When the Misidentified Adverbial Phrase Functions as a Complement

Yige Chen[1] **Kyuwon Kim**[2] **KyungTae Lim**[3*] **Jungyeul Park**[4*] **Chulwoo Park**[5]
[1]The Chinese University of Hong Kong, Hong Kong [2]Seoul National University, South Korea
[3]SeoulTech, South Korea [4]The University of British Columbia, Canada
[5]Anyang University, South Korea
yigechen@link.cuhk.edu.hk guwon0406@snu.ac.kr
ktlim@seoultech.ac.kr jungyeul@mail.ubc.ca cwpa@anyang.ac.kr

Abstract

This study investigates the predicate-argument structure in Korean language processing. Despite the importance of distinguishing mandatory arguments and optional modifiers in sentences, research in this area has been limited. We introduce a dataset with token-level annotations which labels mandatory and optional elements as complements and adjuncts, respectively. Particularly, we reclassify certain Korean phrases, previously misidentified as adverbial phrases, as complements, addressing misuses of the term adjunct in existing Korean treebanks. Utilizing a Korean dependency treebank, we develop an automatic labeling technique for complements and adjuncts. Experiments using the proposed dataset yield satisfying results, demonstrating that the dataset is trainable and reliable.

1 Introduction

Research on the predicate-argument structure of natural languages has been at the center of the studies of syntax and semantics. The relations between the predicate and its arguments in a sentence or a clause not only reveal the type of arguments the particular predicate needs syntactically, but also shed light on the possible semantics the predicate can provide. In the field of natural language processing (NLP), tasks such as syntactic parsing and semantic role labeling (SRL) largely rely on some assumptions and groundwork concerning the predicate-argument structure to understand which arguments play an essential role in the clause, and which arguments do not. Despite the importance of understanding the predicate-argument structure and determining whether a syntactic daughter[1] of a predicate appears to be mandatory or optional, studies on differentiating core and non-core argument candidates[2] have been lacking. This is largely due to the fact that such datasets are not necessarily associated with any specific NLP task directly. On the other hand, both linguistics and NLP can benefit from a dataset that clearly marks the core and non-core syntactic daughters dependent on the predicate in Korean. Moreover, whether a syntactic daughter is mandatory or optional can serve as an additional feature during syntactic and semantic processing of the Korean language and boost the performance of the models on related NLP tasks. Unfortunately, there is no dataset with the aforementioned annotation for the Korean language, and no method that labels classified syntactic daughters has been proposed.

Korean is an agglutinative language that follows a subject-object-verb (SOV) word order. One of the major characteristics that differentiates Korean from many other non-agglutinative languages is its extensive use of postpositions as case markers. Postpositional markers are suffixes that adhere to the root of the words, and they are functional morphemes that modify the semantics of the stem lexeme. Postpositions of arguments are usually morphological cases, which are externalizations of the corresponding grammatical cases the arguments carry. On the other hand, Kim and Sells (2010) argue that some oblique markers do not indicate any semantic role. Rather, they present semantic and pragmatic information given a construction. Park and Kim (2023) further investigate Korean postpositions under the framework of Categorial Grammar, suggesting that not all Korean postpositions mark the case.

In this study, we present a corpus with token-

[*]Corresponding authors

[1]A syntactic daughter of a node is a constituent that is dependent on the node in the syntax tree (Pollard and Sag, 1987).

[2]In this study, we define an argument candidate as an element that may be either a complement (i.e., a core argument) or an adjunct (i.e., a non-core element that is sometimes misidentified as an argument, which we aim to correctly classify). We intentionally use the term *candidate* to indicate that the element may or may not be a true argument.

Findings of the Association for Computational Linguistics: EACL 2024, pages 12326–12336

level annotations revealing the mandatory and optional elements in the predicate-argument structure. To the best of the authors' knowledge, this is the first study attempting to reclassify certain phrases in Korean, previously misidentified as adverbial phrases[3] and modifiers, into complements, in order to address the misuse of the term *adjunct* in existing Korean treebanks. We follow the binary distinction based on the dictionary definition and clearly define and distinguish the core and non-core argument candidates as complements and adjuncts, respectively, and demonstrate the linguistic rationale behind our proposed binary distinction. We utilize an existing Korean dependency treebank, and develop a novel technique for the automatic generation of complement and adjunct labels onto the sentences in the treebank. A dataset is therefore constructed with all complements and adjuncts for the verbal predicates in the treebank's sentences annotated. We further illustrate different levels of Korean predicate-argument structures by analyzing some converted data from the treebank, and train sequence labeling models using the constructed dataset. Experiment results suggest that our dataset is trainable and reliable.

