

Attention-Based BiLSTM For Negation Handling In Sentimen Analysis

Riszi Wijayatun Pratiwi^{*1}, Yunita Sari², Yohanes Suyanto³

¹Master Program of Computer Science, FMIPA UGM, Yogyakarta, Indonesia

^{2,3}Department of Computer Science and Electronics, FMIPA UGM, Yogyakarta, Indonesia

e-mail: ^{*1}riszkiwijayatun@mail.ugm.ac.id, ²yunita.sari@ugm.ac.id, ³yanto@ugm.ac.id

Abstrak

Penelitian tentang analisis sentimen beberapa tahun ini telah terjadi peningkatan. Akan tetapi pada penelitian analisis sentimen masih sedikit yang menggagas tentang penanganan negasi, salah satunya dalam kalimat Bahasa Indonesia. Hal ini mengakibatkan kalimat yang mengandung unsur kata negasi belum ditemukan polaritasnya secara tepat.

Tujuan dari penelitian ini adalah menganalisis pengaruh kata negasi berbahasa Indonesia. Berdasarkan kelas positif, netral dan negatif, dengan menggunakan attention-based Long Short Term Memory dan Metode ekstraksi fitur word2vec dengan arsitektur Continuous bag-of-word (CBOW). Dataset yang digunakan berupa data dari Twitter. Performa model dilihat pada nilai akurasi.

Penggunaan word2vec dengan arsitektur CBOW dan penambahan layer attention pada metode Long Short Term Memory (LSTM) dan Bidirectional Long short Term Memory (BiLSTM) memperoleh hasil akurasi 78.16% dan untuk BiLSTM menghasilkan akurasi 79.68%. sedangkan pada algoritma FSW 73.50% dan FWL 73.79%. Bisa disimpulkan attention based BiLSTM memiliki akurasi tertinggi, akan tetapi penambahan layer attention pada metode Long Short Term Memory tidak terlalu signifikan untuk penanganan negasi. karena pada penambahan layer attention tidak dapat menentukan kata yang ingin diperhatikan.

Kata kunci— LSTM, attention-based LSTM, BiLSTM, Attention based BiLSTM Negasi, Analisis sentimen.

Abstract

Research on sentiment analysis in recent years has increased. However, in sentiment analysis research there are still few ideas about the handling of negation, one of which is in the Indonesian sentence. This results in sentences that contain elements of the word negation have not found the exact polarity.

The purpose of this research is to analyze the effect of the negation word in Indonesian. Based on positive, neutral and negative classes, using attention-based Long Short Term Memory and word2vec feature extraction method with continuous bag-of-word (CBOW) architecture. The dataset used is data from Twitter. Model performance is seen in the accuracy value.

The use of word2vec with CBOW architecture and the addition of layer attention to the Long Short Term Memory (LSTM) and Bidirectional Long Short Term Memory (BiLSTM) methods obtained an accuracy of 78.16% and for BiLSTM resulted in an accuracy of 79.68%. whereas in the FSW algorithm is 73.50% and FWL 73.79%. It can be concluded that attention based BiLSTM has the highest accuracy, but the addition of layer attention in the Long Short Term Memory method is not too significant for negation handling. because the addition of the attention layer cannot determine the words that you want to pay attention to.

Keywords—LSTM, attention-based LSTM, BiLSTM, Attention based Negation BiLSTM, sentiment analysis

1. INTRODUCTION

Sentiment analysis is a field of study that analyzes a opinions, sentiments, evaluations, attitudes and emotions from written language [1]. The are many that have applied sentiment analysis to review. Research by [2] utilizes a dictionary-based method (lexicon based) to research sentiment analysis. From this research also have problems because word of negation cannot be handled. The handling of the word negation in Indonesian is still under consideration in the sentiment analysis. Negation has a big impact on sentiment analysis, if left untreated it can affect the polarity value [3].

