symmetry

MDPI

Article

A Transfer Learning-Based Pairwise Information Extraction Framework Using BERT and Korean-Language Modification Relationships

Hanjo Jeong

Department of Software Convergence Engineering, Mokpo National University, Muan-gun 58554, Republic of Korea; hanjojeong@mnu.ac.kr; Tel.: +82-61-450-2773

Abstract: Most named entity recognition approaches employing BERT-based transfer learning focus solely on extracting independent and simple tags, neglecting the sequence and dependency features inherent in the named-entity tags. Consequently, these basic BERT-based methods fall short in domains requiring the extraction of more intricate information, such as the detailed characteristics of products, services, and places from user reviews. In this paper, we introduce an end-to-end information extraction framework comprising three key components: (1) a tagging scheme that effectively represents detailed characteristics; (2) a BERT-based transfer learning model designed for extracting named-entity tags, utilizing both general linguistic features learned from a large corpus and the sequence and symmetric-dependency features of the named-entity tags; and (3) a pairwise information extraction algorithm that pairs features with their corresponding symmetric modifying words to extract detailed information.

Keywords: transfer learning; BERT; information extraction; review analysis; Korean natural language

check for updates

Citation: Jeong, H. A Transfer Learning-Based Pairwise Information Extraction Framework Using BERT and Korean-Language Modification Relationships. *Symmetry* **2024**, *16*, 136. https://doi.org/10.3390/sym16020136

Academic Editor: Tomohiro Inagaki

Received: 26 December 2023
Revised: 15 January 2024
Accepted: 22 January 2024
Published: 23 January 2024

Copyright: © 2024 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Most of the transfer learning-based models for named-entity recognition (NER) in various fields use the BERT (Bidirectional Encoder Representations from Transformers) model [1–5]. This is due to its ability to capture contextual information bidirectionally (from both left and right contexts), which enables it to comprehend the meaning of sub-words/words within a sentence based on their context [6,7]. Transfer learning approaches can be categorized into fine-tuning and feature-based approaches [6]. The fine-tuning approaches simply add an additional layer to the final layer of the BERT learning framework and conduct fine-tuning across the entire network, utilizing only the BERT encoders. On the other hand, feature-based approaches employ both encoder and decoder layers of attention-based transformer models [7]. Thus, the hidden vectors from either the penultimate layer or the last four layers of the BERT-based pre-training network are used as only the input features for one or more decoder layers of the transformer network without fine-tuning any parameters of BERT.

In the field of NER, most transfer-learning-based approaches employ fine-tuning models by adding an extra layer consisting of a bidirectional Long Short-Term Memory (BiLSTM) and a conditional random field (CRF) layer to classify tokens within bi-directional contexts (left and right tokens) of each token [8–11]. Feature-based approaches that use BERT solely for pre-training are less common in NER downstream tasks, being primarily applied in neural machine translation (NMT) [12–14]. This is attributed to the lesser importance of the sequence of named entities compared to the sequence of sub-word tokens, which are crucial for generating sentences in a target language in NMT tasks.

However, the sequence and dependencies of named entities might be beneficial for classification tasks in domains that require adherence to specific sequencing and dependence rules. For instance, in the domain of feature extraction from tourist attractions

using user reviews on TripAdvisor, the sequence and dependence rules of named entities might be crucial for NER tasks, particularly for those named entities that represent features and their modifying words. For NER tasks requiring such consideration, BERT-based fine-tuning models using composite embeddings of subwords and named entities from the pre-training stages are introduced [15,16]. These models outperform the original BERT models that use only subword embeddings, as they learn the sequence and relationship features of the named entities, along with general linguistic features. Nonetheless, their effectiveness compared to the original fine-tuning models is not guaranteed in general because of the difficulty of constructing the named-entity-tagged data.

