Aspect-level Sentiment Analysis for Social Media Data in the Political Domain using Hierarchical Attention and Position Embeddings

Renny Pradina Kusumawardani
Department of Information Systems
Institut Teknologi Sepuluh Nopember (ITS)
Surabaya, Indonesia
renny.pradina@gmail.com

Muhammad Wildan Maulidani
Department of Information Systems
Institut Teknologi Sepuluh Nopember (ITS)
Surabaya, Indonesia
maulidaniw@gmail.com

***Abstract*—In this paper we present our work on aspect-level sentiment analysis on social media data, specifically in the political domain. Aside from being linguistically irregular, political tweets are often ambiguous or contain sentiments of opposite polarity. To address this, we use a deep learning architecture with a hierarchical attention and position embeddings to enable a finer-grained analysis of sentiments at different positions in the text. Our dataset consists of 3022 tweets on the politics domain in Bahasa Indonesia having 1514 unique aspects. We find that there are two important factors for model performance: first, the use of a gating mechanism of appropriate complexity – in our case, LSTM gives the best performance in terms of accuracy and outperforms GRU and RNN by almost 7% in average recall. Second, the use of a trainable embeddings pre-trained on data in similar domains – a trainable Word2Vec embeddings trained on social media data in Bahasa Indonesia gives more than 4% better accuracy than without trainable embeddings. Our analysis also shows that correctly predicted tweets have more variance in attention weights, in contrast to incorrectly predicted ones to which input tokens are often assigned similar weights. This indicates the usefulness of attention mechanism in an aspect-based sentiment analysis.**

Keywords*—*sentiment analysis, aspect-level, social media, deep learning, hierarchical attention, position embeddings, political domain

I. Introduction

Aspect-based sentiment analysis has gained popularity in recent years to its objective to extract the sentiment polarity of specific aspects in a text containing human opinions on a particular subject. It aims to answer the ambiguity of assigning a single polarity label to a text, when it is entirely possible that it consists of different sentiment polarities, assigned to various aspects. For example, in the sentence "Infrastructure development is now very advanced and good, but unfortunately the country's finances are bad because of continuing debt", the sentiment towards the "development" aspect is positive while the sentiment for the "state finance" aspect is negative. This makes aspect level sentiment analysis necessary rather than just the overall sentiment analysis of the text. It has been shown in previous studies that the interaction between aspects and the text as a whole is important in sentiment analysis [4].

In recent years, more and more neural network-based architectures have been proposed for aspect-based sentiment analysis and have succeeded in achieving state-of-the-art results. [5] propose the use of position information from aspects when encoding sentences. As in the previous sentence, the additional words available are 'very advanced', 'good' and 'bad'. If position information is used in the appropriate additional word model for aspects of 'state finance' it is a word 'bad' which is closer in position to aspects, so that the use of position information is used through position embedding in the hierarchical attention-based position network (HAPN) architecture for analysis aspect level sentiment.

Previous studies used semi-detailed datasets consisting of restaurant or laptop datasets of SemEval 2014. In this work we use social media data has its own difficulties such as short sentences, slang words, grammatical errors and spelling, making it challenging in sentiment analysis. In this research, hierarchical attention based on position aware network (HAPN) is applied for the analysis of aspect-level sentiments on social media data in the political domains, particularly on Twitter. Political tweets, and political opinions in general, are often more difficult to analyze, as they are frequently ambiguous or having more than one polarity of sentiments, each associated to different aspects of the discussion. Therefore, we use HAPN as it allows the model to focus attention to different regions of a sentence. We study the effect of various underlying basic sequential mechanism such as LSTM and GRU, the use of pre-trained embeddings, and different optimizers, learning rates, and dropout values. We are also interested in the question of whether attention mechanism is indeed useful for this task, therefore we also analyze the weights assigned to input tokens.