Research on negation handling for Indonesian tweet by [4]. In this study, using algorithms First Sentiment Word (FSW) and Fixed Window Length (FWL) . However, this research using the dictionary as its base. The more complete the sentiment dictionary you have, the more sentiment words that can be detected.

Neural networks can achieve this important word using attention, focusing on part of a subset with information they're given [5]. [6] Using attention to handle negation words in Electronic Health Center (EHR) data. Attention mechanism that is combined in the Bidirectional Long Short Term Memory (BiLSTM) model is called attention-based BiLSTM. This shows that the attention based BiLSTM method is appropriate for classification of text.

For these reasons, in this study, we will discuss the handling of negation in sentimen analysis which has previously been studied by [4] using FSW and FWL algorithm. This study, the BiLSTM method is used which refers to [6]. While for the data used from [4]. Then the result will be compared between the fsw and fwl algorithm with attention in the BiLSTM method.

2. METHODS

2.1 Architecture System

In this research the system architecture that will be built has for parts, they are data collection, preprocessing, feature atraction, the last classification and evaluation systems. The process can be seen in Figure 1 and Figure 2.

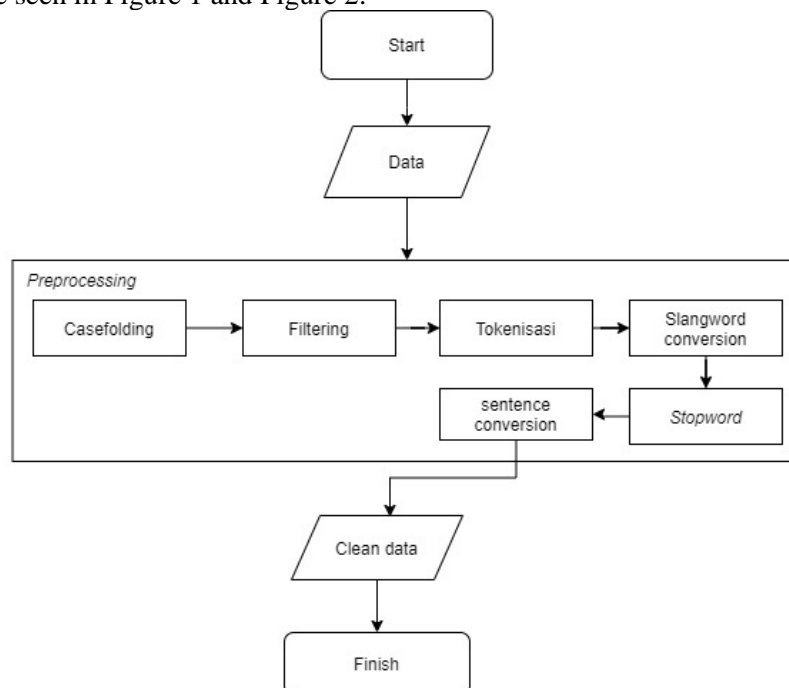


Figure 1 Data preprocessing process

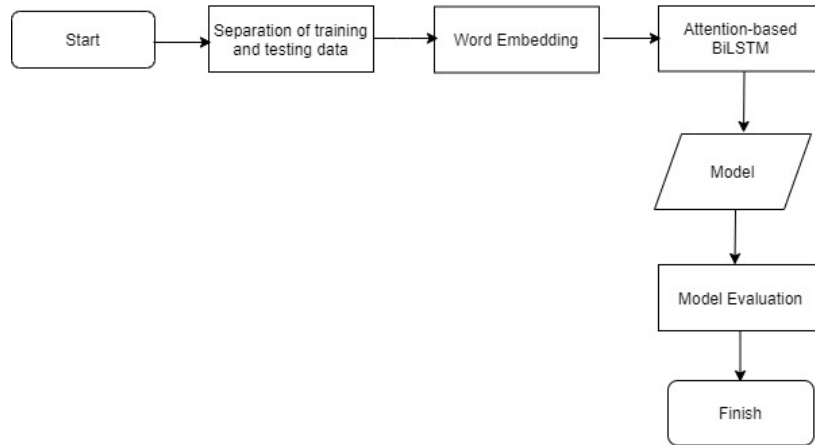


Figure 2 Modeling sentimen analysis

2.2 Data Collection

The data of reviews tweet used in this research uses a dataset from research [4]. Dataset used Indonesian Language tweet data that have been labelled as positive, neutral and negative.

