multimodal model's early layer consists of the same layer as the used unimodal model until its feature extraction layer. The multimodal model applies early fusion which fuse the output of every unimodal feature extraction layers as the feature and the output features from each unimodal are then concatenated and counted as multimodal model's feature. **Figure 3.10** is an example of a multimodal model combined from text, audio, and motion capture. There are 5 multimodal models that are used in this research, 4 which are the combinations of top performing unimodal model and 1 baseline model from (Tripathi et al., 2018), which can be seen in ***Table* 3.6**.

Table 3.6 Multimodal models

No	Model Name
1	Text + Motion Capture
2	Text + Audio
3	Motion Capture + Audio
4	Text + Motion Capture + Audio

No	Model Name
5	(Tripathi et al., 2018) Text + Motion Capture + Audio

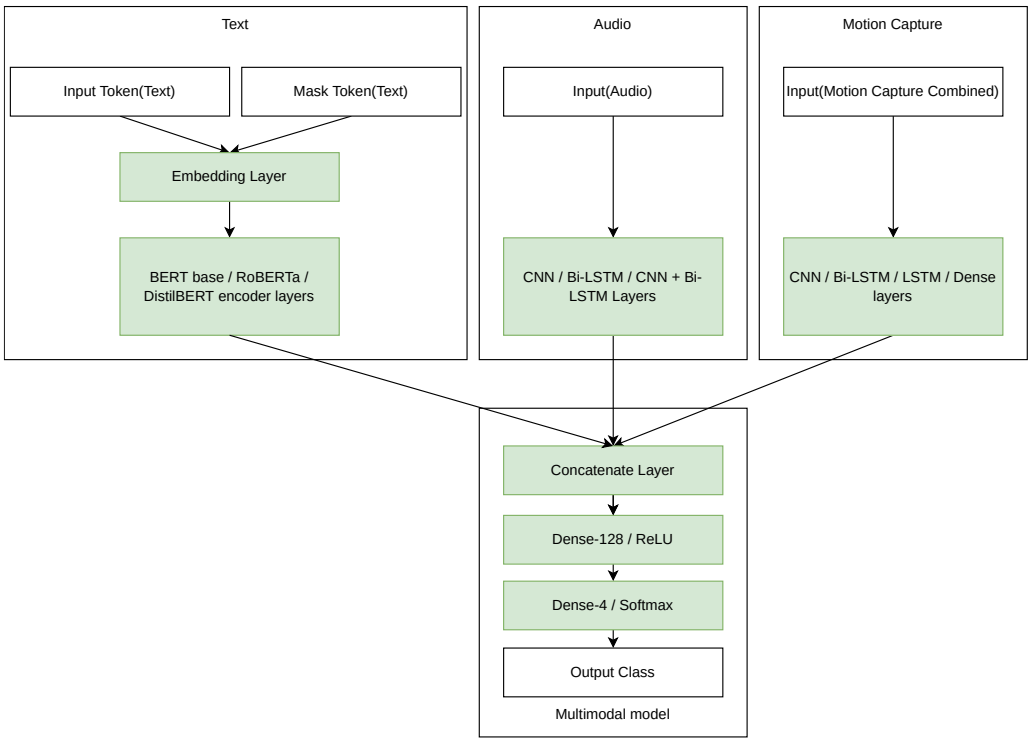

Figure 3.10 Combination of Text BERT and Motion Capture CNN

3.6. Classification and Evaluation

At the classification step, we train the input data to improve the accuracy and reduce the loss by trial and error. The evaluation metrics used in this research are accuracy, loss, validation accuracy, validation loss, and F1 score. The accuracy of the model itself is taken from all true predicted class, divided by additions of true predicted and false predicted. To evaluate the F1 score, we will need the sum of 4

different values as a parameter to calculate every performance metrics which are True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). The value of every metrics is dependent to the value of each prediction in confusion matrix, which can be seen in **Table 3.7**.