In this research, we introduce a tagging scheme designed to represent modification and dependency relationships in Korean natural language, applicable to the automatic extraction of information such as the characteristics of products, restaurants, tourist attractions, and more. The named-entity tags defined in this scheme adhere to certain implicit rules in their sequencing and symmetric dependence, reflecting the nature of the modification relationships, where features are symmetrically dependent on their modifying words. Therefore, considering and learning the sequence and dependencies of these tags, along with general linguistic features, is beneficial. To utilize these named-entity tag features for training and prediction in the circumstance of limited tagged training data, we adopt a feature-based approach. This involves using BERT for pre-training and then inputting the pre-trained contextual embeddings into transformer-based decoders, along with embeddings that represent the tag sequence at the same subword-token-wise level. These decoders also process the embeddings of previously predicted named-entity tags and classify the tags based on both the pre-trained BERT features and the sequence and symmetric-dependency features of the named-entity tags. Furthermore, they utilize all previously predicted tags to classify subsequent tags in a sentence in the same manner as decoders in the transformer network. Additionally, this paper introduces a modification-relationship extraction algorithm that automatically extracts modification information based on the recognized named entities using the aforementioned transfer-learning framework.

2. Materials and Methods

2.1. Data Collection and Tagging

2.1.1. Data Collection

We collected about 7000 review data, which is composed of about 20,000 sentences, from TripAdvisor. The review data is equally selected for famous travel destinations and tourist attractions in 7 major regions in Korea including Seoul, Gyeonggi-do, Gangwon-do, etc.

2.1.2. Data Tagging

The BIO (B: Beginning, I: Inside, O: Outside) tagging framework is used to tag sentences with vocabulary that represents features and words symmetrically modifying those features, such as nouns (feature names), predicates (descriptive/emotional vocabulary), negation words, and emphasized words (weighted vocabulary). To tag the sentences, the collected review data is first separated into individual sentences.

To learn and extract symmetric modification relationships from natural language sentences using a deep learning algorithm, various types of modification relationships were defined in our previous work [17], as shown in Table 1. In Korean sentences, modification relationships can be classified into two types based on word order: prepositional modification and postpositional modification. Prepositional modification refers to cases where predicate or adjective words, located before their modifying words, modify nouns. Conversely, in postpositional modification types, the predicate words are located after their modifying words. Negation in modification phrases is similarly divided into prepositional and postpositional negation words, which denote the negation of predicate words before and after them, respectively. Similarly, emphasis words that emphasize the predicate words are divided into prepositional and postpositional types.

Table 1. Word-order types of modification phrases and sentences in Korean.

Modification Relationship	Type of Word Order	Example Phrase or Sentence
Prepositional modification	(Predicate, Noun)	아름다운_beautiful 야경_night view
Prepositional modification and Prepositional negation	(Negation, Predicate, Noun)	안_not 예쁜_beautiful 야경_night view
Prepositional modification and Postpositional negation	(Predicate, Negation, Noun)	아름답지_beautiful 않은_not 야경_night view
Postpositional modification	(Noun, Predicate)	야경_night view- 이_is 아름답습니다_beautiful
Postpositional modification and Prepositional negation	(Noun, Negation, Predicate)	야경_night view- 이_is 안_not 예쁩니다_beautiful
Postpositional modification and Postpositional negation	(Noun, Predicate, Negation)	야경_night view- 이_is 아름답지_beautiful 않습니다_not

A sentence may contain more than one type of modification relationship. Therefore, to determine which predicate words modify which nouns (features) and which negation or emphasis words modify which predicate words, the relationships are categorized into these two types based on word order. This modification tagging framework enables the accurate extraction of modification relationships among various combinations of nouns, predicates, negation words, and emphasis words.

Table 2 shows a list of tags used to represent all modification types described in Table 1 within Korean sentences. Predicate words, negation words, and emphasis words are represented using prepositional and postpositional tags. These tags are assigned based on their modifying words and word orders, reflecting the symmetric nature of the modification relationships. Figure 1 shows an example of a Korean sentence where multiple modification relationships are tagged using the definitions provided in Table 2. Through the use of these prepositional and postpositional tags, we can accurately extract modification relationships related to features such as 'price' and 'night view.' For instance, the word 'expensive' is tagged as a postpositional predicate word, indicating its role in modifying the preceding 'price' feature. Similarly, the word 'too' is tagged as a prepositional emphasis word, signifying its modification of the subsequent 'expensive' word.

Table 2. Tag list for the modification types in Korean sentences.

