Improving Relation-Aware Aspect-Based Sentiment Analysis with Semi-Supervised Learning and Adaptation to Korean Dataset

Huijae Kim[1], **Hyeon Pyo**[2], **Jongmyeong Kim**[2], **Soyeon Jeon**[3]
Department of Industrial & Systems Engineering, KAIST, Korea[1],
School of Computing, KAIST, Korea[2],
Department of Political Science and International Relations, Seoul National University, Korea[3],
{heejae6021, megatwins, grayapple, syjeon821}@kaist.ac.kr

Abstract

Sentiment have been a subject that many researchers seek to extract through text analysis. Through this project, firstly, we replicate Relation-Aware Collaborative Learning (RACL) (Chen and Qian, 2020), which recently showed good performance in the sentiment analysis task, and confirm whether actual learning and prediction work well. Secondly, we present the problem that too much tagged information is needed to learn RACL, and apply semi-supervised learning as a way to solve this problem. Finally, we conduct experiments using Korean cosmetic review data to check whether RACL with semi-supervised learning can perform well in Korean. We find that the RACL with semi-supervised learning performs better than the RACL model itself and show the good performance of Korean ABSA model with RACL. Finally, the effects of hyper-parameters are tested and the best hyper-parameter value is represented.

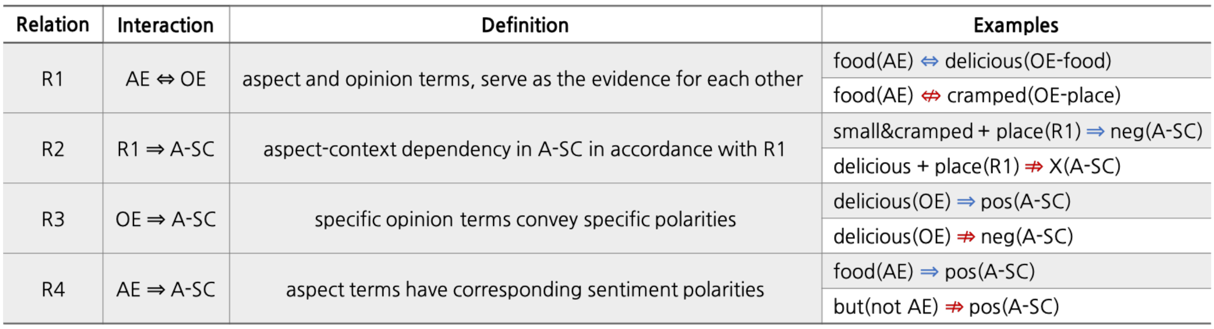
1 Introduction

We aim to apply Aspect-based Sentiment Analysis (ABSA) to Korean data and to incorporate semi-supervised learning techniques to a previous work, Relation-Aware Collaborative Learning (RACL) (Chen and Qian, 2020).

Earlier sentiment analysis techniques often rely on structurally defined units of analysis. This means that their outcome is defined for structural tokens of the text, such as a word or a sentence. However, it is many times the case that one sentence conveys multiple sentiments, or at the other extreme, no sentiment is conveyed by even a whole paragraph. It can therefore be said that structurally defined units of analysis do not truly reflect the units of sentiment as carried by the text. For example, the restaurant review, 'The place is small and cramped but the food is delicious' has two topics: place and food. Sentiment analyses depending on structurally defined units of analysis (in this case, a sentence) would annotate this sentence as contrasting or neutral since 'place' is related to negative sentiments while 'food' is related to positive sentiments.

Newer methods collectively known as ABSA can resolve such a discrepancy between units of analysis and units of sentiment. Whereas non-ABSA techniques suffered because their units of analysis (*e.g.*, a sentence) often do not match the actual units of sentiment (*e.g.*, two sentiments in one sentence), ABSA techniques utilize a more abstractly defined unit of analysis - that is, the aspect. An aspect, unlike a sentence, is free from structural restrictions. For example, an aspect does not have to contain a complete set of subject and predicate. An aspect is a structure-free unit of analysis that could be of arbitrary length, as long as it corresponds to a single sentiment. In theory, one aspect could thus be an entire paragraph long or even a just a morpheme long, if that is the sentiment it corresponds to. This definition of the unit of analysis in ABSA makes it convenient because the unit of analysis (aspect) now automatically and necessarily matches the unit of sentiment: one aspect is defined for one sentiment. Continuing from the same restaurant review example, the review undergoing an ABSA method would output two units of analysis (*i.e.*, two aspects: food and place) and their respective two units of sentiment (*i.e.*, two sentiments: positive for food and negative for place).