2 Linguistic Debates

For several decades, discussions have unfolded around the characteristics and semantic functions of arguments and modifiers within the realm of generative grammar. Linguists who utilize structures such as Principles and Parameters (P&P) (Chomsky, 1986, p.150-151) or Head-driven Phrase Structure Grammar (HPSG) (Pollard and Sag, 1994) make a tripartite distinction (Carnie, 2002; Sag et al., 2003). They argue for the uniqueness of the specifier (an immediate subordinate of the phrase, often the subject) and the complement (the counterpart of the head, typically the object) within a verb phrase, apart from the adjuncts (optional and serve to refine the head's meaning). Categorial Grammar (CG) (Ajdukiewicz, 1935; Bar-Hillel, 1953) gives an alternative analysis of the predicate-argument structure. A binary distinction is made between elements associated with the predicate: the mandatory complements that complete the meaning of their head and the optional adjuncts that offer further semantics to the head's meaning (Dowty, 2003). What P&P and HPSG categorize as both 'specifier' and 'complement' are collectively termed as 'complement' under Categorial Grammar.

In our study, we utilize the manually-created subcategorization frame information and make a binary distinction between mandatory complements (i.e., arguments) and optional adjuncts (i.e., modifiers) within the predicate-argument structure of Korean. Complements are syntactically obligatory elements that serve as arguments at the semantic level, while adjuncts are syntactically optional and function as modifiers semantically. Complements and adjuncts can be distinguished through the application of a relativization test, wherein complements can function as the head of a relative clause, while adjuncts cannot (Park, 2002). Another criterion has been suggested by utilizing particle-elimination wherein complements can occur without case makers, while adjuncts cannot (Kim, 2004). Previous Korean treebanks classify all phrases with non-subject and non-object postpositions as adverbial and categorize them as adjuncts, whereas by strictly following the distinction mentioned above, it becomes evident that some of these so-called 'adverbial phrases' are selected by the predicate and should therefore be considered complements. Specifically, adverbial phrases, when correctly identified in their syntactic function, always act as adjuncts and never as complements.

3 Methods

Data In this study, we use the KLUE dependency treebank as the main source. The KLUE dependency treebank consists of 14,500 annotated sentences in total, whereas 10,000 sentences from the training set and 2,000 sentences from the development set are available. We use only its training set for the purpose of this study. Unlike the annotation scheme of Korean UDs, the KLUE dependency treebank adopts the Dependency annotation proposed by Korea's Telecommunications Technology Association (TTA) in which 9 syntax tags and 6 function tags are used to mark the dependency relations. Apart from the dependency relation and its head, the treebank also includes the canonical form of the token and its language-specific part-of-speech (XPOS) for each token (Appendix A.1).

To better understand the possible arguments of Korean verbs, our study utilizes the Sejong dictio-

[3]The term 'adverbial phrase' in this study particularly refers to phrases that possess adverbial characteristics and function as components modifying the verb. This should be distinguished from the concept of an adverbial phrase (AdvP) in generative syntax, where it denotes phrases with adverbs as their heads.

nary. This dictionary is part of the Sejong corpus, coordinated by the National Institute of Korean Language. The Sejong dictionary is organized such that for each verbal predicate in Korean, all possible senses and their detailed syntactic and semantic information are provided (Appendix A.2). Specifically, we rely on the subcategorization frames of the verbal predicate. It is worth noting that the subcategorization frame encompasses the required arguments alongside their postpositions.

Extraction of complements and adjuncts A heuristic method is implemented for the extraction of complements and adjuncts from the KLUE dependency treebank, based on existing annotations of dependency relations in the treebank and the Sejong dictionary. For an annotated sentence from the treebank, the method first locates all possible verbal predicates in the sentence, based on the XPOS tag of the token and whether the canonical form of the verbal component of the token has its corresponding entry in the Sejong dictionary. For each verbal predicate that is found, the whole dependency tree is traversed to find the constituents that serve as its daughters. Following this, the heuristic method refers to the subcategorization frames of this verb from the Sejong dictionary, and for each frame, it compares the postpositions of the arguments specified in the frame and the postpositions of the constituents that are daughters of the verb. Those constituents whose postpositions can be found in the frames are considered complements, whereas the daughters whose postpositions are not included in the frame arguments are classified as adjuncts. The only exceptions are the syntactic daughters of the verb which do not have postpositions. In these cases, they are excluded from the aforementioned binary distinction, and are neither counted as complements nor counted as adjuncts. Subsequently, the frame(s) that best fit the daughters of the verbal predicate are returned. Examples of projected sentences are presented in Appendix B, and sequence labeling experiments on the constructed dataset are conducted in Appendix C.

