HOW DOES BERT ADDRESS POLYSEMY OF KOREAN ADVERBIAL POSTPOSITIONS *-ey*, *-eyse*, AND *-(u)lo*?

Anonymous authors
Paper under double-blind review

ABSTRACT

The present study reports computational accounts of resolving word-level polysemy in a lesser-studied language—Korean. Postpositions, which are characterized as multiple form-function mapping and thus polysemous, pose a challenge to automatic analysis and model performance in identifying their functions. In this study, we devised a classification model by employing BERT and introduces a computational simulation that interactively demonstrates how a BERT model simulates human interpretation of word-level polysemy involving Korean adverbial postpositions *-ey*, *-eyse*, and *-(u)lo*. Results reveal that (i) there is an inverse relationship between the classification accuracy and the number of functions that each postposition manifests, (ii) the model performance is affected by the corpus size of each function, and (iii) the performance gradually improves as the epoch proceeds.

1 INTRODUCTION

Polysemy, one type of ambiguity, occurs when one form delivers multiple, and yet related, meanings/functions and vice versa (Glynn & Robinson, 2014). Traditional word-embedding models showed an unsatisfactory level of performance in polysemy interpretation. This is due to the technical nature of these models: they are *static* in that a single vector is assigned to each word (Desagulier, 2019; Ethayarajh, 2019; Liu et al., 2019a). To overcome this issue, recent studies have proposed a *contextualized* word-embedding model which considers neighborhood information about a polysemous word on the basis of sequences of words around the target word. Various models have been suggested for this task, such as Embeddings from Language Models (Peters et al., 2018), Generative Pre-Training (Radford et al., 2018), and Bidirectional Encoder Representations from Transformer (BERT; Devlin et al., 2018). Among these models, BERT shows the best performance in many language tasks such as translation, classification, and question-answering (e.g., Devlin et al., 2018; Tang et al., 2019).

Despite a good deal of research on BERT in English, very few studies have investigated BERT-based polysemy interpretation in languages that are typologically different from English. We turn our attention to Korean, an agglutinative Subject–Object–Verb language in which multiple postpositions or affixes with dedicated forms and meanings are attached to the stem of nominals or predicates. A postposition is a function word providing grammatical information to words it is attached (Sohn, 1999). It normally involves many-to-many associations between form and function; as such, a postposition is polysemous (Choo & Kwak, 2008).

Several studies have used word-embedding models to capture and tease apart the different meanings/functions of Korean postpositions (e.g., Bae et al., 2014; 2015; Kim & Ock, 2016; Lee et al., 2015; Mun & Shin, 2020; Shin et al., 2005). However, the model performance reported in the previous studies is unsatisfactory, with the accuracy ranging from 0.621 (Bae et al., 2014) to 0.837 (Kim & Ock, 2016). One possible reason for this unsatisfactory performance is that they did not consider contextual information. Against this background, the current study employs BERT for the same kind of classification task for Korean postpositions. BERT produces contextual embeddings, and this characteristic may help us to create a better classification system for postpositions. Still unclear is the particular reason for BERT's superior performance over the others. In order to further understand how BERT recognizes the word-level polysemy, we propose a BERT-based visualization system in addressing polysemy interpretation of three adverbial postpositions, *-ey*, *-eyse*, and *-(u)lo*, which are frequently used and documented in the previous studies (e.g., Cho & Kim, 1996; Jeong, 2010; Nam, 1993; Park, 1999; Song, 2014).

2 KOREAN ADVERBIAL POSTPOSITIONS: *-ey*, *-eyse*, AND *-(u)lo*

In order to determine the number of functions of each postposition, this study considers the major functions of these postpositions which are frequently attested in the Sejong dictionary: eight for *-ey*, two for *-eyse*, and six for *-(u)lo* (Shin, 2008). *-ey* involves the following functions: agent (AGT), criterion (CRT), effector (EFF), final state (FNS), goal (GOL), instrument (INS), location (LOC), and theme (THM).

(1) *-ey* as AGT (agent)

가두	진출이	경찰에	저지되었다.
katwu	cinchwul-i	kyengchal-ey	ceci-toy-ess-ta.
street	go.out-NOM	police-AGT	stop-PSV-PST-DECL

'By going out to the street was stopped by the police.'

