CHAPTER 3
RESEARCH METHODOLOGY

1. Data collection

Dataset will be collected from Twitter using a python *snscrape* library which is easy to use to collect tweets as a dataset from Twitter. In this research the dataset was specifically about tweets that mention *@TheBatman* account and only limited for tweets from March 5 until June 5. With the total of 1950 tweets, the dataset needs to be reviewed whether the tweets are positive or negative with the help of 2 reviewers. After the data has been reviewed, comparison between 2 reviewers needs to be done and to find its Kappa value. Kappa value is frequently used to test the reliability of classification models. After the Kappa value has been found, non-matching marks between 2 reviewers will be conducted and will be deleted if it's not a match.

2. Data pre-processing

Data preprocessing is a process of preparing the raw data and making it suitable for a machine learning model. It is the first and most crucial step that needs to be done while creating a machine learning model. Data preprocessing needs to be done because real-world data generally contains noises, missing values, and maybe in an unusable format which cannot be directly used for machine learning models. Hence, data preprocessing is a required task for cleaning the data and making it suitable for a machine learning model which will also increase the accuracy and efficiency of a machine learning model.

In this case, processing the tweets needs to be done in order to increase the model's performance during training. This task included removing html tags, punctuations and numbers, single character removal, removing multiple spaces, and lastly stopwords removal

3. Word embedding

Word embedding is a learned representation for text where words that have the same meaning have a similar representation. Word embeddings are in fact a class of techniques where individual words are represented as real-valued vectors in a predefined vector space. Each word is mapped to one vector and the vector values are learned in a way that resembles a neural network, and hence the technique is often lumped into the field of deep learning.

In word embedding each word is represented by a real-valued vector, often tens or hundreds of dimensions. This is contrasted to the thousands or millions of dimensions required for sparse word representations, such as a one-hot encoding.

There are 3 kinds of word embedding algorithms like embedding layer, word2vec, and especially in this research Global Vector (*GloVe*) are used. The global vectors algorithm is an extension to the word2vec method for efficiently learning word vectors developed by Pennington, et al. at Stanford. *GloVe* is an approach to pair both the global statistics of matrix factorization techniques like Latent Semantic Analysis (LSA) with the local context-based learning in word2vec.

Rather than using a window to define local context, *GloVe* constructs an explicit word-context or word co-occurrence matrix using statistics across the whole text corpus. The result is a learning model that may result in generally better word embedding. Which is why in this research pre-trained *GloVe* word embedding with 100-dimensional vectors will be used. *GloVe* with 100 dimensions will be loaded and a dictionary will be created which will contain words as keys and their embedding list as values.