Tag	Description
F	Noun word/s representing the name of feature
PB	Predicate word/s modifying feature comes Before F
PA	Predicate word/s modifying feature comes After F
NB	Negation word/s comes Before PB or PA
NA	Negation word/s comes After PB or PA
EB	Emphasis words comes Before PB or PA
EA	Emphasis words comes After PB or PA

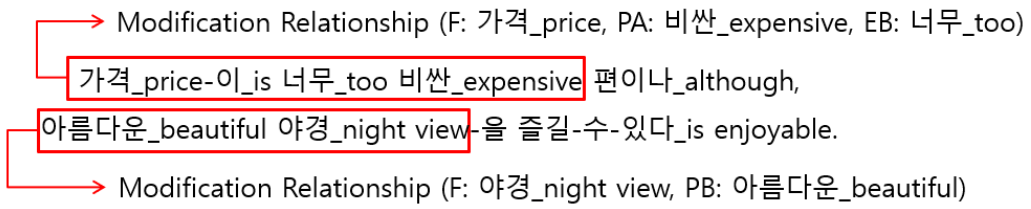


Figure 1. An example sentence having multiple modification relationships.

For the final data tagging in the training and test data of the transfer learning model, we created B (Begin) and I (Inside) tags for each category defined in Table 2, along with an O (Outside) tag for non-named entity words. These tags are as follows: B-F, I-F, B-PB, I-PB, B-PA, I-PA, B-NB, I-NB, B-NA, I-NA, B-EB, I-EB, B-EA, I-EA, O. This tagging system aids in

precisely recognizing the relationships among various combinations of nouns, predicates, negation words, and emphasis words.

2.2. Transfer Learning-Based Pairwise Information Extraction Framework

The pairwise information extraction framework presented in this paper is based on transfer learning and is designed to learn and extract the tags that express the modification relationships defined in Section 2.1. To accomplish this, a named entity recognition algorithm based on transfer learning is employed in the framework to learn and extract the tags as named entities. Additionally, a modification extraction algorithm is introduced in the framework to extract the modification relationships based on the named-entity tags that were extracted.

2.2.1. BERT-Based Transfer Learning Model for the Named Entity Recognition Algorithm of Modification Relationships

Figure 2 shows our proposed BERT-based transfer learning model. A special [CLS] token is added at the beginning of the sentence. B-PB and B-F tokens are examples of the named entity tags generated in our decoder according to the sub-word tokens $s_1, s_2, \ldots s_n$. Due to the difficulty of constructing the named-entity-tagged training data, a feature-based approach using BERT is used. Our transfer learning model uses a BERT-based pre-trained model to obtain feature vectors representing the general linguistic features, and the pre-trained feature vectors are fed into the transformer-based decoder to learn and predict the named entities together with some named entity-tagged training data.

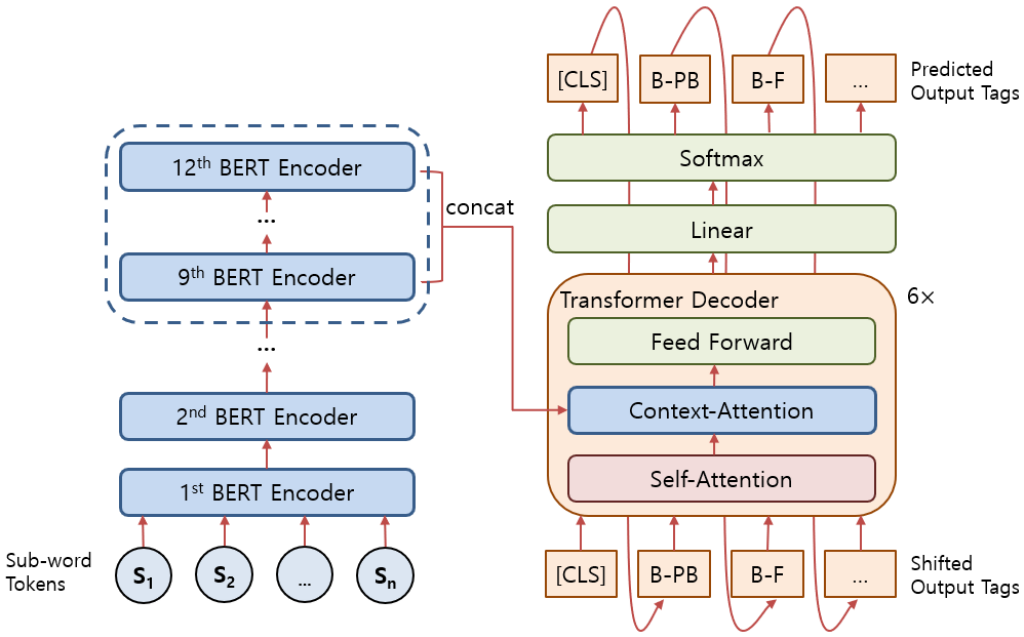


Figure 2. BERT-based transfer learning model.