2.3 Preprocessing

Preprocessing is very important in sentiment analysis, because preprocessing manage data to get data that is clean to be processed in the making word vectors and sentiment classification were more accurate [7]. As for the step in preprocessing is a) Case folding, b) Filtering, c) Tokenizing, d) slangwords conversion, and e) stopwords removal. For example of the preprocessing process can be seen in Table 1:

Table 1 The preprocessing Example

Preprocessing	Reviews
	Tidak salah pilihanku. Orang- orang baik sepertinya juga dukung jokowi. #Jokowimaju
Casefolding	tidak salah pilihanku. orang-orang baik sepertinya juga dukung jokowi #jokowimaju
Filtering	tidak salah pilihanku orang orang baik sepertinya juga dukung jokowi jokowimaju
Tokenizing	[tidak] [salah] [pilihanku] [orang] [orang] [baik] [sepertinya] [juga] [dukung] [jokowi] [jokowimaju]
Slangwords conversion	x (because in the example there are no non-standard words)
Stopword Removal	[tidak] [salah] [pilihanku] [orang] [orang] [baik] [dukung] [jokowi] [jokowimaju]

An explanation of each step can be seeb below :

2.3.1 Casefolding

Not all text documents are letter-consistent, so in this process can be change the letter characters in the comment to all lowercase characters.

2.3.2 Filtering

In this process adjustments are made by removing special characters and reviews such oter characters (\$,%*, and so on). This process will be eliminates words that do not match the parsed results. For example usernames that start with the symbol “@”, hashtags “#”, Uniform

Resource Locator (URL) and emoticons. Signs, symbol or numbers are omitted because they have just a little effect on labelling process.

2.3.3 Tokenizing

Tokenizing servers to break the review down into word units. The tokenizing process is carried out by looking at every space in the review. Based on these spaces the words can be separated.

2.3.4 Slangword conversion

Slangword conversion is the process of converting non-standard words into standard words. This stage is carried out using the help of a slangword dictionary and its equivalent in standard words. This stage will check the words contained in the slang word dictionary or not. If a nonstandard word is found in the slang word dictionary, the nonstandard word will be converted to the standard word found in the slang word dictionary.

2.3.5 Stopword Removal

This stage serves to eliminate words that have no influence (which, and, or, to, from, etc.) in the later classification process.

2.4 Sentence Conversion

The steps in this process creating a word dictionary, converting sentences into numbers, and padding. The results of this sentence conversion process will be used as input to the BiLSTM method. The first process is making a word dictionary that is used to provide the word is contained in a sentence in the tweet data that has gone through a preprocessing process

2.5 Word2vec

Word2Vec, developed by Thomas Mikolov, is an implementation of artificial neural networks that can process words from very large datasets in a relatively short time. This tool works by taking a corpus of text as input, then produces a vector representation of each word in the corpus as output [8]. There are two Word2Vec modeling architectures that can be used to represent word vectors, the architecture is Continuous Bag-of-words (CBOW) and Skip-gram [9]. In this study used Continuous bag of words (CBOW) architecture and skipgram. Vector size of 200 dimensions, this refers to [6].