ABSA involves completing three subtasks: aspect term extraction (AE), opinion term extraction (OE), and aspect-based sentiment classification (A-SC). Throughout this paper, the shorthand notations are abused to refer to both each subtask or the corresponding module that completes the subtask, depending on the context. AE distinguishes various topics. More specifically, the minimal unit



Relation	Interaction	Definition	Examples
R1	AE ⇔ OE	aspect and opinion terms, serve as the evidence for each other	food(AE) ⇔ delicious(OE-food)
			food(AE) ⇎ cramped(OE-place)
R2	R1 ⇒ A-SC	aspect-context dependency in A-SC in accordance with R1	small&cramped + place(R1) ⇒ neg(A-SC)
			delicious + place(R1) ⇏ X(A-SC)
R3	OE ⇒ A-SC	specific opinion terms convey specific polarities	delicious(OE) ⇒ pos(A-SC)
			delicious(OE) ⇏ neg(A-SC)
R4	AE ⇒ A-SC	aspect terms have corresponding sentiment polarities	food(AE) ⇒ pos(A-SC)
			but(not AE) ⇏ pos(A-SC)

Figure 1: Description of relations

that contains meaning such as words in English or morpheme in Korean will be extracted from the data. They are clustered considering the semantic similarity (*e.g.*, cosine similarity). Then the clusters are assigned aspects considering the commonality within the cluster. Then using the lexicon for negative and positive word dictionaries, and part-of-speech tagging, sentiments of each aspect (A-SC) are classified.

The methods of ABSA are classified into two categories: separate methods and unified methods. Separate methods (*e.g.*, CMLA (Wang et al., 2017), DEDNN (Xu et al., 2018)) have two different frameworks for AE and A-SC respectively, and results from AE and A-SC must be combined together in a pipeline for a complete ABSA task. In this process, the relations between AE and A-SC are not fully considered. On the other hand, unified methods (*e.g.*, T-Net (Li et al., 2018), TransCap (Chen and Qian, 2019)) deal with AE and A-SC in the same framework and extract both results from the framework. The paper our team selected, RACL (Chen and Qian, 2020), improves the performance of ABSA. It considers more interactive relations between the subtasks using multi-task learning and relation propagation mechanisms which are stacked multiple layers.

Two major shortcomings compromise the effectiveness of separate methods. One is that their inherent pipeline structure neglects any useful clues each subtask could provide for other subtasks. For instance, once the opinion term ‘delicious’ is identified, one strong candidate aspect term would be ‘food’. This AE-OE relation must therefore be considered. Here is another example, this time one between the AE-OE pair and A-SC. If the aspect-opinion pair ‘small place’ is identified, then the A-SC must jointly consider the aspect-opinion term because ‘small’ could be a positive attribute for aspects such as ‘mistakes’. The interplay between AE, OE, and A-SC must therefore fully exploit the inter-subtask clues derivable from the text. The other shortcoming is that the pipeline-like structure of separate methods makes them vulnerable to error propagation. If there are errors in AE/OE, which are often the earlier subtasks in separate methods, then likely the succeeding subtask of A-SC cannot be trusted to render very promising results. In other words, AE, OE, and A-SC must simultaneously learn from the text and from each other without a one-way stream of potential error propagation.