To maximize the benefits of pre-training, our transfer learning model employs embeddings created by averaging the attention-value vectors from the last four hidden layers of the pre-trained BERT model. This approach has proven to be the most effective among various feature-based methods [6]. Subsequently, we feed these pre-trained contextual embeddings to a decoder. This decoder generates context attention, in conjunction with self-attentions, for training output-tag embeddings, similar to the decoder layers in the transformer network [7]. The TripAdvisor review data, tagged using the scheme introduced in Section 2.1.2, is utilized for training the decoder.

Figure 3 presents an example of the sub-word token sequence for a sentence and its corresponding label sequence, both of which are used as inputs for training the decoder layers. Subsequent to the fine-tuning of the decoder layers, the final results of the NER,

particularly for the modification-relationship tags, are derived from the predictions of the BIO-tags, as shown in Figure 4. The '[SEP]' token serves as a separator to distinguish between two sentences, similar to its usage in the original BERT paper [7]. However, our model uses only one sentence for each encoder and decoder: a sub-word token sequence for the encoder and its associated label sequence for the decoder. Our model uses word-piece embeddings [18], just as the pre-trained Korean BERT model uses word-piece embeddings. Thus, we generate embeddings for sequences of input word tokens and entity tokens, both of the same length, n. The sub-word tokens represent the word-piece tokens derived from the input words, while the entity tokens are generated for BIO tags corresponding to these word-piece tokens.

['[CLS]', '길_road', '##에_on', '쓰레기_trash', '하나_one', '없이_no', '너무_very', '깨끗_clean', '##했고_was', '주차장_parking lot', '##도_also', '잘_well', '되어_equipped', '있어서 _because', '구경_sightseeing', '##하기에_for', '좋았_good', '##답니다_was', '[SEP]', '[PAD]', ... , '[PAD]']

['[CLS]', 'B-F', 'O', 'O', 'O', 'O', 'B-EB', 'B-PA', 'O', 'B-F', 'O', 'B-PA', 'I-PA', 'O', 'O', 'O', 'O', 'O', '[SEP]', '[PAD]', ... , '[PAD]']

Figure 3. An example of tagged sentence with the BIO tags.

[['길_road', 'F'], ['##에_on', 'O'], ['쓰레기_trash', 'O'], ['하나_one', 'O'], ['없이_no', 'O'], ['너무_very', 'EB'], ['깨끗_clean', 'PA'], ['##했고_was', 'O'], ['주차장_parking lot', 'F'], ['##도 _also', 'O'], ['잘 되어_well equipped', '<PA>'], ['있어서_because', 'O'], ['구경_sightseeing', 'O'], [##하기에_for', 'O'], ['좋았_good', 'O'], ['##답니다_was', 'O']]

Figure 4. An example of the result of the NER prediction.

Formally, given a sequence of n sub-word (word-piece) tokens $s_1, s_2, \ldots s_n$ and n BIO-entity tokens $e_1, e_2, \ldots e_n$, our encoder and decoder compute the sequence of continuous representations by the attention mechanism defined at the transformer model [6]. Based on the continuous representations, our model generates an output sequence $e_1, e_2, \ldots e_n$ of BIO-entity tokens, producing one prediction at a time. The decoder operates in an auto-regressive manner [19], using the previously predicted BIO-entity tokens to predict the subsequent BIO-entity tokens. Equation (1) formalizes the maximum likelihood objective used to train in our model.

$$P(e_t|s_{1:n};\theta) = \prod_{t=1}^{n} P(e_t|e_{0:t-1}, s_{1:n};\theta) \tag{1}$$

Given a sequence of input vectors $s_1, s_2, \ldots s_n$, each of the output vectors $e_1, e_2, \ldots e_n$ is computed based on the averaged vector of the pretrained contextual vectors and previously predicted BIO-entity token embeddings. The Transformer encoder processes an input embedding matrix $H \in \mathbb{R}^{l \times d}$, where l represents the sequence length and d represents the input dimension. The self-attention value vector in the encoder, denoted as $Attn_{enc}$ is computed using Equation (2) through the scaled dot-product attention.