2.6 Attention based bidirectional LSTM

First, build BiLSTM as a model. Taking word embedding as input. This layer will change the positive integer index in the input into a fixed-size vector based on the vector dimensions of the word dictionary based in word2vec model. In LSTM, this layer determines the previous input, whether it can pass in the cell state or not. What determines the data can be continued or not is the sigmoid layer called “forget gate”. Output 1 means “let pass” and 0 means “forget information” [10]. The calculation of the forget gate value was with equation (1).

$$f_t = \sigma (W_f \cdot [h_{t-1}, x_t] + b_i) \quad (1)$$

For the next step was to determine the new information that were going to be stored in the cell state. First sigmoid layer was called the input gate which determined which part to update. Then, the tanh layer created a new candidate value vector, ct to be added to the cell state. And the next step, the two were combiner to make an update to the state. To calculate the input gate value with equation (2) and the new candidate value with equation (3).

$$i_t = \sigma (W_i \cdot [h_{t-1}, x_t] + b_i) \quad (2)$$

$$\tilde{c}_t = \tanh (W_c \cdot [h_{t-1}, x_t] + b_c) \quad (3)$$

Then next step to updating the old cell state C_{t-1} into the new cell state C_t . By multiplying the old cell state with the forget gate f_t then added $i_t * \tilde{c}_t$. To be clearer, in the equation (4)

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \quad (4)$$

Next step is output gate. First, running the sigmoid layer which determined which cell would be the output, then place the cell state through the tanh and increased the output of the sigmoid gate. So that only the part we specified was the output. Calculation of output gate with equation (5) and (6).

$$O_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (5)$$

$$h_t = O_t * \tanh(C_t) \quad (6)$$

All hidden states are fed into a subsequent attention layer [11]. We added attention layer because not all words contribute equally to the negation detection. The normalized word weight is at obtained through a softmax function equation (8). The aggregate of all information in the sentence v is the weighted sum of each h_t with at , as corresponding weights.

$$u_t = \tanh(W_\alpha h_t + b_\alpha) \quad (7)$$

$$a_t = \frac{\exp u_t^T u}{\sum_r \exp u_r^T u} \quad (8)$$

$$v = \sum_t a_t h_t \quad (9)$$

This vector v is then fed to a fully connected layer with softmax activation to perform the final classification. The prediction is a vector $y \in \mathbb{R}^2$ with the probabilities of being positive, neutral or negative. The model architecture is shown in Figure 3 and Figure 4.

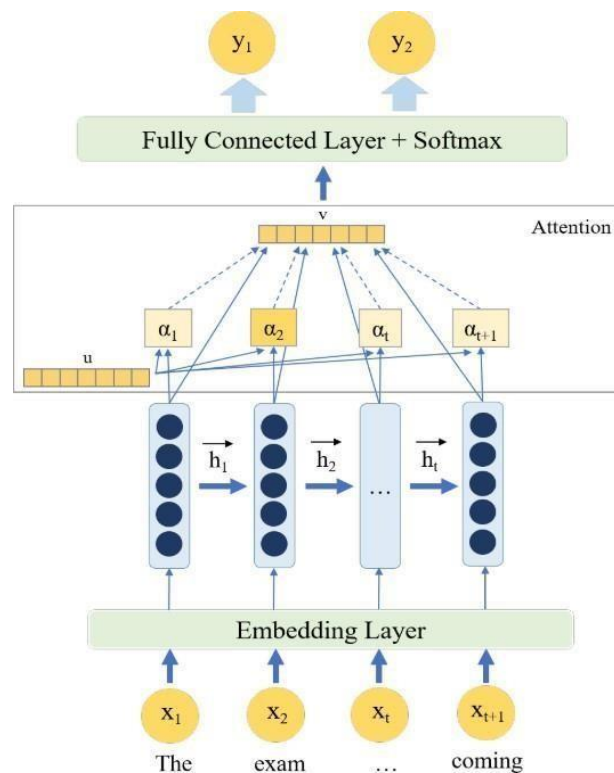


Figure 3 The architecture of attention based LSTM [11]