(2) *-ey* as CRT (criterion)

영호는	20만원에	모니터를	낙찰했다.
Yenghuy-nun	20manwen-ey	monithe-lul	nakchalhay-ss-ta.
Yenghuy-TOP	200,000 won-CRT	moniter-ACC	sell-PST-DECL

'Yenghuy sold the monitor (to a bidder) for 200,000 won.'

(3) *-ey* as EFF (effector)

문들이	거센	바람에	모두	건들댄다.
mwun-tul-i	keseyn	palam-ey	motwu	kentultay-n-ta.
door-PL-NOM	strong	wind-EFF	all	sway-PRS-DECL

'The doors all sway by the strong wind.'

(4) *-ey* as FNS (final state)

김교수는	조교에	박군을	추천했다.
kimkyoswu-nun	cokyo-ey	park-kwun-ul	chwuchenhay-ss-ta.
professor.Kim-TOP	assistant-FNS	Park-Mr-ACC	recommend-PST-DECL

'Professor Kim recommended Park as an assistant.'

(5) *-ey* as GOL (goal)

철수가	던진	칼이	땅바닥에	내리꽂혔다.
Chelswu-ka	tenc-i-n	khal-i	ttangpatak-ey	naylyekkoc-hi-ess-ta.
Chelswu-TOP	throw-CST-PRS	knife-NOM	ground-GOL	stick-PSV-PST-DECL

'The knife thrown by Chelswu stuck to the ground.'

(6) *-ey* as INS (instrument)

그	어린	소년은	화롯불에	손을	녹이고	있었다.
ku	eli-n	sonye-nun	hwalospwul-ey	son-ul	nok-i-ko	iss-ess-ta.
That	young-REL	boy-TOP	fire-INS	hand-ACC	melt-CST-and	be-PST-DECL

'The young boy was using the fire to warm his hands.'

(7) *-ey* as LOC (location)

그는	온종일	서재에	파묻혀	지낸다.
ku-nun	oncongil	secay-ey	phamwut-hi-e	cinay-n-ta.
He-TOP	all day	study.room-LOC	bury.in-PSV-PRS	be-PRS-DECL

'He is buried in his study room all day.'

(8) *-ey* as THM (theme)

현대인들은	모두	참된	지식에	허기져있다.
hyentayin-tul-un	motwu	chamtoy-n	cisik-ey	hekicye-iss-ta.
modern.people-PL-TOP	all	true-REL	knowledge-THM	hungry-PRS-DECL

'All modern people are hungry for true knowledge.'

-eyse has only two functions, location (LOC) (9) and source (SRC) (10). *-eyse* manifests fewer functions than *-ey* (Choo & Kwak, 2008). However, frequency is equally high compared to that of *-ey* (e.g., Cho & Kim, 1996; Song, 2014).

(9) *-eyse* as LOC (location)

철수는	서울에서	태어났다.
Chelswu-nun	sewul-eyse	thayena-ss-ta.
Chelswu-TOP	seoul-LOC	born-PST-DECL

'Chelswu was born in Seoul.'

(10) *-eyse* as SRC (source)

광부들이	바다에서	석유를	뽑아올린다.
kwangpwutul-i	pata-eyse	sekyu-lul	ppopaoll-i-n-ta.
miner-PL-NOM	sea-SRC	oil-ACC	pull-CST-PRS-DECL

'Miners pull oil from the sea.'

-(u)lo engages in six functions: criterion (CRT), direction (DIR), effector (EFF), final state (FNS), instrument (INS), and location (LOC) (Shin, 2008).

(11) *-(u)lo* as CRT (criterion)

적당한	시간	간격으로	배차되었다.
cektangha-n	sikan	kankyek-ulo	paycha-toy-ess-ta.
appropriate-REL	time	interval-CRT	arrange-PSV-PST-DECL

'It was arranged at appropriate time intervals.'

(12) *-(u)lo* as DIR (direction)

범인은	어두운	골목으로	달아났다.
pemin-un	etwuwun	kolmok-ulo	talana-ss-ta.
criminal-NOM	dark	alley-DIR	flee-PST-DECL

'The criminal fled into a dark alley.'

(13) -*(u)lo* as EFF (effector)

환자가	위암으로	매우	괴로워하고	있습니다.
hwanca-ka	wiam-ulo	maywu	koyloweha-ko	iss-supni-ta.
patient-NOM	stomach.cancer-EFF	very	suffer-and	be-HON-DECL

'The patient is suffering greatly due to stomach cancer.'