Table 3.7 Confusion Matrix

		Predicted			
		Angry	Happy	Sad	Neutral
Actual	Angry	**95.88%**	1.19%	0.48%	2.46%
	Happy	1.76%	**94.49%**	0.91%	2.85%
	Sad	1.22%	3.50%	**83.37%**	11.91%
	Neutral	5.62%	6.71%	6.39%	**81.28%**

- True Positive, indicates that the predicted value is same as actual value. E.g., a predicted value of angry image is angry
- False Negative, a value that implies the predicted value is incorrect. E.g., on angry class viewpoint, an actual angry image is being predicted as happy, sad, or neutral image is counted as false negative
- False Positive, a value that implied the predicted value is correct, but the actual value is incorrect. E.g., on angry class viewpoint, a predicted value of an angry image is actually happy, sad, or neutral image
- True Negative, a value that implied the predicted value and the actual value has no relationship with the following classes. E.g., from angry class viewpoint, a predicted of a not angry image for the actual not angry image is counted as True Negative

Then, the classification report of every model after the prediction is collected, which consists of performance metrics like overall precision, recall, and F1 score for every class which can be seen in **Figure 3.11** for example. The final f1 score is obtained by average the f1 score of every class.

	precision	recall	f1-score	support
Happy	0.44	0.60	0.51	315
Angry	0.66	0.40	0.50	233
Sad	0.70	0.57	0.63	216
Neutral	0.53	0.54	0.53	341
accuracy			0.53	1105
macro avg	0.58	0.53	0.54	1105
weighted avg	0.56	0.53	0.54	1105

accuracy: 53.48%

Figure 3.11 Classification report

Precision in multi-class classification is obtained by true positive divided by the addition of true positive and false positive of the specific class which can be seen in **equation 3.1**. **Recall** in multi-class classification is obtained by calculating true positive of the specific class, divided by addition of true positive and false negative of the specific class which can be seen in **equation 3.2**. **F1 score** make use of both precision and recall in its formula, which can be seen in **equation 3.3**.

$$Precision(class = a) = \frac{TP(class = a)}{TP(class = a) + FP(class = a)} \quad (3.1)$$

$$Recall(class = a) = \frac{TP(class = a)}{TP(class = a) + FN(class = a)} \quad (3.2)$$

$$F-1\ Score(class=a)=\frac{2x\ Precision(class=a)*Recall(class=a)}{Precision(class=a)+Recall(class=a)} \quad (3.3)$$

Based on the classification report, training, and prediction results of unimodal model, the top performing unimodal model from all modalities are used as a base to create multimodal models. The multimodal models will further be evaluated with the same metrics used to evaluate unimodal model. Based on the collected metrics, this research analyses from the approach of top performing model, least performing model, and offer some objective points for every modality and multimodal result.

CHAPTER IV
RESULTS & DISCUSSIONS

4.1. Training Environment

All models in this research have the same hyperparameter applied to ensure the fairness of the result for comparison purposes. This research was carried out in the Google Colab environment, runtime environment of GPU Tesla K80 with a specification of 12.69 GB RAM and 78.2 GB of Disk, The hyperparameter used in this research are an Epoch of 50, Batch Size of 32 because of memory limitation, and a starting learning rate of 0.001 which is the default value of Adam optimizer.

4.2. Results

The test that occurred in this research is split into two segments. The first segment consists of training and testing the performance of every unimodal model. Then, it is followed by training and testing the performance of the multimodal models which are the combination of every best F1 score performing unimodal.

4.2.1. Unimodal model

There are 3 different modalities in this research, with each of them having several models. The facial motion capture modality uses 4 models (LSTM, Dense, bi-LSTM, and CNN), The text modality uses 3 BERT based models (RoBERTa, DistillBERT, and BERT base), and lastly, the audio uses 3 models (bi-LSTM, CNN, and a mixed of bi-LSTM and CNN model).

Table 4.1 Motion Capture Hand Result

Mocap Hand Model	Training Accuracy	Training Loss	Validation Accuracy	Validation Loss	F1 Score
LSTM	0.6387	0.8668	0.4862	1.3361	0.465
Dense	0.3366	1.2982	0.0.3179	2.7523	0.1675
bi-LSTM	0.6673	0.7858	0.5192	1.2232	0.5
CNN	0.7358	0.6636	0.5509	1.2279	0.5775

Based on **Table 4.1** with 18 columns of data, it can be seen that the CNN model has the most accuracy and validation accuracy, the smallest loss, a small difference in validation loss with the smallest validation loss, and the highest F1 score. The second-best model can be seen in bi-LSTM which has the smallest Validation loss and an F1 score of 0.5. The dense model turns out to be the worst model with the smallest accuracy and F1 score and biggest loss.