AE, OE, and A-SC should each be annotated to analyze data using ABSA. However, labeling all the data requires a skilled human agent to put in their time and effort. To deal with this situation, our team incorporate semi-supervised learning in finding a way to utilize unlabeled data with relatively low cost. Semi-supervised learning is an approach to machine learning which uses a small amount of labeled data with a large amount of unlabeled data during training. Furthermore, already existing Korean data was insufficiently annotated for ABSA purposes. Should incorporating semi-supervised learning work in analyzing ABSA, other Korean datasets which were previously not opt for ABSA purposes will be transferable. Our team’s objective is to utilize unlabeled data 10% of the data are labeled and manually checked and semi-supervised learning deals with the remaining data.

2 Approach

To solve the problem of constructing real-world datasets, we apply a semi-supervised learning method to the ABSA model. In semi-supervised classification, many methods have been studied, such as self-training, co-training, perturbation-based methods, manifolds and so on (Van Engelen and Hoos, 2020). Among those methods, self-training is a method to train a model by labeled data, predict unlabeled data using the model, and re-train the model with both the labeled and unlabeled data. Since it has the simple algorithm to apply into any model, we use self-training as our semi-supervised learning approach.

To improve the performance of self-training, we refer Noisy Student Training which achieved the state-of-the-art on ImageNet in 2019 (Xie et al., 2020). The algorithm of Noisy Student Training is shown as follows in Algorithm 1.

Algorithm 1 Noisy Student Training's algorithm

```
Train a teach model with labeled data
while $k \leq$ number of iteration do
    Infer pseudo-labels on unlabeled data
    Train a student model with combined data and noise added
    Make the student a new teacher
    $k \leftarrow k + 1$
```

Noisy Student Training improved the performance of self-training since it develops better student models than the previous teacher model by injecting noise and iterating the training process. The paper suggested data augmentation (Cubuk et al., 2020), dropout (Srivastava et al., 2014), and stochastic depth (Huang et al., 2016) for noises, but the paper is for image-related tasks not text-related. Since we have text review data for our input, we need to replace the methods. We use Random Swap, one of the Easy Data Augmentation techniques for text classification, which is randomly choosing two words in a sentence and swap their positions (Wei and Zou, 2019). Therefore, the algorithm of our modified Noisy Student Training for text is illustrated in Figure 2.

Since the RACL is a multi-task learning model consisting of 3 subtasks: AE, OE, and A-SC, the labeled data $\{(S_1, Y_1^A, Y_1^O, Y_1^S), ..., (S_n, Y_n^A, Y_n^O, Y_n^S)\}$ and the unlabeled data $\{\tilde{S_1}, \tilde{S_2}, ..., \tilde{S_m}\}$ are required for our Noisy Student Training, where a sentence $S_i \in R^e$, a tag sequence of aspects $Y_i^A \in R^e$, a tag sequence of opinion $Y_i^O \in R^e$, and a tag sequence of sentiment classification $Y_i^S \in R^e$, given the maximum length of a sentence e. Given the labeled data, we train a teacher model. Using the model, we generate pseudo labels $\{(\tilde{Y}_1^A, \tilde{Y}_1^O, \tilde{Y}_1^S), ..., (\tilde{Y}_m^A, \tilde{Y}_m^O, \tilde{Y}_m^S)\}$ for the unlabeled data. Then, we add noises to the unlabeled data with pseudo labels by Random Swap with a hyper-parameter of ratio, which represents how many words in a sentence be swapped. Random Swap makes the student model ensure prediction consistency across diverse versions of a sentence. The student is trained to be able to predict correct labels with modified sentences and this improves the performance of the model. And then, we form a new train dataset with the original labeled data and the pseudo-labeled data with noises and train a student model. These steps are repeated the number of students times which is a hyperparamter to be set. Finally, we obtain the final model from the last student model.