II. Related Work

Various approaches have been proposed to overcome the problem of aspect level sentiment analysis. Many neural network models have been proposed in recent years. Most of the models in the aspect level sentiment analysis pay more attention to the use of target information or aspects and words in sentences. [1] propose attention-based neural networks that introduce multiple mechanisms of multiple-attention to capture sentiments from features located far from the target. [5] propose the use of position embeddings when modeling sentences to produce a basic representation of position, as well as the use of attention mechanisms that are integrated in various layers. From the various models above, we explore the use of hierarchical attention-based position aware network (HAPN) model for our data. Particularly, we study how social

978-1-7281-8235-3/20/$31.00 ©2020 IEEE

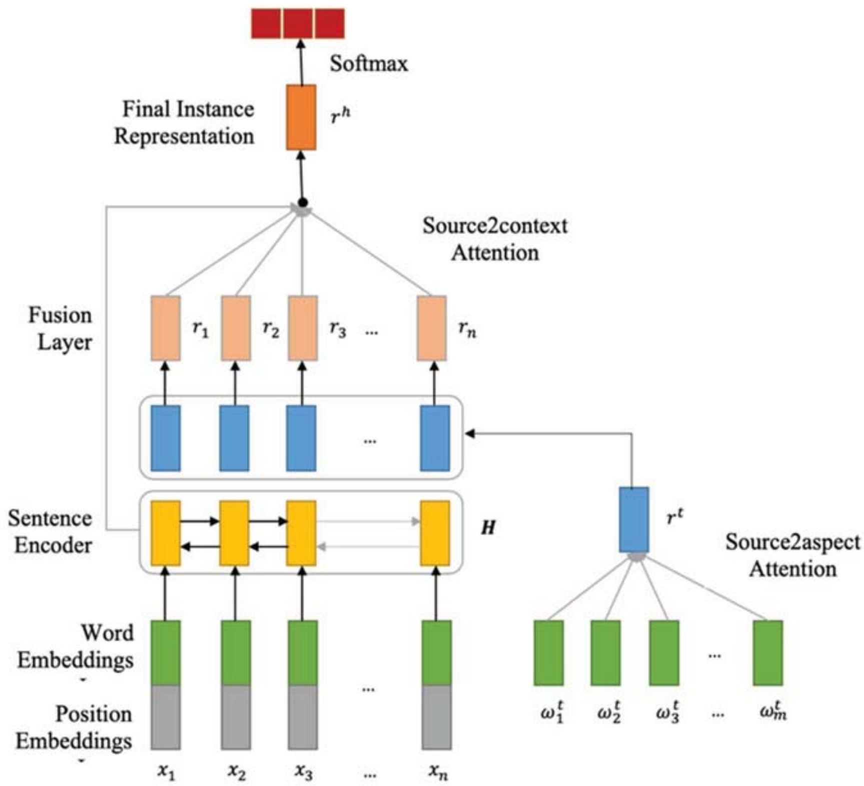

Fig. 1. Architecture of the hierarchical attention position-aware network.

media data could be extracted for polarity of opinions contained in the text, with regards to its highly irregular nature.

III. System Model

A. Architecture

Figure 1 shows the architecture of HAPN, proposed by [5]. It is a deep learning architecture utilizing attention mechanism at different levels and using an embedding to model position. There are four major parts of the architecture, i.e., input embeddings, an encoding layer, a fusion layer, and an output layer.

The four layers can be briefly explained as follows:

1) *Input embeddings:* The embeddings layer consists of a word embeddings layer $w_w \in \mathbb{R}^{d_w \times v_w}$ where d_w is the dimension of the word embeddings and v_w is the vocabulary size, and a position embeddings layer $w_p \in \mathbb{R}^{d_p \times v_p}$, with d_p the dimension of the embeddings and v_p the possible distances between a word and a target. While we can use pre-trained embeddings for words, for position embeddings the values are initially selected at random.