Table 4.2 Motion Capture Head Result

Mocap Hand Model	Training Accuracy	Training Loss	Validation Accuracy	Validation Loss	F1 Score
LSTM	0.9824	0.0554	0.3688	3.644	0.365
Dense	0.718	0.9968	0.3688	5.0209	0.3325
Bi-LSTM	0.984	0.0369	0.3688	3.6755	0.38
CNN	0.9196	0.2172	0.3552	2.8926	0.4

In **Table 4.2** with 6 columns of data, the Bi-LSTM model outperforms the 3 other models in 3 metrics, which are accuracy, loss, and validation accuracy. The second-best performing model is CNN, which has the smallest validation loss and an F1 score of 0.4. The dense model also showed poor performance in the motion capture head with the smallest accuracy, smallest F1 score, and biggest loss.

Table 4.3 Motion Capture Rotated Result

Mocap Hand Model	Training Accuracy	Training Loss	Validation Accuracy	Validation Loss	F1 Score
LSTM	0.44	1.2296	0.44	1.2173	0.25
Dense	0.3024	1.3672	0.3201	1.3622	0.1225
Bi-LSTM	0.4151	1.237	0.4412	1.2232	0.275
CNN	0.7381	0.6369	0.5452	1.2138	0.5475

In **Table *4.3*** with 165 columns of data, the CNN model outperforms all other models in all fields. With the differences of 0.2975 in F1 score with the second-best model, closely winning the smallest validation loss, a 0.14 increase of validation accuracy with the second-best performing model, the smallest loss and highest accuracy. The Dense model is also the worst performing model in motion capture rotated, although the difference in performance is smaller than the other models.

Table 4.4 Motion Capture Combined Result

Mocap Hand Model	Training Accuracy	Training Loss	Validation Accuracy	Validation Loss	F1 Score
LSTM	0.5286	1.0982	0.4876	1.1387	0.495
Dense	0.5798	6.486	0.5317	7.0407	0.4975
Bi-LSTM	0.5317	1.1055	0.5192	1.1293	0.48
CNN	0.8414	0.4124	0.655	1.1344	0.6625

With 165 columns of data, it can be seen in **Table *4.4*** that the CNN model shows that it is the best model for the motion capture combined with the F1 score of 0.6625, with an increase of 0.165 in the F1 score. The validation loss result is close to the smallest validation loss of the Bi-LSTM model and has the highest accuracy and validation accuracy, and the smallest loss. The dense model shows better accuracy than LSTM and bi-LSTM model, even though the loss value is still

much bigger, there is a progressive improvement for the Dense model compared to previous motion capture data.

From the discussion result of **Table *4.1***, **Table *4.2***, **Table *4.3***, and **Table *4.4***, the best model is the CNN model that achieves the best performance in 3 motion capture, which is followed by the Bi-LSTM model. The advantage of the CNN model in the past results shows that the more columns data used as input, the better the performance of the CNN model. The Dense model starts as the worst model but shows the improvement that can rival other models with enough columns of data in terms of accuracy.

Table 4.5 Text Model Result

Text Model	Training Accuracy	Training Loss	Validation Accuracy	Validation Loss	F1 Score
roBERTa	0.6318	0.8796	0.6452	0.8796	0.6475
DistillBERT	0.8675	0.3662	0.6606	1.0519	0.66
BERT base	0.8786	0.3256	0.6715	1.0945	0.675

Table *4.5* shows the training and testing result for the 3 text models. BERT base outperforms other models in 4 metrics, which shows the BERT base as the best model in this research. Its score closely with the 2nd best performing model, DistillBERT with an accuracy difference of 0.0111, loss difference of 0.0406, validation accuracy of 0.0109, and F1 score of 0.015. Although the worst-performing roBERTa has a difference of 0.03 in F1 score from the best performing model, roBERTa has the smallest validation loss and difference between accuracy and validation accuracy, which shows a lack of overfitting that is shown in the other

two models. In this research, the best performing BERT base model is representing the text-based model for multimodal.