(14) -*(u)lo* as FNS (final state)

그는	대표	강사로	초빙되었다.
ku-nun	tayphyo	kangsa-lo	choping-toy-ess-ta.
He-TOP	representative	lecturer-FNS	invite-PSV-PST-DECL

'He was invited as a representative lecturer.'

(15) -*(u)lo* as INS (instrument)

전선이	연줄로	감겼다.
censen-i	yencwul-lo	kam-ki-ess-ta.
wire-NOM	connection.wire-INS	wind-PSV-PST-DECL

'The wire wound around with the connection wire.'

(16) -*(u)lo* as LOC (location)

경찰이	피해자를	검찰로	압송했다.
kyengchal-i	phiuyca-lul	kemchal-lo	apsonghay-ss-ta.
police-NOM	suspect-ACC	prosecution-LOC	transport.do-PST-DECL

'The police transported the suspect to the prosecution.'

3 METHODS

3.1 CREATING INPUT CORPUS

The Sejong primary corpus[1], the representative corpus in Korean, does not code the information about the functions of postpositions directly in each sentence (which is necessary for model training). We thus annotated a portion of the original corpus data manually. For this purpose, we extracted sentences that have only one postposition and predicate. This also allowed us to control for additional confounding factors which might have interfered with the performance of our model. We then extracted 5,000 sentences randomly for each postposition from the initial dataset.

-ey		***-eyse***		***-(u)lo***	
Function	Frequency	Function	Frequency	Function	Frequency
LOC	1,780	LOC	4,206	FNS	1,681
CRT	1,516	SRC	647	DIR	1,449
THM	448			INS	739
GOL	441			CRT	593
FNS	216			LOC	158
EFF	198			EFF	88
INS	69				
AGT	47				
Total	4,715	Total	4,853	Total	4,708

Table 1: By-function frequency list of *-ey*, *-eyse*, and *-(u)lo* in cross-validated corpus.

[1]Sejong corpus is available at:https://www.korean.go.kr

Three native speakers of Korean annotated each postposition for its function in this 15,000-sentence corpus. Fleiss' kappa scores showed that the annotators' outcomes were almost identical: 0.948 (-*ey*), 0.928 (-*eyse*), and 0.947 (-*(u)lo*). We further excluded instances which showed disagreement among the annotators. The final corpus consisted of 4,715 sentences for -*ey*, 4,853 sentences for -*eyse*, and 4,708 sentences for -*(u)lo*. Table 1 presents the detailed by-function frequency list of the three postpositions [2].

3.2 Creating training and test sets

We pre-processed the data in consideration of how BERT works (we used the original BERT model for this task). First, we added [CLS] ('classification'; indicating the start of a sentence) before a sentence and [SEP] ('separation'; indicating the end of a sentence) after a sentence to indicate where the sentence starts and ends. These indicators made it possible for the BERT model to recognize a sentence boundary in a text, allowing the model to learn word meaning while considering inter-sentential variations. Second, we made a separate column ('Label') to indicate the intended function of each postposition in each sentence (Figure 1). We then split the corpus into two sub-sets, one with 90 per cent of the corpus for the training and with the remaining 10 per cent of the corpus for the testing.

Index	Label	Sentence
1,862	1	[CLS] 한참 만에 오반장이 침묵을 깼다. [SEP]
1,863	1	[CLS] 정말 오랫만에 먹어보는 고기였다. [SEP]
1,864	1	[CLS] 옛날 구한말에 유명한 얘기가 있었죠? [SEP]
1,865	1	[CLS] 한밤중에 신나게 한바탕했지요. [SEP]
1,866	1	[CLS] 그런데 몇 시에 왔어? [SEP]
1,867	1	[CLS] 겨울에 꽃이라니요. [SEP]
1,868	1	[CLS] 아침에 엄마한테 돈을 달랬어요. [SEP]
1,869	1	[CLS] 결혼은 반드시 적령기에 해야 한다. [SEP]
1,870	1	[CLS] 한 달에 얼마씩은 정확하게 들어오니까. [SEP]
1,871	1	[CLS] 그럼 일 주일 후에 뵙겠습니다. [SEP]