2) *Sentence encoder:* The sentence encoder produces an abstract representation of a sentence based on the word and position embeddings from the preceding layer. It is a bidirectional recurrent layer. [] uses GRU as the underlying network. In addition to that, in this paper we perform experiments with other architectures, i.e. RNN and LSTM.

3) *Fusion layer:* The fusion layer receives the input of the source2aspect attention to capture a relation between specific aspect and target words. The source2context attention is then used for capturing the most indicative words in context, resulting in a weighted embeddings for the input sentence. It produces a final representation of a sentence based on the weighted combination of the hidden state.

4) *Output layer:* Based on the final representation, the output layer predicts the sentiment of the target sentence using a softmax.

B. Dataset

The dataset was extracted from tweets in Indonesian language that focused on political topics. We collect data using keywords related to political figures, political parties and political issues in Indonesia such as "jokowi", "pdip", "ministries", and "election". The total number of tweets we tag for aspects and their sentiments is 3022. Of these, 80% is used as training data, and 20% is used as test data.

TABLE I. A Comparison on Statistics between Our Data (Politic-Twitter) and SemEval 2014 (Laptop Review)

Dataset	Politic-Twitter	SemEval- Laptop
Number of data	3022	1900
Number of aspects	3875	3012
Number of unique aspects	1514	1184
Number of tokens	67087	33924
Number of unique tokens	10102	4533
Average length	22.19	17.85
Token to type ratio	0.927	0.921

Table 1 shows the comparison of data characteristics between the Politic-Twitter dataset and the SemEval Laptop dataset. From the table, we see that the Politic-Twitter data has more than twice as many unique tokens than the SemEval-Laptop data. While the Politic-Twitter dataset is approximately twice as large as the SemEval dataset, if the language used in both datasets has the same level of regularity, we would not expect this difference to be as stark. However, it is well-known that the language in social media is more informal and may contain more variations in spelling or grammatical errors. Secondly, in the Politic-Twitter dataset, there are more unique aspects. Both factors combined, we see that the Politic-Twitter dataset is quite different when compared to the SemEval-Laptop dataset, and we would expect it to be harder. With the publication of this paper, we plan to make our dataset public, since while the domain and the language of our data is specific, we observe that many of its characteristics is possessed by social media data in general, thus it might be useful for researchers especially in the domain of noisy-user generated natural language text.

TABLE II. PARAMETER VALUES IN THE EXPERIMENTS

No	Parameter	Values
1	Word Embeddings + Trainable	Word2Vec, GloVe, and without embeddings (zero), both trainable and with frozen embeddings
2	Model	SimpleRNN, GRU, LSTM
3	Optimizer	Adam, Adadelta, SGD, RMSprop
4	Learning Rate	0.001, 0.002, 0.005, 0.01
5	Dropout	0 – 0.5 in increments of 0.1

C. Experiments

In the experiments, we observe the influence of different values on parameters of the HAPN architecture on its performance on the Politic-Twitter dataset. As a baseline we use the same configuration values as in [5]; that is, the dimension of position embeddings is 50, the sequential parts of the model is implemented as GRUs, and in training Adam is used as optimizer with a learning rate of 0.01, dropout rate 0.5 and with frozen embeddings values. In [5] the embeddings used are the GloVe [6] 300-dimension pretrained word vectors. In our work we also explore the use of an embedding trained on Bahasa Indonesia data and the trainability of embeddings. These embeddings, along with parameter values used in our experiments are listed in Table 2. The code for the experiments is implemented in Keras. The experiment environment is listed in Table 3.

TABLE III. EXPERIMENT ENVIRONTMENT SPECIFICATION

Processor	Intel® Core™ i7-8550U CPU @ 1.80GHz 1.99 GHz
Memory	8192 MB RAM
GPU	Nvidia GeForce MX130
Operating System	Windows 10 Home
System Architecture	64-bit *Operating System*, *x64-based processor*

IV. RESULTS AND DISCUSSION

A. The Influence of Network Parameters

In Table 4, we list the accuracy and the average recall which results from different embeddings, and whether they are trainable. We also show the percentage of coverage of the embeddings, i.e., the percentage of types in the dataset which exists in the pre-trained embeddings.