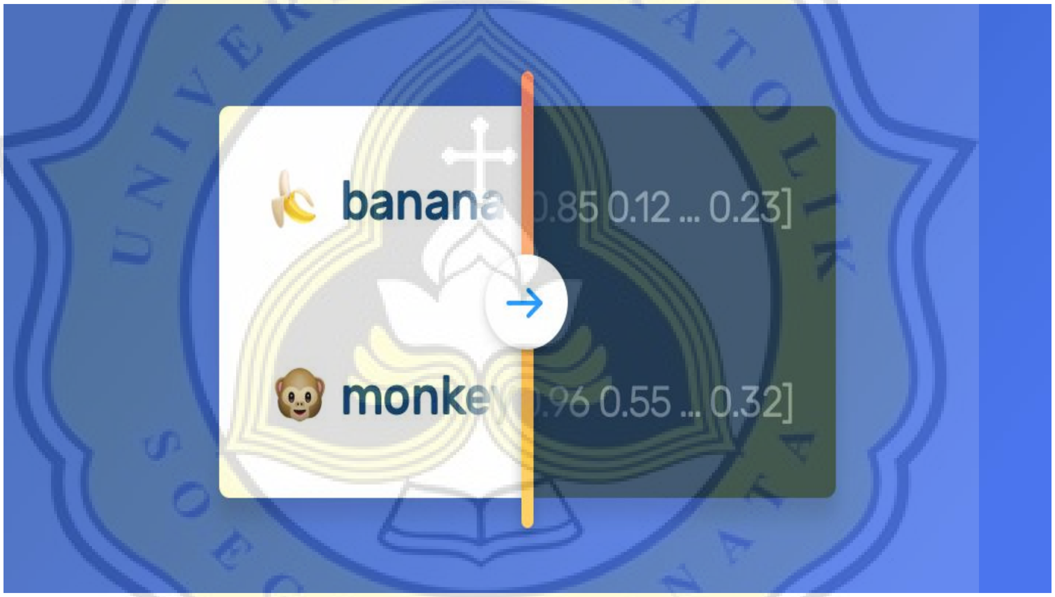

Figure 3.1 how word embeddings work for neural network models.

4. Implementation

The implementation in this step will be divided into three kinds of neural network, which is Convolutional Neural Network (CNN), and lastly Long Short-Term Memory neural network (LSTM). For each embedding layer will be set the same and the optimizer of each model will use the “adam” optimizer and use sequential models so that the models are equal.

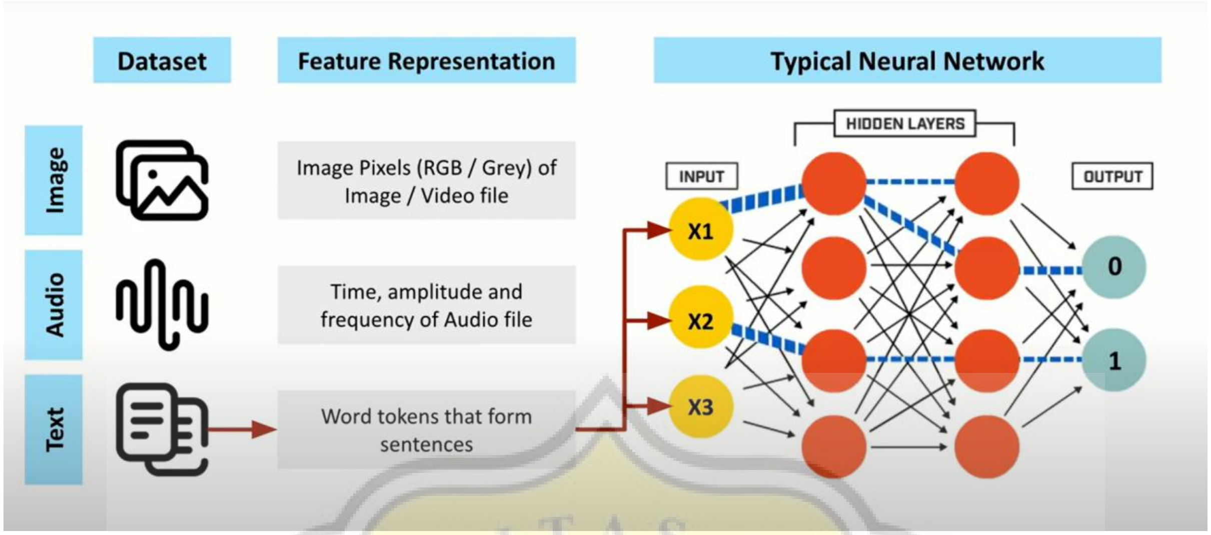
Dataset
Feature Representation
Typical Neural Network
Image
Audio
Text
Image Pixels (RGB / Grey) of Image / Video file
Time, amplitude and frequency of Audio file
Word tokens that form sentences
HIDDEN LAYERS
INPUT
OUTPUT
X1
X2
X3
0
1

Figure 3.2 how neural network models attempted to mimic the computation abilities of the human brain.

Feature Representation holds an important role for neural networks to perform better. For image data, image pixels form the basis of derivation of these features. For sound; it's the time, amplitude, and frequency of the audio file and in this research text data is the word tokens that form the user review sentences.

5. Evaluation

After the data has been implemented a comparison will be conducted between the results of each neural network model to decide which model is most suitable for textual sentiment analysis.