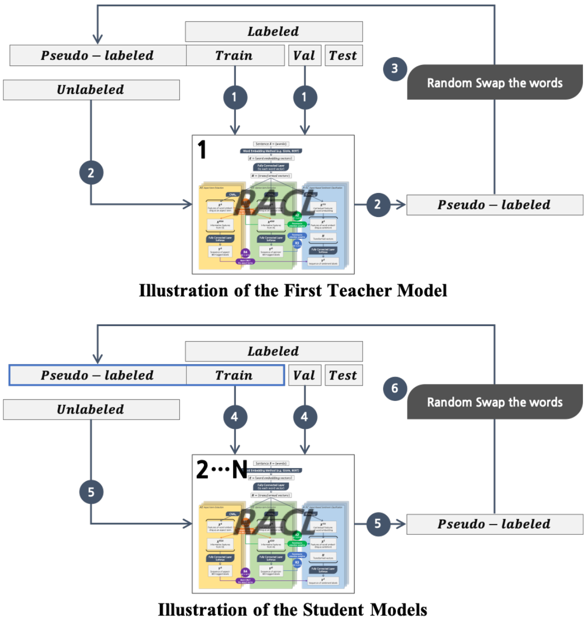


Illustration of the First Teacher Model

Illustration of the Student Models

Figure 2: Illustration of the process of our Noisy Student Training for text

3 Experiments

3.1 Replication & Improvement

Datasets Customer review data of restaurant and laptop from SemEval 2014(Pontiki et al., 2014) and 2015(Pontiki et al., 2015) annotated for 3 subtasks has been used in the original RACL. 2 Subtasks AE and OE, each token is tagged by B(Beginning of a term)-I(Inside a term)-O(Outside of a term) tag scheme. Sentiment corresponding to aspect is labeled positive, neutral or negative. For replication and improvement, we use only the restaurant review data of 2014(Restaurant14). Moreover, to construct labeled and unlabeled dataset for semi-supervised learning, we set 1800 of 2435 training data of Restaurant14 to unlabeled data by removing the label tags, and use the left training data as the

Dataset	Type		Sentence
Restaurant14 (English)	train	labeled	635
		unlabeled	1800
	val		609
	test		800
Cosmetics19 (Korean)	train	labeled	1281
		unlabeled	1000
	val		319
	test		399

Table 1: The statistics of datasets

Environment	Original	Our	
	1080Ti	A100	V100
Python	3.6.10	3.6.10	3.7.2
Tensorflow	1.5.0	1.15.2	1.13.1
Numpy	1.16.4	1.17.3	1.16.2
Scikit-learn	0.22.2	0.23.1	0.22.1

Table 2: Comparison of implementation environments

training labeled data.

Settings We replicate RACL in the environment described as below Table 2. We try to establish the same environment as the experimental environment of the paper. Hyper-parameters such as batch size, number of filters, dropout probability, normalization scale, and learning rate used the same values as in the paper, and various changes are made to the kernel size K and the number of layers L to check the correlation with performance.

For the improvement part of the model, we need more hyper-parameters such as the ratio of Random Swap. We set the ratio of Random Swap to $\{0, 0.1, 0.2\}$. To test semi-supervised learning's optimal label/unlabel ratio and robustness on small datasets, size of labeled data and unlabeled data are set to $\{100, 200, 600\}$ and $\{100, 900, 1800\}$

Same with the RACL paper, the metrics that we use are F_1 scores of each subtask: aspect term extraction, opinion term extraction and aspect-based sentiment classification (AE-F_1, OE-F_1, SC-F_1). For overall performance, we also consider ABSA-F_1 as the result for an aspect term when it would be considered as correct when both AE and SC results are correct.

We run the experiments in DGX3-A100 and Tesla V100 from KISTI. In the environment of DGX3-A100, Glove-based RACL with semi-supervised learning takes about 5 seconds per epoch on average.

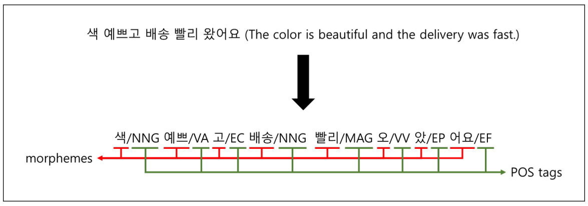

색 예쁘고 배송 빨리 왔어요 (The color is beautiful and the delivery was fast.)
색/NNG 예쁘/VA 고/EC 배송/NNG 빨리/MAG 오/VV 았/EP 어요/EF
morphemes
POS tags