Figure 1: Example sentences used in the BERT training (-*ey*, CRT)

3.3 Developing BERT classification model

We set the parameters related to BERT training such as *batch size* (32), *epoch* (50), *seed* (42), *epsilon* (0.00000008), and *learning rate* (0.00002), as advised by McCormick (2019). We then employed a pre-trained language model in order to obtain high accuracy of outcomes; for this purpose, we used a Korean BERT model (KoBERT; Jeon et al., 2019). Before the actual BERT training, we transformed the input data into three embedding types—*token* embeddings, *position* embeddings, and *segment* embeddings (cf., Devlin et al., 2018)—in the following ways.

First, for the *token* embedding, we used *KoBertTokenizer* for the sentence tokenization (the maximum number of tokens for each sentence was set to 128). Second, we converted each token into numeric values indicating unique indices of the tokens in the vocabulary of KoBERT for the *position* embedding. Third, for the *segment* embedding, we converted the number of tokens of each sentence into 128 numeric values using 0 (i.e., not existed) or 1 (i.e., existed). The labels of the data indicating the intended function of each postposition in the sentence were stored separately.

After this transformation step, we proceeded to the model training as follows. We first loaded KoBERT through the function *BertForSequenceClassification* from *transformers* (Wolf et al., 2019).

[2]Our corpus is available at: we will propose a URL

Next, we fine-tuned the pre-trained model by using the training set, with a view to reducing loss values and updating the *learning rate* for better classification accuracy of the model. We then loaded the testing set to evaluate whether the fine-tuned model successfully recognized the intended functions of each postposition in each sentence. In this part, the rates of accuracy for each function and the total accuracy rate were calculated by comparing the intended function of each postposition in each test sentence with the classified function of each postposition via the BERT model. Lastly, we employed *t*-distributed Stochastic Neighbor Embedding (t-SNE; Maaten & Hinton, 2008) for dimension reduction of classification embeddings from the postposition per epoch. In addition, to statistically confirm the changes of sentence-level embedding outcomes by each epoch, we performed density-based clustering (Sander et al., 1998). These outcomes were fed into the visualization system, which we outline next.

3.4 DEVELOPING VISUALIZATION SYSTEM

We designed a visualization system with JavaScript, HTML, and CSS environments, using the test set under the two-dimensional distribution. For the interface of this system, we created three areas showing model performance: a distributional map for sentence-level embeddings, accuracy/loss charts relating to the model, and graphs for the density-based clustering. To manipulate visualization outcomes, Figure 2(a) provides options to select the postpositions and checkboxes to highlight and tracking interesting sentences according to the index number or the function of these postpositions. The distributional map as in Figure 2(b) presents the relationship between the sentences with the selected postposition (represented as dots) involving different functions (represented as colors). A slider at the bottom of the map allows for changing the epochs; the patterns of clustering change as the slider moves. Each dot shows the details of the sentence (e.g., an index of the selected sentence, the intended function used in the sentence, the original sentence) once the mouse pointer is located on the dot. The right side of the system as in Figure 2(c) provides users with various information about the model performance: overall accuracy, by-function accuracy, and loss rates in the classification task by epoch. This section also provides accuracy rates of each function by hovering around the mouse pointer onto the specific-colored lines. The bar chart at the bottom right side of the system presents the number of clusters produced by the model. This chart also provides a hovering function, providing the actual number of clusters per epoch. The particular hovering activity is interlocked with the density cluster view, located at the bottom left of the system, by presenting the clustering results according to the selected epoch.