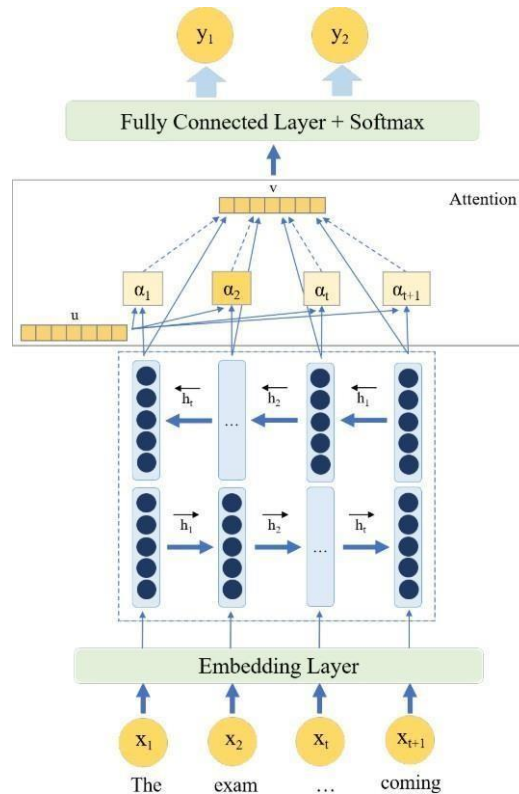


Figure 4 The architecture of attention based BiLSTM [11]

3. RESULTS AND DISCUSSION

This section discusses the results of sentiment classification test from the model that had been built. Sentiment classification test was done by measuring the values of accuracy, precision, recall and f1 score.

Total tweet rivew data used were 612 data for positive, neutral and negative sentiment. The training data used was 80% of the total data. While 20% of the total data was used as test data.

Classification test was done by measuring the value of accuracy, precision, recall and f1 score obtained by comparing each review with the results of the calculation of the attention based Bidirectional Long Short Term Memory method carried out by the system. There were two architecture of the attention based Long Short Term Memory method tested, attention based LSTM and attention based BiLSTM which were compared with LSTM, BiLSTM model and First sentiment window and fixed window length algorithm.

3.1 Attention based LSTM Classification Test

The classification results of the calculation sentiment classification using the attention based LSTM method is shown in Table 2.

Table 2 attention based LSTM tets result

Parameter	Value	Accuracy
Word2vec architecture	CBOW	78.15%
Neuron	200	
Epoch	20	
L2 regularization	0.00001	
Activation function	softmax	

Based on Table 2, it shows that the best accuracy result in the overall test of the attention based LSTM method was 78.15%. with the best parameters of CBOW architecture, 200 neurons each layer, 20 epochs, L2 regularization of 0.00001 and softmax activation function.

3.2 Attention based BiLSTM Classification Test

While overall in the test result from the calculation of sentiment classification using the attention based BiLSTM method showed that the accuracy value was better than the attention based LSTM. The overall accuracy value was 79.68%. This means that by increasing added BiLSTM model, it can increase the results of sentiment classification accuracy. The results sentiment classification using the attention based BiLSTM method are shown in Table 3.

Table 3 attention based BiLSTM test result

Parameter	Value	Accuracy
Word2vec architecture	CBOW	79.68%
Neuron	200	
Epoch	20	
L2 regularization	0.00001	
Activation function	softmax	

Based on Table 3, it shows that the best accuracy result in the overall test of the attention based BiLSTM method was 79.68% with the best test parameters of CBOW architecture, 200 neurons each layer, 200 epoch, L2 regularization of 0.00001 and softmax activation function.

3.4 LSTM Classification Test

To see the attention gain performance with LSTM model, we will compare it with LSTM model without attention model. And the result can be seen in Table 4.

Table 4 LSTM test result

Parameter	Value	Accuracy
Word2vec architecture	CBOW	75.60%
Neuron	200	
Epoch	20	
L2 regularization	0.00001	
Activation function	softmax	

Based on Table 4, it shows that the best accuracy result in the overall test of the LSTM method was 75.60% with the best test parameters of CBOW architecture, 200 neurons each layer, 200 epoch, L2 regularization of 0.00001 and softmax activation function.