Figure 3: Example of sentence expression using morphemes and POS tags

3.2 Korean ABSA

Datasets To test the applicability of RACL to Korean, we construct Korean ABSA data. First, we secure about 23,000 cosmetic review data from Data Repository for Business Research, The Korean Academic Society of Business Administration(Shin, 2019). Although the dataset includes only one aspect term and one polarity for a sentence, the data we need should contain multiple aspect terms, opinion terms and polarities if a sentence has many aspect terms. Therefore, we manually tag aspect terms and opinion terms based on the BIO scheme and polarities for about 2000 sentences. We only consider positive and negative in polarity, while the RACL paper considered positive, negative and neutral. We use the tagged data as training, validation and test set for RACL, and the 1000 unlabeled data from the remaining data for semi-supervised learning.

Settings In order to reflect the characteristics of Korean are verbs and adjectives are conjugated, and the proposition has a grammatical role in the sentence, we use morphemes as unit tokens of training. We represent a sentence as a sequence of morphemes using Korean Morpheme Analyzer(KMA) provided by SK Telecom. This KMA expresses the role of morphemes in sentences using Sejong part-of-speech(POS) tags. The example of sentence expression using morphemes and POS tags is shown in Figure 3.

We use only GloVe embedding (Pennington et al., 2014) to use sentences expressed with morpheme and part-of-speech tags as training data for quick experiments, while the original RACL model used both GloVe and FastText (Joulin et al., 2016) or BERT (Devlin et al., 2019). The embedding weights used in the model is trained by the corpus combining Korean Wikipedia data with the cosmetic review data. The Wikipedia is written by various people. This means that it well reflects the language usage of ordinary people, so it is often

Dataset	Model	AE-F_1	OE-F_1	SC-F_1	ABSA-F_1
Restaurant14(2436)	RACL-Glove(paper)	85.37	85.32	74.46	70.67
	RACL-Glove(replication)	85.40	85.50	73.20	69.80
Restaurant14(600)	RACL-Glove	81.51	81.28	71.73	65.50
	RACL-Glove+SSL	82.64	82.85	74.20	68.45
Cosmetics19	RACL-Glove	73.22	89.26	49.08	65.45
	RACL-Glove+SSL	76.42	97.23	48.29	69.29

Table 3: Comparison of original, replicated and modified RACL model.

Unlabeled/Labeled	100	200	600
0 (fully-supervised)	43.15	55.76	65.56
100	48.94(5.79)	58.85(+3.09)	66.43 (+0.87)
900	47.16(+4.01)	59.81(+4.05)	67.53 (+1.97)
1,800	46.37(+3.22)	57.00(+1.24)	67.10 (+1.54)

Table 4: ABSA-F_1 of RACL+SSL with different size of unlabeled and labeled data

used as learning data for embeddings.

For Korean ABSA task, learning rate is tuned to 0.0005 based on the validation loss convergence in epoch-to-loss plots. Number of full and warm-up iterations are set to 90 and 50 to enhance the capability to capture best model for data in new language. Other model hyper-parameters including batch size are same with the replication. The kernel size K and the number of layers L are also tuned to optimize the model's capability on Korean sentences.

We run the experiments for Korean ABSA task in the same environments described in Section 3.1.

4 Results

4.1 Comparison Results

The comparison results of the baseline and the improved RACL model are shown in Table 3. SSL denotes semi-supervised learning with unlabeled data. The full and reduced dataset are demonstrated as Res14(2436) and Res14(600) where the number of labeled data in written in the brackets.

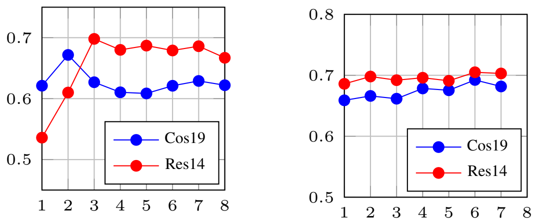
F_1 scores of replicated model on the Restaurant14 are almost same with the original. Slight difference in original replicated results is due to to random initialization and different package versions in implementation.