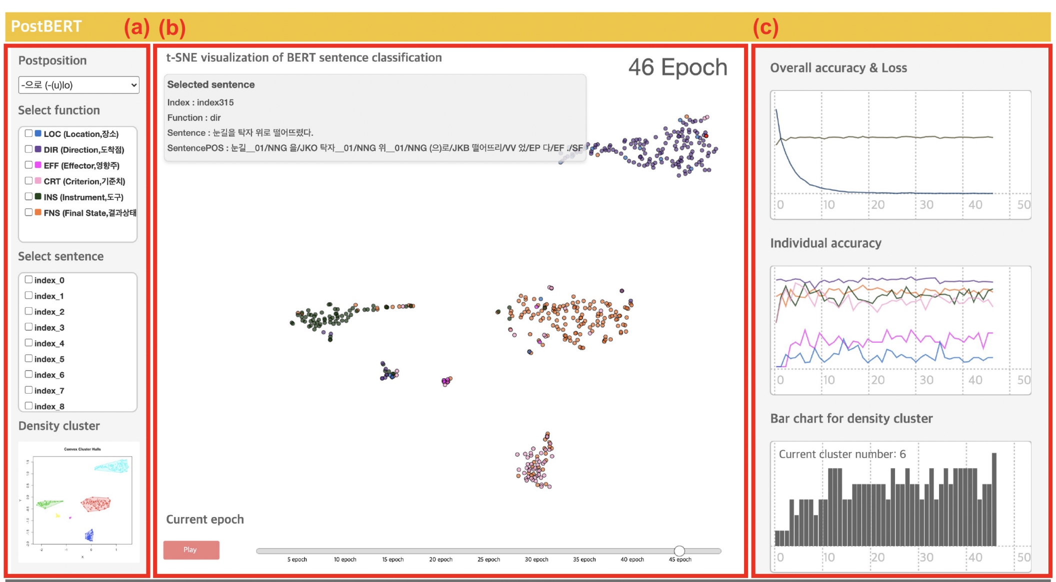

Figure 2: The overall interface of the visualization system (Available at: http://13.125.253.195/PostBERT/).

Epoch	Classification accuracy								
	Overall	*AGT*	*CRT*	*EFF*	*FNS*	*GOL*	*INS*	*LOC*	*THM*
1	0.682	0	0.876	0	0	0.044	0	0.911	0.198
10	0.819	0	0.930	0.433	0.578	0.313	0.133	0.954	0.688
20	0.817	0.067	0.897	0.533	0.533	0.186	0.067	0.960	0.916
30	0.824	0.067	0.915	0.378	0.444	0.328	0.067	0.948	0.718
40	0.826	0.067	0.892	0.489	0.467	0.326	0.133	0.942	0.768
50	0.824	0.067	0.912	0.411	0.389	0.409	0.1	0.940	0.683
Average	0.815	0.041	0.911	0.439	0.497	0.328	0.076	0.947	0.713

Table 2: By-function accuracy for the BERT model: *-ey*

Epoch	Classification accuracy		
	Overall	*LOC*	*SRC*
1	0.863	0.980	0.174
10	0.9	0.939	0.559
20	0.898	0.937	0.651
30	0.896	0.949	0.464
40	0.912	0.963	0.523
50	0.916	0.960	0.598
Average	0.898	0.948	0.535

Table 3: By-function accuracy for the BERT model: *-eyse*

Epoch	Classification accuracy						
	Overall	*CRT*	*DIR*	*EFF*	*FNS*	*INS*	*LOC*
1	0.704	0.476	0.943	0	0.764	0.477	0
10	0.814	0.83	0.918	0.367	0.771	0.835	0.1
20	0.812	0.694	0.951	0.3	0.838	0.709	0.044
30	0.816	0.708	0.941	0.333	0.811	0.752	0.05
40	0.819	0.694	0.927	0.267	0.855	0.777	0.05
50	0.821	0.692	0.957	0.4	0.836	0.723	0.1
Average	0.813	0.721	0.938	0.278	0.815	0.763	0.106

Table 4: By-function accuracy for the BERT model: *-(u)lo*

4 RESULTS: THREE CASE STUDIES

In order to reports the BERT model performance of classifying the functions of postpositions and assess how our visualization system works, we conducted three case studies.

4.1 DOES THE NUMBER OF FUNCTIONS FOR A POSTPOSITION AFFECT MODEL PERFORMANCE?

The presenting tables (Tables 2-4) show the classification accuracy of the BERT model for each postposition. The result show that the BERT model performed better for *-eyse*, which has only two functions (SRC and LOC), than for the other two postpositions (*-ey* and *-(u)lo*). The average classification accuracy for *-ey*, *-eyse* and *-(u)lo* is a satisfactory level of accuracy considering previous reports (Bae et al., 2014; Kim & Ock, 2016).

To statistically explore the classification by postposition/epoch, we performed a two-way ANOVA. As Table 5 shows, there is no statistical significance in the accuracy across the postpositions but significant difference in the accuracy across the epochs. This indicates the general tendency that model performance improved in proportion to the number of epochs (see also case study 3).