Table 4.6 Audio Test Result

Audio Model	Training Accuracy	Training Loss	Validation Accuracy	Validation Loss	F1 Score
bi-LSTM	0.32	1.3526	0.3294	1.3564	0.18
CNN + bi-LSTM	0.9767	0.0604	0.4226	3.7153	0.4075
CNN	0.9789	0.0603	0.5294	2.6747	0.5375

In **Table *4.6***, the best performing model is the stand-alone CNN model with the 4 best results of accuracy, loss, validation accuracy, and F1 score metrics. When incorporating bi-LSTM in the CNN model, there is a reduction in validation accuracy and an increase of loss. The bi-LSTM model itself has the poorest performance, with the lowest accuracy, validation accuracy, F1 score, and biggest loss.

Table 4.7 Augmented Audio Test Result

Audio Model	Training Accuracy	Training Loss	Validation Accuracy	Validation Loss	F1 Score
bi-LSTM	0.308	1.3593	0.314	1.3622	0.13
CNN + bi-LSTM	0.9708	0.774	0.4851	3.1189	0.485
CNN	0.9821	0.0734	0.5348	4.127	0.5425

Table *4.7* shows the result of training the augmented data with the same model as **Table *4.6***. Aside from the bi-LSTM model which shows the poorer result, there is an increase in both the validation accuracy of the standalone CNN model and mixed of CNN and bi-LSTM model. With a little difference in accuracy and a big increase in validation accuracy, the augmented technique in audio is proven to reduce overfitting in the model. There is also a small increase of loss in both the standalone CNN model and mixed CNN bi-LSTM model. Based on **Table *4.6*** and **Table *4.7***, the CNN model is the best unimodal model in this research for audio data, augmented or not.

4.2.2. Multimodal Model

The multimodal model used in this research is combined from the top-performing model in every modality. The multimodal built in this research consists of a possible combination of text, audio, augmented audio, and mixed motion capture model. All the unimodal models that will be used in multimodal are presented in **Table *4.8***

Table 4.8 Top Performing Model for Every Modality

Model	Modality	Training Accuracy	Training Loss	Validation Accuracy	Validation Loss	F1 Score
CNN	Audio	0.9821	0.0734	0.5348	4.127	0.5425
BERT base	Text	0.8786	0.3256	0.6715	1.0945	0.675
CNN	Motion Capture	0.8414	0.4124	0.655	1.1344	0.6625

Table 4.9 Multimodal Test Result

No	Multimodal Model	Training Accuracy	Training Loss	Validation Accuracy	Validation Loss	F1 Score
1	Text + Motion Capture	0.9389	0.1703	0.7638	0.9509	0.765
2	Text + Audio	0.9758	0.0738	0.695	1.7082	0.7
3	Audio + Motion Capture	0.9848	0.0452	0.6724	3.1287	0.6775
4	Text + Motion Capture + Audio	0.9728	0.0776	0.6561	1.9717	0.66
5	Text + Motion Capture + Audio (Tripathi et al., 2018)	0.9666	0.0836	0.6731	2.1394	-

In **Table *4.9***, this research roughly compares the last training results of multimodal combination text, audio, and motion capture modality model in

(Tripathi et al., 2018) as model 5 with its proposed designed model as model 4. The evaluated metrics used in the evaluation of model 5 are the same as model 4. However, the comparison is not fair, because of the difference in hyperparameters and split ratio. Model 5 use 30 epoch, a batch size of 64, and a split ratio of 77:22 while this research applies 50 epoch, batch size of 32, and split ratio of 8:2. The results show that model 4 performs a little better at training and a little worse at predicting the actual dataset. The big difference between the training and shows that there is a possibility of overfitting in model 4.