TABLE IV. INFLUENCE OF DIFFERENT EMBEDDINGS

Word Embedding + Trainable	Word Coverage (%)	Accuracy (%)
Word2Vec Indonesia + True	89.70	70.58
Word2Vec Indonesia + False	89.70	68.39
GloVe + True	48.00	68.26
GloVe + False	48.00	67.74
Zero + True	0	67.87
Zero + False	0	66.32

As expected, the embeddings with the highest word coverage is Word2Vec trained on social media data. Interestingly, the Glove embeddings can cover 48% of types in our Politic-Twitter dataset, despite being trained on English data. Indeed, the phenomenon of code-switching is quite common in Indonesian social media, with people inserting English words into their conversation. Nevertheless, the embeddings trained on Indonesian social media data still performs the best, giving 68.39% accuracy when it is frozen, and 70.58% when it is not. However, when the embeddings are frozen, the difference between the Indonesian and Glove embeddings is not very far, with only 0.65% accuracy difference. It is when the embeddings are trainable that the best performance was produced.

TABLE V. PERFORMANCE ON DIFFERENT SEQUENTIAL ARCHITECTURES

Model NN	Accuracy (%)	AvgRec (%)
LSTM	70.84	62.87
GRU	70.58	56.48
SimpleRNN	69.68	55.89

Table 5 shows the results when different underlying sequential network architecture is used for HAPN. For our data, LSTM [3] works better than GRU as used in the original HAPN, achieving an accuracy of 70.84% and average recall of 62.87%. This indicates that a more complex RNN structure is necessary for the higher irregularity of social media data.

Table 6 shows the results of having different optimizers. We see that Adam works the best by a quite large margin when compared to the other optimizers. Furthermore, in Table 7 we note that with Adam optimizer, having smaller learning rate generally works better, but amongst the values that we experiment with, 0.001 works best. As for dropout, larger probability is consistently better in terms of accuracy, but it is not the case with average recall which drops at 0.4. Both accuracy and average recall are highest when the dropout probability of 0.5 is used. Despite the different datasets, these best values are the ones originally used by [5], thus the higher irregularity of social media data might not be too influential for these parameters.

TABLE VI. PERFORMANCE WITH DIFFERENT OPTIMIZERS

Optimizer	Accuracy (%)	AvgRec (%)
Adam	70.84	62.87
Adadelta	53.03	37.65
SGD	50.19	34.76
RMSprop	69.94	58.62

TABLE VII. PERFORMANCE OF ADAM OPTIMIZER WITH DIFFERENT LEARNING RATES

Learning Rate	Accuracy (%)	AvgRec (%)
0.0005	70.58	59.15
0.001	70.84	62.87
0.002	69.29	55.17
0.005	66.32	53.90
0.01	48.65	33.33

TABLE VIII. PERFORMANCE ON DIFFERENT DROPOUT VALUES

Dropout	Accuracy (%)	AvgRec (%)
0	69.94	56.50
0.1	69.68	61.98
0.2	70.19	62.57
0.3	70.19	62.72
0.4	70.58	57.62
0.5	**70.84**	**62.87**

We show a comparison of the best parameters for the original HAPN for SemEval 2014 aspect-based sentiment analysis data and the best for Politic-Twitter in Table 9. They indicate that for handling social media data, the most influential factors are the use of more complex RNN structures (in our case, LSTM). Having trainable embeddings is also important, especially on domains such as politics which often have its own specific jargons and ways in which words could be interpreted.