3.5 BiLSTM Classification Test

In the BiLSTM classification test using same parameters with LSTM classification, and can be seen in Table 5.

Table 5 BiLSTM test result

Parameter	Value	Accuracy
Word2vec architecture	CBOW	76.87%
Neuron	200	
Epoch	20	
L2 regularization	0.00001	
Activation function	softmax	

Based on Table 3, it shows that the best accuracy result in the overall test of the BiLSTM method was 76.87% with the best test parameters of CBOW architecture, 200 neurons each layer, 200 epoch, L2 regularization of 0.00001 and softmax activation function.

3.6 FSW and FWL algorithm Test

The FSW and FWL uses the same data as the data in the attention based BiLSTM/LSTM method. The data had also gone through the same preprocessing process while the feature extraction used was TF (Term-Frequency). Furthermore. The results of the FSW and FWL Algorithm can be seen in Table 6.

Table 6 FSW and FWL algorithm result

	Accuracy	Precision	Recall	F1 score
FSW	67.32%	67.59%	67.58%	66.83%
FWL	68.79%	69.22%	68.88%	68.37%

3.7 Comparison of Accuracy Results

Comparison of the accuracy result from the calculation of sentiment classification using attention based LSTM, attention based BiLSTM and FSW and FWL algorithm is showed in Table 7. The classification of sentiment classification using the attention based BiLSTM method had a better value of accuracy than the attention based LSTM, BiLSTM or LSTM and FSW and FWL algorithm. Because in the attention just focus on capturing important word will then be reweighted can get maximum result. This study focuses on the use of attention to handle negation, but after doing this research it can be concluded that the addition of the attention layer is not to significant for dealing with negation words, because it is important to choose words randomly.

Table 7 Comparison of result

	Accuracy	Precision	Recall	F1 score
Attention based LSTM	78.15	76.56	77.45	77.25
Attention based BiLSTM	79.68	78.37	79.26	78.00
LSTM	75.60	76.82	76.50	75.30
BiLSTM	76.87	76.90	77,30	77.12
FSW	67.32	67.59	67.58	66.83
FWL	68.79	69.22	68.88	68.37

3.5 Result of Prediction

In this part we can see the result of true or false prediction. For details we can be seen Figure 5 and Figure 6.

3.5.1 Result of Prediction True

```

.....
sentimen : jokowi-jk janji sejahteraan tni-polri
actual_label : [1. 0. 0.]
prediksi : [2.7104342 1.0028452 1.00005 ]
.....
sentimen : seberapa mendesak kebutuhan bandara dan pelabuhan baru pasangan jokowi-jk menjangkikan pembangunan 10 bandara...
actual_label : [1. 0. 0.]
prediksi : [2.718277 1.0000017 1.0000001]
.....
sentimen : jokowi tunda kunjungi kalsel
actual_label : [1. 0. 0.]
prediksi : [2.7105775 1.0000277 1.0028147]
.....
sentimen : rumah jokowi-jk diresmikan di pelangkaraya cikidow. buat apa ini koalisi sama saya
actual_label : [1. 0. 0.]
prediksi : [2.7182312 1.0000093 1.0000093]
.....
sentimen : din syamsuddin puji kadar keislaman jokowi
actual_label : [1. 0. 0.]
prediksi : [2.678671 1.0140533 1.000724 ]
.....

```

Figure 5 Review Prediction True

Figure 5 shows the prediction results of a positive tweet review, for example in the following tweet review “jokowi-jk janji sejahteraan tni-polri” has a positive (1) actual value and the predictive value is [2.710, 1.002, 1.0005], then the prediction result is from the highest.

weight 2.710. the weett is in accordance with its actual value. This also applies to reviews of neutral tweet and negative tweets.