Looking at the whole **Table *4.8***, model 1 is the top-performing model with the 3 best metrics, which are validation accuracy, validation loss, and F1 Score. Although model 3 achieves the best accuracy and the smallest loss, its validation accuracy is lower than the other model and the loss is the biggest which shows that model 3 is also quite overfitted. Other model created from the audio unimodal model also tends to show a big gap difference between validation and non-validation results, which shows that the models are overfitted. To reduce the overfitting, this research tried to augment the audio data to reduce the overfitting in the models. The result can be seen in **Table *4.9*.**

Table 4.10 Multimodal Test Audio Augmented Result

No	Multimodal Model	Training Accuracy	Training Loss	Validation Accuracy	Validation Loss	F1 Score
1	Text + Motion Capture	0.9389	0.1703	0.7638	0.9509	0.765
2	Text + Audio	0.9776	0.0674	0.6869	1.7712	0.6975
3	Audio + Motion Capture	0.9721	0.0898	0.6615	1.8232	0.665
4	Text + Motion Capture + Audio	0.9898	0.0324	0.7077	2.0518	0.7125

Compared to **Table *4.8***, the model in **Table *4.9*** Compared to Table 4.8, the model in Table 4.9 which takes augmented audio data as an input has some impact on validation accuracy, loss, and F1 score. Model 4 performance is better than the non-augment data by quite a margin, with an increase of 0.05 validation accuracy and 0.05125 of f1 score. Although model 4 also has an increase of 0.0801 of validation loss which is proven to decrease the performance, it is counted as an improvement if compared with the increased value. With a little increase of accuracy and a big increase of validation accuracy, it can be said that the audio

augmentation reduces the overfitting on model 4. The augmentation impact on other models is not significant and fixed, it can be seen in the small decrease of accuracy, validation accuracy, and f1 score in model 2 and model 3.

Based on the result seen in **Table *4.8*** and **Table *4.9***, more modality is not always improving the performance of the model, which is shown that model 1 which consists of 2 modalities outperforms model 4 which consists of 3 modalities. It can also be seen that the reason that model 1 is better than model 4 is that model 4 is incorporated with another modality, which is audio. Audio modality is the worst of the unimodal model used for multimodal combinations. **Table *4.10*** shows the improvement of model 4, which can be traced back as a result of improvement in the audio model, so the performance of the unimodal model will implicate the multimodal model created from it. Although having more modality is not always improving the performance, the application of multimodal is better than unimodal in most cases, which can be seen in the comparison of unimodal BERT base compared to models 1, 2, and 3 which incorporated another unimodality that is not better than BERT base if counted alone.

CHAPTER V
CONCLUSION AND FUTURE WORK

5.1. Conclusion

Although the multimodal approach can improve the performance of the learning model, there is still a lack of research that discusses the factor that makes the model improve. To find out the factors that determine the improvement of multimodal, this research proposed several state-of-the-art unimodal model and multimodal models to evaluate the characteristics of every modality and find the dominant factor that influences the performance of the multimodal model built from it. The Dataset proposed in this research is IEMOCAP dataset. The modalities used in this research are text, motion captures which are further split into the head, hand, and rotated, and audio. This research also applies data augmentation on audio in the preprocessing step to reduce overfitting occurred during the research. Overall, this research evaluates a total of 22 unimodal model which split into 16 model for 4 motion capture, 3 model for text, and 3 model audio modality which are evaluated twice with addition of voice augmentation and 4 multimodal models which consists of all possible combinations of top performing unimodal model. This research also compares the work with (Tripathi et al., 2018) as a baseline model.

After evaluation, the CNN model performances in audio and motion capture are the highest and the BERT base model achieves the highest performance in text modality. The experiment results on both unimodal and multimodal shows that the performance on the multimodal depends on the performance of every unimodality. The multimodal model's performance still performs better than the best unimodal model it is parted off. Although the multimodal performs better, the utilization of

more modalities will not always improve the performance of the model, which can be seen that the results of multimodal model 1's performance are better than model 3. The performance of multimodal model 1 also boosted after the implementation of data augmentation on the audio model, although the performance of other multimodal models is not positively affected.

5.2. Future Works

This experiment is not perfect, and many parts can still be improved in the future. Some of the improvements that can be taken in the future are modifying or changing the unimodal model to improve the performance of the model. Moreover, in the future, another new modality may be introduced. With a new modality, maybe a new perspective can be gained. There is also a possibility of applying another way of preprocessing the data which further boosts the performance of the model. Lastly, a new method to apply train and test data may change the performance result and open further discussion.