4 Data Analysis

We count the numbers of instances that exactly match, partially match, or do not match the frame arguments, based on the extracted adjuncts and complements of the predicates. Table 1 shows the number of perfect match, partial match (three subtypes), and no match cases within the entire training set of the KLUE dependency treebank.

Match type	Counts
Perfect Match	3,577
Partial Match, all sentence args	8,610
Partial Match, all frame args	1,430
Partial Match, both	88
No Match	9,259
Total	22,964

Table 1: Counts of matched and non-matched instances.

Figure 1 illustrates three types of projections between sentence argument candidates (i.e., the nominal daughters of a predicate in a sentence ending with postpositions) and frame arguments (i.e., the prescribed possible arguments as listed in a subcategorization frame of the predicate) for different predicates in a single sentence. A perfect match, as in Figure 1a, indicates a one-to-one correspondence between the frame arguments and the sentence argument candidates for a certain frame of the verb. In this case, all three arguments suggested by the frame are mapped onto the three argument candidates in the sentence based on their postpositions[4], and vice versa. On the other hand, a partial match may be one of the following three cases: (1) all frame arguments are found in the sentence, but not vice versa; (2) all sentence argument candidates are found in the frame, but not vice versa; (3) frame(s) that satisfy (1) and frame(s) that satisfy (2) are found, but no perfect match exists. No match denotes the case where there's no perfect or partial match as defined above, or the detected predicate has no argument candidate.

One of the most common causes of case (1) is noun phrase ellipsis (NP-ellipsis), in which nouns or noun phrases get elided. In Figure 1b, the subject of the clause is *A ssi-neun* ('Mr. A') which is shared with the preceding clause as shown in Figure 1a. The latter subject is therefore elided, resulting in a mismatch where the subject, as suggested by the frame, cannot be located in the clause. For case (2), instead of being a complement, an argument candidate may be an adjunct, as long as it cannot be located in the frame. Figure 1c serves as a good example, where its subject and object are paired with the corresponding arguments in the frame and are complements, but the remaining

[4]Korean postpositions denoting the same case may be realized in different forms. The accusative marker *eul*, for instance, is realized as *leul* in Figure 1a. Moreover, both the nominative marker *eun/neun* and the topic marker *i/ga* indicate subjects, and as a result are considered identical during the mapping.

Predicate: 제출하다 (*jechulhada* 'to submit')
Type: Perfect Match

Frame:	X=N0-이	Y=N1-을	Z=N2-에|에게|로	V
	arg="X" tht="AGT"	arg="Y" tht="THM"	arg="Z" tht="GOL"	
Sentence:	[A 씨는]$_{AGT}$	[전치 6주 진단서를]$_{THM}$	[경찰에]$_{GOL}$	[제출하고]$_{TARGET}$ ...
	A ssi-neun	*jeonchi 6ju jindanseo-leul*	*gyeongchal-e*	*jechulhago* ...
	Mr. A.top	a 6-week medical certificate.acc	the police.dat	submit

'Mr. A submitted a 6-week medical certificate to the police and...'

(a) A perfect match (one-to-one correspondence between frame arguments and sentence arguments).

Predicate: 마치다 (*machida* 'to finish')
Type: Partial Match, all arguments in the frame are paired

Frame:	X=N0-이	Y=N1-을	V
	arg="X" tht="AGT"	arg="Y" tht="THM"	
Sentence:	...	[고소인 진술을]$_{THM}$	[마쳤으며]$_{TARGET}$ ...
	...	*gosoin jinsul-eul*	*machyeoss-eumyeo* ...
		accuser's statement.acc	complete

'...and completed the accuser's statement and...'

(b) A partial match (all frame arguments paired).

Predicate: 부르다 (*buleuda* 'to call/summon')
Type: Partial Match, all nominal daughters of the predicate in the sentence ending with postpositions are paired

Frame:	X=N0-이		Y=N1-을		V
	arg="X" tht="AGT"		arg="Y" tht="THM"		
Sentence:	... [경찰은]$_{AGT}$	[조만간]$_{ADV}$	[김현중을]$_{THM}$	[피고소인 신분으로]$_{AJT}$	[불러]$_{TARGET}$ ...
	... *gyeongchal-eun*	*jomangan*	*gimhyeonjung-eul*	*pigosoin sinbun-eulo*	*bulleo* ...
	the police.top	soon	Kim Hyun-joong.acc	with defendant status	summon

'...and the police plans to summon Kim Hyun-joong as a defendant in the near future and...'