In fully supervised RACL-Glove, ABSA-F_1 score shows drop of 5.17 when data size is reduced to 600 from 2436. This drop is restricted to 1.35 with SSL. Especially, SC-F_1 is kept almost constant to the original result with full data. This proves robustness of our semi-supervised learning strategy. More details of the improved results are discussed in the following section.

4.2 Effects of Semi-supervised Learning Hyper-parameters

In every experiments in Table 4, semi-supervised learning achieves higher ABSA-F_1 scores than those of the fully-supervised learning with the same number of labeled data. RACL with SSL is more robust in the smaller size of unlabeled data than using the full unlabeled data of size 1800. However, reducing the size of unlabeled data to 100 does not fully utilize the power of the semi-supervised learning strategy. Performance is still boosted when the unlabeled data size reduces to 100 in training with an extremely small(100) size of labeled data. It implies that not only the absolute sizes but the relative sizes of labeled and unlabeled data matter. Our experiment narrows the range of the possible optimal ratio of unlabeled data to labeled data as 1 to 4.5. Since both the absolute size of labeled data and the ratio of unlabeled data to labeled data affect the performance of semi-supervised learning, we suggest fine-tuning the size of unlabeled data starting from the large possible range. Moreover, SSL is especially more effective at conserving the model capability when there are few labeled data.

Confidence sorting increases the model performance further. Before adding the data with pseudo labels to the new training dataset, the RACL+SSL model sorts the sentences based on the evaluated confidence of pseudo-labels. Top n most confident sentences are selected and added to the training set. The number of chosen examples n is con-



(a) Effects of K

(b) Effects of L

Figure 4: Effects of model hyper-parameters

trolled to 0(fully-supervised learning), 100, 900, and 1800. Overall, the performance is enhanced most by choosing the top 100 confident pseudo-labels from the pool of 1800. The increase shrinks as the number of pseudo labels increases to 900 and 1800. Results with 100 labeled data are unstable as there are too few initial labels. This shows a limitation of self-training that the effect of wrong pseudo-labels could be amplified. However, self-training still clearly boosts its performance. SSL is especially more powerful when there are little labeled data.

The ratio of Random Swap is controlled to 0.1, 0.2 and is compared with baseline shown in Table 5. The experiment was conducted with 0.5 of the ratio of unlabeled data to labeled data. Performance boosts with the ratio of 0.1. It seems to be the optimal ratio that successfully augment the data while preserving the semantics of Korean sentences.

4.3 Korean ABSA

ABSA-F$_1$ score is increased by applying semi-supervised learning. OE-F$_1$ score for the Cosmetics19 is relatively higher than the Restaurant14 data. Opinion terms in the Cosmetics19 dataset lack diversity. The total set of opinion terms is really small and this may result in a high OE-F$_1$ score close to 100. On the other hand, SC-F$_1$ is lower than 50 in the data. A case study of sentences in the data reveals that certain opinion terms imply different sentiments when used in different contexts. The limitation stems from both natures of cosmetics review sentences and GloVe embedding that cannot capture the dual semantics of a single token. We expect that it could be improved with further work with encoders such as BERT.

4.4 Effects of Model Hyper-parameters

In the original paper, it was proposed that two hyper-parameters K(the kernel size of the CNN encoder) and L(the number of stacked RACL layers) decide the learning capability and performance of the model.

As Figure 4 shows, the model with $K = 3$ yields the best ABSA-F$_1$ for Restaurant14(English), and performance decreases as K increases over 3. This replication result is aligned with the result in the original paper. The effect of kernel size overall is similar in Cosmetics19(Korean) except for the performance peak in $K = 2$.

For all datasets, performance roughly increases with more layers until it reaches the optimum. In Cosmetics19, the model with $L = 6$ shows significantly good performance. The layer number greater than 6 drops the ABSA-F$_1$ score, which implies the model is over-complicated.

5 Discussion

Our team grafts semi-supervised learning to the ABSA model based on RACL and adapts the model to Korean cosmetic dataset. For Korean dataset, our team uses GloVe embedding and manually tag labels of about 2000 sentences for training. Our team refers to Noisy Student Training, a self-training technique for semi-supervised learning, and adds Random Swap to add noises and confidence-based masking to improve the self-training.