CHAPTER VI
REFERENCES

Affonso, C., Rossi, A. L. D., Vieira, F. H. A., & de Carvalho, A. C. P. de L. F. (2017). Deep learning for biological image classification. *Expert Systems with Applications*, *85*, 114–122. https://doi.org/10.1016/j.eswa.2017.05.039

Baruah, A., Das, K., Barbhuiya, F., & Dey, K. (2020). Aggression Identification in {E}nglish, {H}indi and {B}angla Text using {BERT}, {R}o{BERT}a and {SVM}. *Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying*, *May*, 76–82.

Becker, K., Moreira, V. P., & dos Santos, A. G. L. (2017). Multilingual emotion classification using supervised learning: Comparative experiments. *Information Processing and Management*, *53*(3), 684–704. https://doi.org/10.1016/j.ipm.2016.12.008

Busso, C., Bulut, M., Lee, C. C., Kazemzadeh, A., Mower, E., Kim, S., Chang, J. N., Lee, S., & Narayanan, S. S. (2008). IEMOCAP: Interactive emotional dyadic motion capture database. *Language Resources and Evaluation*, *42*(4), 335–359. https://doi.org/10.1007/s10579-008-9076-6

Cambria, E., Hazarika, D., Poria, S., Hussain, A., & Subramanyam, R. B. V. (2018). Benchmarking multimodal sentiment analysis. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *10762 LNCS*, 166–179. https://doi.org/10.1007/978-3-319-77116-8_13

Chaparro, V., Gomez, A., Salgado, A., Quintero, O. L., Lopez, N., & Villa, L. F. (2018). Emotion Recognition from EEG and Facial Expressions: A Multimodal Approach. *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS*, *2018-July*, 530–533. https://doi.org/10.1109/EMBC.2018.8512407

Chollet, F. (2017). Xception: Deep Learning with Depthwise Separable Convolutions. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1251–1258.

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding. *ArXiv*.

Devlin, J., Chang, M.-W., Lee, K., Google, K. T., & Language, A. I. (2018). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *Naacl-Hlt 2019*, *Mlm*. https://github.com/tensorflow/tensor2tensor

Es-Sabery, F., Hair, A., Qadir, J., Sainz-De-Abajo, B., Garcia-Zapirain, B., & Torre-Diez, I. (2021). Sentence-Level Classification Using Parallel Fuzzy Deep

Learning Classifier. *IEEE Access*, *9*, 17943–17985. https://doi.org/10.1109/ACCESS.2021.3053917

Gallo, I., Calefati, A., & Nawaz, S. (2017). Multimodal Classification Fusion in Real-World Scenarios. *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, 36–41. https://doi.org/10.1109/ICDAR.2017.326

González-Carvajal, S., & Garrido-Merchán, E. C. (2020). Comparing BERT against traditional machine learning text classification. *ArXiv*.

Graves, A., & Schmidhuber, J. (2005). Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks*, *18*(5–6), 602–610. https://doi.org/10.1016/j.neunet.2005.06.042

Guggenmos, M., Schmack, K., Veer, I. M., Lett, T., Sekutowicz, M., Sebold, M., Garbusow, M., Sommer, C., Wittchen, H.-U., Zimmermann, U. S., Smolka, M. N., Walter, H., Heinz, A., & Sterzer, P. (2020). A multimodal neuroimaging classifier for alcohol dependence. *Scientific Reports*, *10*(1), 298. https://doi.org/10.1038/s41598-019-56923-9

Gunatilaka, A. H., & Baertlein, B. A. (2001). Feature-level and decision-level fusion of noncoincidently sampled sensors for land mine detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *23*(6), 577–589. https://doi.org/10.1109/34.927459

Hochreiter, S., & Urgen Schmidhuber, J. (1997). Long Shortterm Memory. *Neural Computation*, *9*(8), 17351780.

Huang, F., Zhang, X., Zhao, Z., Xu, J., & Li, Z. (2019). Image–text sentiment analysis via deep multimodal attentive fusion. *Knowledge-Based Systems*, *167*, 26–37. https://doi.org/10.1016/j.knosys.2019.01.019

Huang, Y., Yang, J., Liao, P., & Pan, J. (2017). Fusion of Facial Expressions and EEG for Multimodal Emotion Recognition. *Computational Intelligence and Neuroscience*, *2017*. https://doi.org/10.1155/2017/2107451

Kingma, D. P., & Ba, J. L. (2015). Adam: A method for stochastic optimization. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*.