(c) A partial match (all sentence arguments paired).

Figure 1: Examples of subcategorization frame to sentence projections. X=N0-이 denotes that argument X is nominal and ends with the postposition 이/가/은/는 *i/ga/eun/neun* (topic/nominative case). Y=N1-을 denotes that argument Y is nominal and ends with the postposition 을/를 *eul/leul* (accusative case). Z=N2-에|에게|로 denotes that argument Z is nominal and ends with the postposition 에/에게/로 *e/ege/lo* (dative case).

candidate *pigosoin sinbun-eulo* ('with defendant status') with the postposition *eulo* does not have its corresponding argument in the frame, and is considered an adjunct.

A significant distinction is that some seemingly 'adverbial' phrases are now considered complements rather than adjuncts. Figure 1a features three arguments, including the subject, the object, and the dative *gyeongchal-e* ('to the police'). While the subject and the object are considered complements indisputably, the phrase *gyeongchal-e* is considered an adjunct in the treebank. However, the subcategorization frame from the dictionary suggests that the phrase, carrying a dative case with the postposition *e*, is a required argument of the predicate and should therefore be considered a complement, as marked in Figure 1a.[5]

It is worth mentioning that we consider adverbs and clausal constructions neither complements nor adjuncts, and excluded them from the list of argument candidates. They usually lack postposition markers and do not contribute to the semantics of the predicate in a way complements and adjuncts do. As Figure 1c suggests, the adverb *jomangan* ('soon') is not a valid argument candidate and is not marked as either a complement or an adjunct by the proposed extraction method.

Within the 32,223 sentence-frame pairs where a match, either perfect or partial, is guaranteed, 21,155 complements and 6,757 adjuncts are found. This suggests an overwhelmingly larger portion of complements compared to adjuncts in Korean.

5 Related Work

There have been numerous linguistic resources constructed for the purpose of syntactic and semantic analyses. Among them, dependency treebanks provide syntactic information in dependency grammar,

[5]A similar example can be found in Figure 4 in Appendix B, where the seemingly 'adverbial' phrase *jeomsu cha seunglilo* ('with decisive score margins') is labeled adjunct (NP_AJT) in the treebank but complement (COMP) in our dataset.

and they are usually utilized for dependency parsing tasks in natural language processing. Universal Dependencies (UDs) (de Marneffe et al., 2021) is designed to ensure consistent labeling of grammatical structures, such as parts of speech, morphological features, and syntactic dependencies, across different languages. For Korean, annotation guidelines have been proposed for the construction of Korean UD treebanks (Seo et al., 2019).

Three different constituency treebanks, namely the KAIST treebank (Choi et al., 1994), the Sejong treebank, and the Penn Korean treebank (Han et al., 2002), have been proposed for the Korean language. However, only the Sejong and the Penn Korean treebanks represent constituents of a clause, including subjects and objects. This is facilitated by the distinction made possible by nominative and accusative postpositions in Korean. A noun phrase that ends with an adverbial postposition, such as a dative, locative, or temporal phrase, is categorized as either an adjunct (`-AJT`) or a complement (`-COMP`) in both the Sejong and Korean Penn treebanks. Although `-COMP` in the Penn Korean treebank, distinct from subjects (`-SBJ`) or objects (`-OBJ`), signifies a complement, its function aligns more closely with that of an adjunct.

Several resources of dependency parsing data for Korean have been made available in the past, including the GSD treebank (McDonald et al., 2013), the Kaist dependency treebank (Chun et al., 2018), the Penn Korean universal dependency treebank[6] (Han et al., 2002), and the Korean Language Understanding Evaluation (KLUE) benchmark dependency parsing dataset (Park et al., 2021).

Semantic analyses of the Korean language benefit from the task of SRL. Kim et al. (2014) adapt and conduct SRL on Korean, and Chen et al. (2024) introduce annotation strategies for Korean SRL with a constructed dataset. However, none of the previous work has pointed out that some so-called 'adverbial phrases' are misidentified and should be classified as complements, diverging from the perspectives of syntacticians and semanticists.