Comparison	$\|F\|$	p
Postposition	0.070	.792
Epoch	6.457	.012*
Postposition x Epoch	0.579	.448

Table 5: Results of the two-way ANOVA

Note. * $<$ .05

We conducted additional by-postposition pairwise comparisons through a two-sample t-test. As Table 6 shows, the model performance in *-eyse* is significantly better than in the other two postpositions. Considering the different number of functions (e.g., two for *-eyse*, six for *-(u)lo*, and eight for *-ey*), this finding indicates an inverse relationship between the classification accuracy and the number of functions that each postposition manifests.

Comparison	$\|t\|$	p
-ey vs. *-eyse*	22.588	$<$ .001***
-ey vs. *-(u)lo*	0.533	.594
-eyse vs. *-(u)lo*	28.301	$<$ .001***

Table 6: Statistical comparison of each postposition: Two-sample t-test

Note. *** $<$ .001

4.2 Do the asymmetric proportions of the functions in each postposition affect the model performance?

The answer is *they do*. The average classification accuracy of each function for *-ey* is the highest for LOC (0.947) and the lowest for AGT (0.041); for *-eyse*, it is the highest in LOC (0.948) and the lowest in SRC (0.535); for *-(u)lo*, it is the highest in DIR (0.938) and the lowest in LOC (0.106) (Tables 2-4). As for the occurrences of individual functions per postposition, LOC for *-ey*, LOC for *-eyse*, and DIR for *-(u)lo* account for the larger portion of the entire corpus than other functions (see Table 1). This finding thus indicates that the model performance is affected by the asymmetric proportions of the functions comprising the use of each postposition.

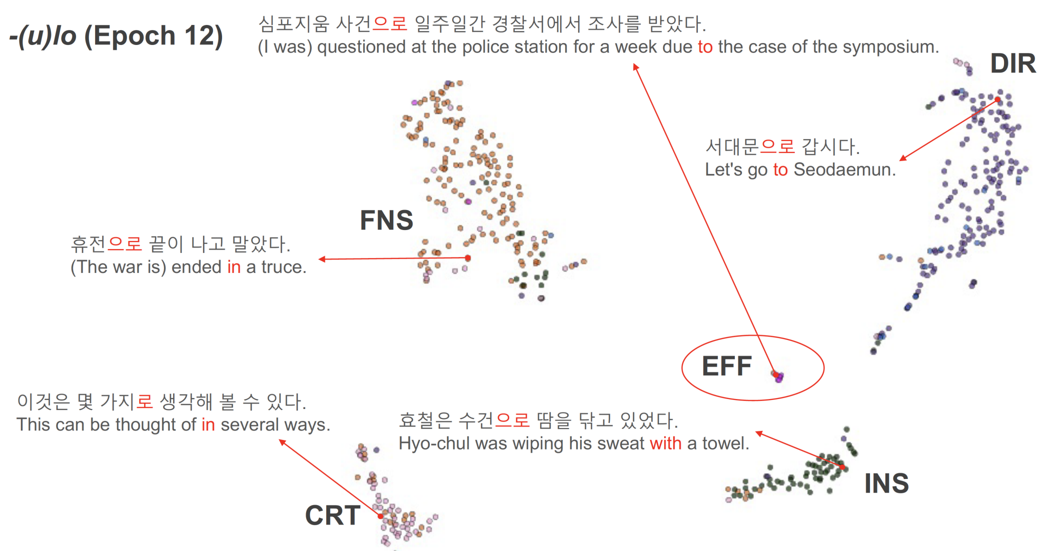

Figure 3: The distributional map for *-(u)lo* in Epoch 12.