TABLE IX. COMPARISON OF THE BEST PARAMETERS FOR SEMEVAL 2014 AND POLITIC-TWITTER WITH THE HAPN ARCHITECTURE

Parameter	SemEval 2014	Politic-Twitter
Word Embedding	Glove	Word2vec Indonesia social media
Trainable	False	True
RNN Model	GRU	LSTM
Optimizer	Adam	Adam
Learning Rate	0.001	0.001
Dropout	0.5	0.5

B. *The Effect of Attention Layer*

Table 10 shows the effect of the attention layer on the predicted results of sentiment level aspects of the HAPN model. Where in each dataset, each word in the data will be converted into an attention value. The standard deviation values and the average of each word attention in one opinion in the dataset will be calculated and then the standard deviation values and the average for all sentences in the dataset will be taken again. It can be seen in table 10, that attention values for successfully predicted data have a standard deviation and an average of standard deviations that are higher than attentional values for data that fail to predict correctly. It can be concluded that the variation of attentiveness will affect the prediction of sentiment. Table 10 also shows the effect of word length in sentences on income attentional value. The smaller the word length is, the greater the standard deviation and the average of the standard deviations.

TABLE X. STATISTICS OF ATTENTION WEIGHTS IN TEST DATA SENTENCES

	Mean of mean	Std of mean	Mean of std	Std of std
All data	0.0660	0.0476	0.0347	0.0384
True Predicted	0.0666	0.0480	0.0371	0.0403
False Predicted	0.0646	0.0468	0.0289	0.0330
Length 0-12	0.1370	0.0499	0.0648	0.0568
Length 12-20	0.0670	0.0098	0.0386	0.0316
Length 20-30.5	0.0413	0.0055	0.0243	0.0208
Length > 30.5	0.0268	0.0032	0.0145	0.0131

V. CONCLUSION

In this work, we have studied the use of hierarchical attention with position embeddings as in the HAPN for aspect-based sentiment analysis on social media data, specifically on the politic domain. We found that for our data which is linguistically highly irregular, using appropriate embeddings is indeed important, and it is necessary that the embeddings are trainable to allow for domain adaptation. LSTM performs better than other, simpler RNN architectures, indicating that a higher level of architecture complexity is necessary. We have also shown sentences with correct prediction results in average have larger variation in attention weights, therefore showing that attention is a useful mechanism in an aspect-based sentiment analysis.

Several venues remain to be explored. Since it has been shown that embeddings play an important role, we would like to experiment with other embeddings, for example FastText, and newer architectures such as BERT [2]. For this, more powerful computational resources will be necessary. Also, social media texts have higher degree of linguistic regularity when compared to more formal-style texts such as in Wikipedia or news articles. Despite the variations and abbreviations, often case people understand what is meant by the author. We believe that it might be because the letters that is contained in a token, together with the context, are enough to convey the information to the reader. Therefore, it might be useful to explore the use of character embeddings, for example BPE [7].

REFERENCES

[1] P. Chen, Z. Sun, L. Bing, and W. Yang, “Recurrent attention network on memory for aspect sentiment analysis,” in *Proceedings of EMNLP*, 2017, pp. 463-472.

[2] J. Devlin, M-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding,” In the *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2019, pp. 4171-4186.

[3] S. Hochreiter and J. Schmidhuber, “Long Short-Term Memory,” *Neural Computation*, 9(8):1735–1780, November 1997.

[4] L. Jiang, M. Yu, M. Zhou, X. Liu, and Tiejun Zhao, “Target-dependent twitter sentiment classification,” in *Proceedings of ACL: Human Language Technologies-Volume1*, 2011, pp. 151-160.

[5] L. Li, Y. Liu, and A. Q. Zhou, “Hierarchical Attention Based Position-aware Network for Aspect-level Sentiment Analysis,” *Proceedings of the 22nd Conference on Computational Natural Language Learning (CoNLL),* 2018.

[6] J. Pennington, R. Socher, and C. D. Manning, “Glove: Global vectors for word representation,” in *Proceedings of EMNLP*, 2014, pp. 1532-1543.

[7] R. Sennrich, B. Haddow, and A. Birch, "Neural Machine Translation of Rare Words with Subword Units," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2016, pp. 1715-1725.