6 Conclusions

In this study, we refine the distinction between core and non-core argument candidates, namely complements and adjuncts, in Korean. A binary distinction based on the linguistic features of Korean is described herein, and we propose and implement an effective method to automatically generate the labels of complement/adjunct. Data from a dependency treebank is leveraged to construct a corpus with annotations indicating both complements and adjuncts of all possible verbal predicates from the sentences in this treebank. Sequence labeling models trained on the proposed dataset give satisfactory results, indicating that the constructed dataset is of good quality.

We believe that the dataset can serve as supplementary data in both linguistics and NLP. The proposed method would save the labor of the linguists such that they could focus on analyses without sacrificing time on finding examples with specific predicate-argument structures. The dataset is also suitable for training language models for a better understanding of the Korean sentence structure, which can benefit downstream tasks.[7]

Limitations

Given the comprehensive nature of this study, we believe there are no significant limitations that need to be acknowledged to provide context for the findings or to guide future research efforts.

Acknowledgments

We would like to thank Professor Haihua Pan for his insightful feedback throughout the study, particularly on the disambiguation of the terminologies of adverbial phrases, adjuncts, and complements. This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant, funded by the Korea government (MSIT) (No.RS-2024-00456709, A Development of Self-Evolving Deepfake Detection Technology to Prevent the Socially Malicious Use of Generative AI) awarded to KyungTae Lim.

References

Kazimierz Ajdukiewicz. 1935. Die syntaktische Konexität. *Studia philosophica*, 1:1–27.

Yehoshua Bar-Hillel. 1953. A quasi-arithmetical notation for syntactic description. *Language*, 29(1):47–58.

Andrew Carnie. 2002. *Syntax: A Generative Introduction*. Introducing linguistics. Wiley-Blackwell, New Jersey, United States.

[6] https://catalog.ldc.upenn.edu/LDC2023T05

[7] Dataset available at https://github.com/universalkorean/adverbial-phrase

Yige Chen, KyungTae Lim, and Jungyeul Park. 2024. A Linguistically-Informed Annotation Strategy for Korean Semantic Role Labeling. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 733–738, Torino, Italy. ELRA and ICCL.

Key-Sun Choi, Young S. Han, Young G. Han, and Oh W. Kwon. 1994. KAIST Tree Bank Project for Korean: Present and Future Development. In *Proceedings of the International Workshop on Sharable Natural Language Resources*, pages 7–14, Nara Institute of Science and Technology. Nara Institute of Science and Technology.

Noam Chomsky. 1986. *Knowledge of Language: Its Nature, Origin, and Use*. Praeger Scientific, New York.

Jayeol Chun, Na-Rae Han, Jena D. Hwang, and Jinho D. Choi. 2018. Building Universal Dependency Treebanks in Korean. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised Cross-lingual Representation Learning at Scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Marie-Catherine de Marneffe, Christopher D. Manning, Joakim Nivre, and Daniel Zeman. 2021. Universal Dependencies. *Computational Linguistics*, 47(2):255–308.

David Dowty. 2003. The dual analysis of adjuncts/complements in Categorial Grammar. In Ewald Lang, Claudia Maienborn, and Cathrine Fabricius-Hansen, editors, *Modifying Adjuncts*, pages 33–66. De Gruyter Mouton, Berlin, Boston.

Chung-Hye Han, Na-Rae Han, Eon-Suk Ko, Martha Palmer, and Heejong Yi. 2002. Penn Korean Treebank: Development and Evaluation. In *Proceedings of the 16th Pacific Asia Conference on Language, Information and Computation*, pages 69–78, Jeju, Korea. Pacific Asia Conference on Language, Information and Computation.

David Jurgens and Ioannis Klapaftis. 2013. SemEval-2013 task 13: Word sense induction for graded and non-graded senses. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 290–299, Atlanta, Georgia, USA. Association for Computational Linguistics.

Jong-Bok Kim and Peter Sells. 2010. Oblique case marking on core arguments in Korean. *Studies in Language. International Journal sponsored by the Foundation "Foundations of Language"*, 34(3):602–635.

Young-Bum Kim, Heemoon Chae, Benjamin Snyder, and Yu-Seop Kim. 2014. Training a Korean SRL System with Rich Morphological Features. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 637–642, Baltimore, Maryland. Association for Computational Linguistics.

Young-hee Kim. 2004. A Criterion for Distinguishing between Arguments and Adjuncts in Korean. *HANGEUL*, 266(4):139–167.

Sangah Lee, Hansol Jang, Yunmee Baik, Suzi Park, and Hyopil Shin. 2020. Kr-bert: A small-scale korean-specific language model. *ArXiv*, abs/2008.03979.