4.3 How does the BERT model classify sentences based on the postpositions' functions as the epoch progresses?

Our visualization system showed that the model was able to recognize the functions of each postposition as the epoch progressed. For *-ey*, all of the sentences were divided into two groups when the epoch was one, but as the epoch progressed, the sentences were divided into three in Epoch 7, four in Epoch 12, and five in Epoch 15. For *-eyse*, the number of clusters was one when the epoch was one, and there were two clusters when the epoch was nine. For *-(u)lo*, the number of clusters increased, starting from one (Epoch 1) to three (Epoch 4), five (Epoch 12), and six (Epoch 46).
In particular, for *-(u)lo*, there are two interesting findings. First, in Epoch 12 (Figure 3), a cluster of EFF (the function with low-frequency occurrences in the data) emerged. This finding indicates that the BERT can identify functions at a satisfactory level, even though they are relatively infrequent, as long as there are sufficient epochs provided. Second, interestingly, LOC could not form a designated cluster in the end. Highlighting and zooming into the individual instances of LOC (Figure 4), we found that many of the LOC instances (11 out of 15) belonged to the DIR group. This is due to (i) the low frequency of LOC in the data and (ii) the semantic closeness between DIR and LOC—they relate to a location and are often difficult to distinguish one from another. This finding indicates that there are still some limitations in regard to the identification of functions given the above complications.

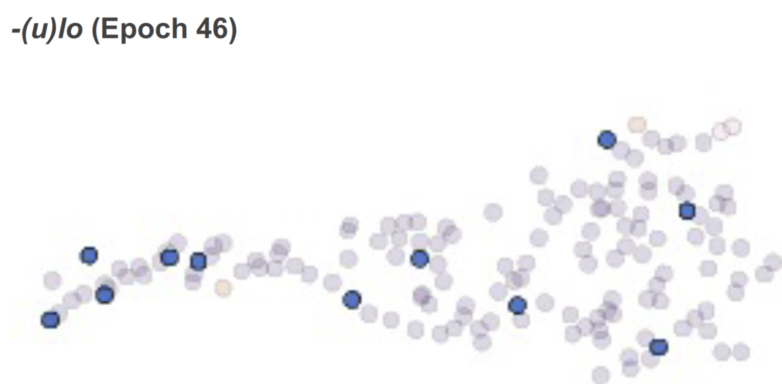

Figure 4: The DIR cluster in the distributional map for *-(u)lo* (Epoch 46) highlighting the LOC instances.

5 Conclusion

In this study, we note three major findings. First, there is an inverse relation between the classification accuracy and the number of functions of each postposition. Second, the model is affected by the corpus size of each function. Third, the model can identify the intended functions of a postposition as the epoch progresses, even though the corpus size of a function is small. However, despite these findings, our BERT model still seems to be affected by the scarcity of input and/or semantic closeness between the items, limiting its performance in the given task to some extent. We believe our visualization system will contribute to extending the current understanding of how BERT works for language tasks (particularly in non-English settings).

The findings of this study should be further verified by incorporating more postposition types that have similar degrees of polysemy that three adverbial postpositions demonstrate, which we plan to pursue next. Researchers will also benefit from considering other *contextualized* word-embedding models such as Generation Pre-trained Transformer 3 (Brown et al., 2020) or a Robustly Optimized BERT Pretraining Approach (Liu et al., 2019b) to better ascertain the advantage of BERT in this kind of task.

REFERENCES

Jangseong Bae, Junho Oh, Hyunsun Hwang, and Changki Lee. Extending korean propbank for korean semantic role labeling and applying domain adaptation technique. *Korean Information Processing Society*, pp. 44–47, 2014.

Jangseong Bae, Changki Lee, and Soojong Lim. Korean semantic role labeling using deep learning. *Korean Information Science Society*, 6:690–692, 2015.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners, 2020.

Jeong-mi Cho and Gil-cheng Kim. A study on the resolving of the ambiguity while interpretation of meaning in korean. *The Korean Institute of Information Scientists and Engineers*, 14(7):71–83, 1996.

Miho Choo and Hye-young Kwak. *Using Korean*. Cambridge University Press, New York, NY, 2008.

Guillaume Desagulier. Can word vectors help corpus linguists? *Studia Neophilologica*, 91(2):219–240, 2019. doi: 10.1080/00393274.2019.1616220. URL `https://doi.org/10.1080/00393274.2019.1616220`.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018. URL `http://arxiv.org/abs/1810.04805`.

Kawin Ethayarajh. How contextual are contextualized word representations? comparing the geometry of BERT, ELMo, and GPT-2 embeddings. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 55–65, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1006. URL `https://www.aclweb.org/anthology/D19-1006`.

Dylan Glynn and Justyna Robinson. *Corpus Methods for Semantics*. Corpus Methods for Semantics. Quantitative studies in polysemy and synonymy. John Benjamins, January 2014. doi: 10.1075/hcp.43. URL `https://halshs.archives-ouvertes.fr/halshs-01284061`.