Results of the experiment follow: First, RACL with semi-supervised learning performs better than the RACL model itself. In detail, RACL with semi-supervised learning is more effective when there are few labeled data. Second, the RACL model with Korean dataset shows good performance, especially in opinion extraction. Third, the effects of the labeled and unlabeled data size and Random Swap for data augmentation are tested. The ratio of unlabeled data to labeled data is controlled in range of 1 4.5 and Random Swap with ratio 0.1 achieves significant increase in performance.

Limitations of our experiment follow: Word equivocality can explain the low performance in A-SC. Some words such as ‘big time’ or ’dang’ can be used to express sometimes positive sentiments but other times negative sentiments. Furthermore, when this is mixed with the case where the same cosmetic features arise different sentiment to the reviewers, it was hard to annotate the sentiment. For instance, some customers liked the color pink in the tint while others did not. When it comes to the review, ’the color is pink, dang!’, our team is unsure whether the sentiment was positive or negative.

Ratio	AE-F_1	OE-F_1	SC-F_1	ABSA-F_1
0	73.02	97.98	42.46	66.45
0.1	73.58	92.74	50.83	68.35
0.2	72.76	97.53	44.62	64.67

Table 5: Effect of Ratio of Random Swap

A few areas need to be considered in further studies, although this research has taken a step in the direction of developing the Aspect-Based Sentiment Analysis model with semi-supervised learning. One area could be comparing the effect of Random Swap with Korean and English data since this swapping can grasp the syntax-wise difference between those two languages. In addition, to deal with the word equivocality, further research can consider applying different embedding methods(*e.g.*, BERT) to grasp more contexts into the analysis.

References

Zhuang Chen and Tieyun Qian. 2019. Transfer capsule network for aspect level sentiment classification. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 547–556.

Zhuang Chen and Tieyun Qian. 2020. Relation-aware collaborative learning for unified aspect-based sentiment analysis. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3685–3694.

Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. 2020. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 702–703.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *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)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Gao Huang, Yu Sun, Zhuang Liu, Daniel Sedra, and Kilian Q Weinberger. 2016. Deep networks with stochastic depth. In *European conference on computer vision*, pages 646–661. Springer.

Armand Joulin, Edouard Grave, Piotr Bojanowski, Matthijs Douze, Hérve Jégou, and Tomas Mikolov. 2016. Fasttext. zip: Compressing text classification models. *arXiv preprint arXiv:1612.03651*.

Xin Li, Lidong Bing, Wai Lam, and Bei Shi. 2018. Transformation networks for target-oriented sentiment classification. *arXiv preprint arXiv:1805.01086*.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Maria Pontiki, Dimitrios Galanis, Harris Papageorgiou, Suresh Manandhar, and Ion Androutsopoulos. 2015. Semeval-2015 task 12: Aspect based sentiment analysis. In *Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015)*, pages 486–495.

Maria Pontiki, Dimitrios Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. Semeval-2014 task 4: Aspect based sentiment analysis. In *Proceedings of the 8th international workshop on semantic evaluation (SemEval 2015)*, pages 27–35.

K-S Shin. 2019. Data repository for business research.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958.

Jesper E Van Engelen and Holger H Hoos. 2020. A survey on semi-supervised learning. *Machine Learning*, 109(2):373–440.

Wenya Wang, Sinno Jialin Pan, Daniel Dahlmeier, and Xiaokui Xiao. 2017. Coupled multi-layer attentions for co-extraction of aspect and opinion terms. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31.

Jason Wei and Kai Zou. 2019. Eda: Easy data augmentation techniques for boosting performance on text classification tasks. *arXiv preprint arXiv:1901.11196*.

Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. 2020. Self-training with noisy student improves imagenet classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10687–10698.

Hu Xu, Bing Liu, Lei Shu, and Philip S Yu. 2018. Double embeddings and cnn-based sequence labeling for aspect extraction. *arXiv preprint arXiv:1805.04601*.