Ryan McDonald, Joakim Nivre, Yvonne Quirmbach-Brundage, Yoav Goldberg, Dipanjan Das, Kuzman Ganchev, Keith Hall, Slav Petrov, Hao Zhang, Oscar Täckström, Claudia Bedini, Núria Bertomeu Castelló, and Jungmee Lee. 2013. Universal Dependency Annotation for Multilingual Parsing. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 92–97, Sofia, Bulgaria. Association for Computational Linguistics.

Chulwoo Park. 2002. Criteria for Distinguishing Complements and Adjuncts in Korean. *Korean Journal of Linguistics (EONEOHAG)*, 34(3):75–111.

Jungyeul Park and Mija Kim. 2023. A role of functional morphemes in Korean categorial grammars. *Korean Linguistics*, 19(1):1–30.

Sungjoon Park, Jihyung Moon, Sungdong Kim, Won Ik Cho, Ji Yoon Han, Jangwon Park, Chisung Song, Junseong Kim, Youngsook Song, Taehwan Oh, Joohong Lee, Juhyun Oh, Sungwon Lyu, Younghoon Jeong, Inkwon Lee, Sangwoo Seo, Dongjun Lee, Hyunwoo Kim, Myeonghwa Lee, Seongbo Jang, Seungwon Do, Sunkyoung Kim, Kyungtae Lim, Jongwon Lee, Kyumin Park, Jamin Shin, Seonghyun Kim, Lucy Park, Alice Oh, Jung-Woo Ha, and Kyunghyun Cho. 2021. KLUE: Korean Language Understanding Evaluation. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1, pages 1–25. Curran.

Carl Pollard and Ivan A. Sag. 1987. *Information-Based Syntax and Semantics, Vol. 1: Fundamentals*. The Center for the Study of Language and Information Publications, Stanford, California, USA.

Carl Pollard and Ivan A. Sag. 1994. *Head-Driven Phrase Structure Grammar*. The University of Chicago Press, Chicago, Illinois, USA.

Ivan A. Sag, Thomas Wasow, and Emily M. Bender. 2003. *Syntactic Theory: A Formal Introduction*, 2nd

editio edition. CSLI Lecture Notes. The University of Chicago Press, Chicago, IL, USA.

Saetbyol Seo, Myeong-ju Kim, YeonSook Sung, and Seong Hee Yoo. 2019. A Proposal on Universal Dependencies (v.2) Annotation for Korean. *Language and Information*, 23(1):91–122.

A Data Source Illustrations

A.1 The KLUE Dependency Treebank

The KLUE dependency treebank[8] in KLUE benchmark (Park et al., 2021) follows the Sejong-style dependency structure. The following dependency tree illustrates the typical structure of data in the KLUE dependency treebank.

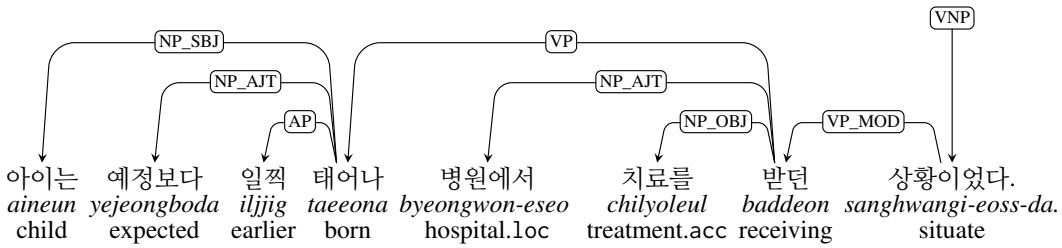

Figure 2: An example of the data in the KLUE dependency treebank (sentence meaning: 'The child was born earlier than expected and was receiving treatment in the hospital').

NP_SBJ, NP_AJT and NP_OBJ represent a nominal subject, an adjunct, and an object, respectively. Instead of utilizing the root tag uniformly, the treebank indicates the property of the root, such as VNP (copular). The last token *sanghwangi-eoss-da.* ('situate') includes the punctuation mark. More importantly, the dependency structure consistently adheres to a right-to-left pattern, affirming that Korean is a head-final language.

A.2 The Sejong Verb Dictionary

The Sejong dictionary is part of the Sejong corpus organized by the National Institute of Korean Language.[9] We are focusing on the verbs in the dictionary, which are sorted such that each verbal lexeme has a separate entry. These entries include the syntactic and semantic information of the verbs for each sense. The possible subcategorization frames and semantic roles of the arguments are provided, along with example sentences. Figure 3 shows an example of the lexeme *jangsighada* while the sense included is 'to decorate'. The XML file also includes other senses of the lexeme, as well as other lexemes with the same surface form.