Heewon Jeon, Donggeon Lee, and Jangwon Park. Korean bert pre-trained cased (kobert), 2019. URL `https://github.com/SKTBrain/KoBERT`.

Byong-cheol Jeong. An integrated study on the particle '*-ey*' based on the simulation model. *The Linguistic Science Society*, 55:275–304, 2010.

Wan-su Kim and Cheol-young Ock. Korean semantic role labeling using case frame dictionary and subcategorization. *The Korean Institute of Information Scientists and Engineers*, 43(12):1376–1384, 2016.

Changki Lee, Soojong Lim, and Hyunki Kim. Korean semantic role labeling using structured svm. *The Korean Institute of Information Scientists and Engineers*, 42(2):220–226, 2015.

Nelson F. Liu, Matt Gardner, Yonatan Belinkov, Matthew E. Peters, and Noah A. Smith. Linguistic knowledge and transferability of contextual representations. 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)*, pp. 1073–1094, Minneapolis, Minnesota, June 2019a. Association for Computational Linguistics. doi: 10.18653/v1/N19-1112. URL `https://www.aclweb.org/anthology/N19-1112`.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach, 2019b. URL `http://arxiv.org/abs/1907.11692`. cite arxiv:1907.11692.

Laurens van der Maaten and Geoffrey Hinton. Visualizing Data using t-SNE. *Journal of Machine Learning Research*, 9(Nov):2579–2605, 2008. ISSN ISSN 1533-7928. URL `http://jmlr.org/papers/v9/vandermaaten08a.html`.

Chris McCormick. Bert fine-tuning tutorial with pytorch, 2019. URL `http://mccormickml.com/2019/07/22/BERT-fine-tuning/`.

Seongmin Mun and Gyu-Ho Shin. Context window and polysemy interpretation: A case of korean adverbial postposition *-(u)lo*. In *IMPRS Conference 2020: Interdisciplinary Approaches to the Language Sciences, Max Planck Institute for Psycholinguistics*, 2020.

Ki-sim Nam. The use of the korean postposition: focus on '*-ey*' and '*-(u)lo*'. *sekwang hakswul calyosa*, 1993.

Jeong-woon Park. A polysemy network of the korean instrumental case. *Korean Journal of Linguistics*, 24(3):405–425, 1999.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations, 2018. URL `http://arxiv.org/abs/1802.05365`. cite arxiv:1802.05365Comment: NAACL 2018. Originally posted to openreview 27 Oct 2017. v2 updated for NAACL camera ready.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training, 2018. URL `https://s3-us-west-2.amazonaws.com/openai-assets/research-covers/language-unsupervised/language_understanding_paper.pdf`.

Jörg Sander, Martin Ester, Hans-Peter Kriegel, and Xiaowei Xu. Density-based clustering in spatial databases: The algorithm gdbscan and its applications. *Data Mining and Knowledge Discovery*, 2(2):169–194, jun 1998. URL `http://dx.doi.org/10.1023/A:1009745219419`.

Hyo-pil Shin. The 21st sejong project : with a focus on selk (sejong electronic lexicon of korean) and the knc (korean national corpus). In *In The 3rd International Joint Conference on Natural Langauge Processing*, 2008.

Myung-chul Shin, Yong-hun Lee, Mi-young Kim, You-jin Chung, and Jong-hyeok Lee. Semantic role assignment for korean adverbial case using sejong electronic dictionary. *Korea Information Science Society*, pp. 120–126, 2005.

Ho-Min Sohn. *The korean language*. Cambridge University Press, Cambridge, UK, 1999.

Dae-heon Song. A study on the adverbial case particles of '*-ey*' and '*-eyse*' for korean language education. *The Association of Korean Education*, 101:457–484, 2014.

Raphael Tang, Yao Lu, Linqing Liu, Lili Mou, Olga Vechtomova, and Jimmy Lin. Distilling task-specific knowledge from BERT into simple neural networks. *CoRR*, abs/1903.12136, 2019. URL `http://arxiv.org/abs/1903.12136`.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. Huggingface's transformers: State-of-the-art natural language processing. *CoRR*, abs/1910.03771, 2019. URL `http://arxiv.org/abs/1910.03771`.