3.5.2 Result of Prediction Not True

```
-----
sentimen : semoga lulus dalam sandiwara ini prabowo-hatta tiba di rspad untuk tes kesehatan
actual_label : [0. 0. 1.]
prediksi : [1.8192905 1.0000354 1.0000046]
-----
```

Figure 6 Review Prediction Not True

In Figure 6 is a review of the negation sentence “semoga lulus sandiwara ini prabowo-hatta di rspad untuk tes kesehatan” actual values negative (-1) but predictive value positif (1), because the words passing “lulus sandiwara” means “lulus kebohongan”. And the predictive value is [1.819, 1.00035, 100046] which is positive.

4. CONCLUSIONS

Based on study, it can be concluded that:

1. The addition of layer attention to the Long Short Term Memory method is not significant for the handling of negation words, because the addition of layer attention cannot determine the words that you want to pay attention to. So that the words you want to pay attention to are obtained in the training process.
2. Attention based BiLSTM method produces more accuracy, namely 79.68% compared to attention based LSTM 78.15%, BiLSTM 76.87%, LSTM 75.60% and FSW 67.32%, FWL 68.79%.

5. SUGGESTION

1. There are still some weakness in this study that can be improved. Some suggestions for tuther research are as follows:
2. Try another method for handling negation words, because if you only add a layer of attention to the neural netowork, the negation word cannot be handled.
3. Using other feature extraction methods, you can uses other methods such as FastText and Glove.
4. If the research is in the form of handling negation, it is better if the dataset used is data tat contains a lot of negation sentences in order to see wethert the method can handle negation sentences.

REFERENCES

- [1] C. Sammut and G. I. Webb, Eds., *Encyclopedia of Machine Learning and Data Mining*. Boston, MA: Springer US, 2017.
- [2] N. Saputra, “Analisis Sentimen Berbasis Lexicon dan Emoticon,” Gadjah Mada University, Yogyakarta, 2015.
- [3] F. T. Al-Khawaldeh, “Speculation and Negation Detection for Arabic Biomedical Texts,” p. 5, 2019.
- [4] A. M. Ningtyas, “Pengaruh Penanganan Negasi dalam Bahas Indonesia untuk Pelabelan otomatis pada Analisis Sentimen Twitter,” Gadjah Mada University, Yogyakarta, 2016.
- [5] C. Olah and S. Carter, “Attention and Augmented Recurrent Neural,” 2016. <https://distill.pub/2016/augmented-rnns/>.

- [6] L. Chen, “Attention-Based Deep Learning System for Negation and Assertion Detection in Clinical Notes,” *Int. J. Artif. Intell. Appl.*, vol. 10, no. 01, pp. 1–9, Jan. 2019, doi: 10.5121/ijaia.2019.10101.
- [7] M. A. Nurrohmat and A. Sn, “Sentiment Analysis of Novel Review Using Long Short-Term Memory Method,” *IJCCS Indones. J. Comput. Cybern. Syst.*, vol. 13, no. 3, p. 209, Jul. 2019, doi: 10.22146/ijccs.41236.
- [8] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, “Distributed Representations of Words and Phrases and their Compositionality,” p. 9.
- [9] A. R. Isnain, A. Sihabuddin, and Y. Suyanto, “Bidirectional Long Short Term Memory Method and Word2vec Extraction Approach for Hate Speech Detection,” *IJCCS Indones. J. Comput. Cybern. Syst.*, vol. 14, no. 2, p. 169, Apr. 2020, doi: 10.22146/ijccs.51743.
- [10] C. Olah, “Understanding LSTM Networks.” <http://colah.github.io/posts/2015-08-Understanding-LSTMs/>.
- [11] G. I. Winata, O. P. Kampman, and P. Fung, “Attention-Based LSTM for Psychological Stress Detection from Spoken Language Using Distant Supervision,” in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Calgary, AB, Apr. 2018, pp. 6204–6208, doi: 10.1109/ICASSP.2018.8461990.