[8] https://klue-benchmark.com/tasks/71/data/description
[9] https://korean.go.kr

```
<orth>장식하다</orth>
<entry pos="vv">
    <morph_grp>
        <var type="spr">장식을 하다</var>
        <str>N.하</str>
        <org lg="si">裝飾</org>
    </morph_grp>
    <sense>
        <sem_grp>
            <sem_class>결과행위</sem_class>
            <trans>decorate</trans>
        </sem_grp>
        <frame_grp type="FTR">
            <frame>X=N0-이 Y=N1-을 Z=N2-로 V</frame>
            <subsense>
                <sel_rst arg="X" tht="AGT">인간</sel_rst>
                <sel_rst arg="Y" tht="THM">추상적대상</sel_rst>
                <sel_rst arg="Z" tht="INS">추상적대상</sel_rst>
                <eg>작가는 드라마의 종말을 해피엔딩으로 장식했다.</eg>
                <eg>그는 자신의 연설을 화려한 미사여구로 장식하기를 좋아했다.</eg>
            </subsense>
        </frame_grp>
    </sense>
</entry>
```

Figure 3: Example partly taken from the entry of the lexeme *jangsighada* in the Sejong dictionary whose sense is 'to decorate'.

B Illustrative Samples of Proposed Data

Figures 4 and 5 are examples from our proposed dataset converted from sentences in KLUE's treebank.

match: AllFrameArgMatch
target: 장식하다
frame: X=N0-이 Y=N1-을 Z=N2-로 V
text: 앞서 부전승으로 16강에 오른 김태훈은 준결승까지 세 경기를 모두 점수 차 승리로 장식하며 정상을 향해 나아갔다.

ID	FORM	LEMMA	XPOS	HEAD	DEPREL	BIO	MISC
1	앞서	앞서	MAG	4	AP	B-COMP	*apseo* ('earlier')
2	부전승으로	부전+승+으로	NNG+NNG+JKB	4	NP_AJT	I-COMP	*bujeonseung-eulo* ('with a bye')
3	16강에	16+강+에	SN+NNG+JKB	4	NP_AJT	I-COMP	*16gang-e* ('to the round of 16')
4	오른	오르+ㄴ	VV+ETM	5	VP_MOD	I-COMP	*oleun* ('advanced')
5	김태훈은	김태훈+은	NNP+JX	13	NP_SBJ	I-COMP	*gimtaehun-eun* ('Kim Tae-hoon')
6	준결승까지	준+결승+까지	XPN+NNG+JX	13	NP_AJT	B-AJT	*jungyeolseungkkaji* ('to the semifinals')
7	세	세	MMN	8	DP	B-COMP	*se* ('three')
8	경기를	경기+를	NNG+JKO	13	NP_OBJ	I-COMP	*gyeonggi-leul* ('matches')
9	모두	모두	MAG	13	AP	B-ADV	*modu* ('all')
10	점수	점수	NNG	11	NP	B-COMP	*jeomsu* ('score')
11	차	차	NNG	12	NP	I-COMP	*cha* ('margin')
12	승리로	승리+로	NNG+JKB	13	NP_AJT	I-COMP	*seunglilo* ('with victories')
13	장식하며	장식+하+며	NNG+XSV+EC	15	VP	TARGET	*jangsighamyeo* ('decorating')
14	정상을	정상+을	NNG+JKO	15	NP_OBJ	O	*jeongsang-eul* ('the top')
15	향해	향하+여	VV+EC	16	VP	O	*hyanghae* ('towards')
16	나아갔다.	나+아+가+았+다+.	VV+EC+VX+EP+EF+SF	0	VP	O	*naagassda* ('moved forward')

Figure 4: Converted instance of an example sentence *apseo bujeonseung-eulo 16gang-e oleun gimtaehun-eun jungyeolseungkkaji se gyeong-gileul modu jeomsu cha seunglilo jangsighamyeo jeongsang-eul hyanghae naagassda* ('Kim Tae-hoon, who advanced to the round of 16 with a bye, moved towards the championship by winning all three of his matches leading up to the semifinals with decisive score margins'), with the target *jangsighamyeo* conjugated from the lexeme *jangsighada* ('to decorate') as in Figure 3.

match: AllFrameArgMatch
target: 돌파하다
frame: X=N0-이 Y=N1-을 V
text: 현재 이 영상은 유튜브에서 누적 조회수 150만 건을 돌파하며 인기를 얻고 있다.