Kirange D. K*, D. R. (2012). Emotion Classification of News Headlines Using Svm. *Asian Journal of Computer Science and Information Technology*, *2*(5).

Kumar, R., & Ojha, A. K. (2019). KMI-Panlingua at HASOC 2019: SVM vs BERT for hate speech and offensive content detection. *CEUR Workshop Proceedings*, *2517*, 285–292.

Lecun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436–444. https://doi.org/10.1038/nature14539

Li, B., Liu, T., Zhao, Z., Wang, P., & Du, X. (2017). *Neural Bag-of-Ngrams*. www.aaai.org

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., & Stoyanov, V. (2019). *RoBERTa: A Robustly Optimized BERT Pretraining Approach*. http://arxiv.org/abs/1907.11692

Ly, A., Uthayasooriyar, B., & Wang, T. (2020). *A survey on natural language processing (nlp) and applications in insurance*.

Mittal, T., Bhattacharya, U., Chandra, R., Bera, A., & Manocha, D. (2020). M3ER: Multiplicative multimodal emotion recognition using facial, textual, and speech cues. *AAAI 2020 - 34th AAAI Conference on Artificial Intelligence*, 1359–1367. https://doi.org/10.1609/aaai.v34i02.5492

Mohri, M., Rostamizadeh, A., & Talwalkar, A. (2013). Foundations of Machine Learning second edition. *Вестник Казнму*, *№3*(1), с.30.

Mondal, A., & Kaur, A. (2016). Comparative Study of Feature Level and Decision Level Fusion in Multimodal Biometric Recognition of Face , Ear and Iris. *International Journal of Computer Science and Mobile Computing*, *5*(5), 822–842.

Mosser, L., Dubrule, O., & Blunt, M. J. (2018). Stochastic Reconstruction of an Oolitic Limestone by Generative Adversarial Networks. *Transport in Porous Media*, *125*(1), 81–103. https://doi.org/10.1007/s11242-018-1039-9

Muniasamy, A., & Alasiry, A. (2020). Deep learning: The impact on future eLearning. *International Journal of Emerging Technologies in Learning*, *15*(1), 188–199. https://doi.org/10.3991/IJET.V15I01.11435

Nanni, L., Maguolo, G., & Paci, M. (2019). *Data augmentation approaches for improving animal audio classification*.

Nitsch, V., & Popp, M. (2014). Emotions in robot psychology. *Biological Cybernetics*, *108*(5), 621–629. https://doi.org/10.1007/s00422-014-0594-6

Odaibo, S. G. (2019). *Is "Unsupervised Learning" a Misconceived Term?*

Padarian, J., & Fuentes, I. (2019). Word embeddings for application in geosciences: development, evaluation, and examples of soil-related concepts. *SOIL*, *5*(2), 177–187. https://doi.org/10.5194/soil-5-177-2019

Paul, S., & Saha, S. (2020). CyberBERT: BERT for cyberbullying identification. *Multimedia Systems*. https://doi.org/10.1007/s00530-020-00710-4

Qaiser, S., & Ali, R. (2018). Text Mining: Use of TF-IDF to Examine the Relevance of Words to Documents. *International Journal of Computer Applications*, *181*(1), 25–29. https://doi.org/10.5120/ijca2018917395

Ren, X., Guo, H., Li, S., Wang, S., & Li, J. (2017). *A Novel Image Classification Method with CNN-XGBoost Model* (pp. 378–390). https://doi.org/10.1007/978-3-319-64185-0_28

Rius, A., Ruisánchez, I., Callao, M. P., & Rius, F. X. (1998). Reliability of analytical systems: Use of control charts, time series models and recurrent neural networks (RNN). *Chemometrics and Intelligent Laboratory Systems*, *40*(1), 1–18. https://doi.org/10.1016/S0169-7439(97)00085-3

Salamon, J., & Bello, J. P. (2016). *Deep Convolutional Neural Networks and Data Augmentation for Environmental Sound Classification*. https://doi.org/10.1109/LSP.2017.2657381

Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019). *DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter*. http://arxiv.org/abs/1910.01108

Schuster, M., & Nakajima, K. (2012). Japanese and Korean voice search. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 5149–5152. https://doi.org/10.1109/ICASSP.2012.6289079

Sony, S., Dunphy, K., Sadhu, A., & Capretz, M. (2021). A systematic review of convolutional neural network-based structural condition assessment techniques. *Engineering Structures*, *226*. https://doi.org/10.1016/j.engstruct.2020.111347

Sultan, H. H., Salem, N. M., & Al-Atabany, W. (2019). Multi-Classification of Brain Tumor Images Using Deep Neural Network. *IEEE Access*, *7*, 69215–69225. https://doi.org/10.1109/ACCESS.2019.2919122

Thanapattheerakul, T., Amoranto, J., Mao, K., & Chan, J. H. (2018). Emotion in a century: A review of emotion recognition. *ACM International Conference Proceeding Series*. https://doi.org/10.1145/3291280.3291788

Thang Duong, C., Lebret, R., & Aberer, K. (2017). Multimodal Classification for Analysing Social Media. *ArXiv*.

Tripathi, S., & Beigi, H. (2018). Multi-Modal Emotion Recognition on Iemocap With Neural Networks. *ArXiv*, 1–5.

Tripathi, S., Tripathi, S., & Beigi, H. (2018). *Multi-Modal Emotion recognition on IEMOCAP Dataset using Deep Learning*.

Wu, M., & Chen, L. (2016). Image recognition based on deep learning. *Proceedings - 2015 Chinese Automation Congress, CAC 2015*, 542–546. https://doi.org/10.1109/CAC.2015.7382560

Ye, Z., Tafti, A. P., He, K. Y., Wang, K., & He, M. M. (2016). SparkText: Biomedical Text Mining on Big Data Framework. *PLOS ONE*, *11*(9), e0162721. https://doi.org/10.1371/journal.pone.0162721

Yingge, H., Ali, I., & Lee, K.-Y. (2020). Deep Neural Networks on Chip - A Survey. *2020 IEEE International Conference on Big Data and Smart Computing (BigComp)*, 589–592. https://doi.org/10.1109/BigComp48618.2020.00016

Yoon, S., Byun, S., & Jung, K. (2019). Multimodal Speech Emotion Recognition Using Audio and Text. *2018 IEEE Spoken Language Technology Workshop, SLT 2018 - Proceedings*, 112–118. https://doi.org/10.1109/SLT.2018.8639583

Youn, Y.-S., Nam, K.-M., Song, H.-J., Kim, J.-D., Park, C.-Y., & Kim, Y.-S. (2016). Classification Performance of Bio-Marker and Disease Word using Word Representation Models. *International Journal of Bio-Science and Bio-Technology*, *8*(1), 295–302. https://doi.org/10.14257/ijbsbt.2016.8.1.26

Zhang, Y., Jin, R., & Zhou, Z. H. (2010). Understanding bag-of-words model: A statistical framework. *International Journal of Machine Learning and Cybernetics*, *1*(1–4), 43–52. https://doi.org/10.1007/s13042-010-0001-0

Zheng, C., Wang, C., & Jia, N. (2021). Emotion Recognition Model Based on Multimodal Decision Fusion. *Journal of Physics: Conference Series*, *1873*(1), 012092. https://doi.org/10.1088/1742-6596/1873/1/012092

CURRICULUM VITAE

Name : Kenny

Place and Date of Birth : Jakarta, 30 May 1999

Address : Perumahan Gading Arcadia blok A no 8, Pegangsaan Dua, Kelapa Gading, Jakarta Utara

Education background:

- 2005 - 2011 : Elementary School at SD Tunas Karya II Elementary school
- 2011 – 2014 : Middle School at Tunas Karya Middle school
- 2014 – 2017 : High School at Don Bosco II high school
- 2017 – 2021 : Bachelor's degree at Bina Nusantara University
- 2021 – now : Master's degree at Bina Nusantara University

Work background:

- 2018 – 2020 : Teaching Assistant at Bina Nusantara University
- 2020 – 2021 : Network administrator at Bina Nusantara University
- 2021 – now : Software Engineer at Tokopedia