ID	FORM	LEMMA	XPOS	HEAD	DEPREL	BIO	MISC
1	현재	현재	MAG	11	AP	O	*hyeonjae* ('currently')
2	이	이	MMD	3	DP	B-COMP	*i* ('this')
3	영상은	영상+은	NNG+JX	9	NP_SBJ	I-COMP	*yeongsang-eun* ('video')
4	유튜브에서	유튜브+에서	NNP+JKB	9	NP_AJT	B-AJT	*yutyubeu-eseo* ('on YouTube')
5	누적	누적	NNG	6	NP	B-COMP	*nujeog* ('cumulative')
6	조회수	조회수	NNG	8	NP	I-COMP	*johoesu* ('views')
7	150만	150+만	SN+NR	8	NP	I-COMP	*150man* ('1.5 million')
8	건을	건+을	NNB+JKO	9	NP_OBJ	I-COMP	*geon-eul* ('cases')
9	돌파하며	돌파+하+며	NNG+XSV+EC	11	VP	TARGET	*dolpahamyeo* ('exceeding')
10	인기를	인기+를	NNG+JKO	11	NP_OBJ	O	*ingi-leul* ('popularity')
11	얻고	얻+고	VV+EC	12	VP	O	*eodgo* ('gaining')
12	있다.	있+다+.	VX+EF+SF	0	VP	O	*issda* ('be')

Figure 5: Converted instance of an example sentence *hyeonjae i yeongsang-eun yutyubeueseo nujeog johoesu 150man geon-eul dolpahamyeo ingileul eodgo issda* ('Currently, this video is gaining popularity with over 1.5 million cumulative views on YouTube'), with the target *dolpahamyeo* conjugated from the lexeme *dolpahada*.

C Experiments and Results

To validate the quality of the constructed dataset, we fine-tune several pre-trained models for sequence labeling using the sentences from the training set of KLUE's dependency treebank. Given a verbal predicate in a sentence, the sequence labeling models tag its complements and adjuncts. Our classification is over 8 classes: `O`, `TARGET`, `B-COMP`, `I-COMP`, `B-AJT`, `I-AJT`, `B-ADV`, `I-ADV`, following the BIO tagging scheme.[10] Since the official test set of KLUE's dependency treebank is not publicly available, we split the training set into two subsets of equal size, and conduct 2-fold cross-validation with one used as the test set and the other further split into a training set and a development set.

The sequence labeling models are based on the pre-trained encoder-only models, including Korean monolingual `KoELECTRA-Base-v3` discriminator model[11], `KLUE-BERT-base` model (Park et al., 2021)[12] and `KR-BERT-char16424` model (Lee et al., 2020)[13], as well as a multilingual `XLM-RoBERTa-base` model (Conneau et al., 2020). They are fine-tuned for the complement/adjunct detection task using our proposed dataset. The hyperparameter settings are presented in Table 3.

	KoELECTRA-Base-v3	KLUE-BERT-base	KR-BERT-char16424	XLM-RoBERTa-base
Exact F_1	0.7631 ± 0.0049	0.7762 ± 0.0092	0.7427 ± 0.0059	0.7634 ± 0.0024
Partial F_1	0.8012 ± 0.0057	0.8125 ± 0.0097	0.7850 ± 0.0079	0.8144 ± 0.0034

Table 2: The cross validation mean $\pm$ standard deviation of exact and partial F_1 scores on the constructed dataset.

To evaluate model performance, we follow the measurements suggested in SemEval'13 (Jurgens and Klapaftis, 2013). Specifically, we use the exact F_1 score to select the best epoch out of the 6 training epochs. Table 2 shows the exact and partial F_1 scores of the fine-tuned models on the dataset using 2-fold cross-validation. All models obtain satisfactory results, showcasing that our proposed dataset is trainable and reliable.

[10] `COMP` denotes complements, `AJT` denotes adjuncts, and `ADV` denotes elements that are not argument candidates, such as adverbs.

[11] https://github.com/monologg/KoELECTRA

[12] https://github.com/KLUE-benchmark/KLUE

[13] https://github.com/snunlp/KR-BERT

Epochs	6
Learning Rate	5e-5
Batch Size (train)	128
Batch Size (eval)	256
Evaluation Strategy	epoch

Table 3: Hyperparameter settings of the models during fine-tuning.