$$\begin{aligned} Q, K, V &= HW_q, HW_k, HW_v \\ Attn_{enc} &= softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \end{aligned} \tag{2}$$

where W_q, W_k, W_v represent learnable matrices corresponding to the hidden state matrices Q (Query), K (Key), and V (Value), respectively. The pretrained contextual vector in the

encoder, denoted as Avg_Attn_{enc} is then generated by averaging the attention vectors from the 9th to 12th layers, as described in Equation (3).

$$Avg_Attn_{enc} = \frac{Attn_{enc}^{(9)} + \cdots + Attn_{enc}^{(12)}}{4} \tag{3}$$

Then, Avg_Attn_{enc} is fed into the decoder to compute the encoder-decoder cross-attention vector, in conjunction with the unidirectional self-attention vector, denoted as $Self_Attn_{dec}$, which is generated in an auto-regressive manner using the previously predicted output embedding in the decoder. Equation (4) represents the attention computation for the encoder-decoder cross-attention vector, employing $Self_Attn_{dec}$ as the Q vector and Avg_Attn_{enc} as both the K and V vectors in the decoder.

$$Attn_{dec} = softmax\left(\frac{Self_Attn_{dec} \cdot Avg_Attn_{enc}}{\sqrt{d_k}}\right) \cdot Avg_Attn_{enc} \tag{4}$$

2.2.2. Modification-Relationships Extraction Algorithm

Figure 5 shows an algorithm for extracting modification relationships, which determines which predicate words actually modify which features, or which negation or emphasis words modify which predicates, from the sequence of vocabulary tokens and named-entity tokens obtained through the BERT-based NER algorithm described in Section 2.2.1. In order to extract the exact pair of modification relationships from the sequence of positionally named entity tokens, the following algorithm is used:

```
feature_list ← Empty
predicate_list ← Empty
negate_word_list ← Empty
emphasis_word_list ← Empty
FUNCTION extract_modification_pair_list (token_label_pair_seq)
    feature_idx ← 0
    predicate_idx ← 0
    FOR EACH token_label_pair in token_label_pair_seq
        IF label is F
            feature_list[feature_idx] ← token_label_pair
            feature_idx ← feature_idx + 1
        IF label is PB
            predicate_list[feature_idx] ← token_label_pair
            predicate_idx ← predicate_idx + 1
        IF label is PA
            predicate_list[feature_idx 1] ← token_label_pair
            predicate_idx ← predicate_idx + 1
        IF label is NB
            negate_word_list[predicate_idx] ← token_label_pair
        IF label is NA
            negate_word_list[predicate_idx − 1] ← token_label_pair
        IF label is EB
            emphasis_word_list[predicate_idx ] ← token_label_pair
        IF label is EA
            emphasis_word_list[predicate_idx − 1] ← token_label_pair
    END FOR
END FUNCTION
```

Figure 5. Modification-relationships extraction algorithm.

1. If a new feature is found while running the for-loop for each vocabulary token and named-entity token pair list in a sentence, the found feature is added to the feature

list first, and the index of the feature list is increased by one for the next pair of modification relationships;
2. If a PB tag, which represents the prepositional predicate, is found, the corresponding vocabulary and named-entity tokens are added to the predicate list with the current feature index;
3. In the case of a PA tag, which represents the postpositional predicate, the index of the predicate list is set to -1 at the current feature index since it modifies the previously located feature;
4. Negation words and emphasis words that modify the predicate are similarly stored in the negation word list and the emphasis word list using the current predicate index in the case of prepositional tags and the previous predicate index in the case of postpositional tags;
5. The algorithm finally returns the predicate list and its modifying feature list with identical indices if they are in the modifying relationships. Additionally, it returns negation word and emphasis word lists with identical indices to their modifying predicate in the predicate list.

3. Experiments and Results

3.1. Experimental Environments and Metrics

In the experiments and validation of our transfer learning-based named entity recognition algorithm for modification relationships, we employed Huggingface's BERT wordpiece tokenizer [20,21] along with the 'kcbert-large' and 'koBERT' pre-trained models [22] available both on the HuggingFace Model Hub [23]. The 'kcbert-large' model, which is pre-trained on a corpus of 110 million comments from Naver News [24], the largest news portal in Korea, is well-suited for extracting information from TripAdvisor review data, as it was trained on actual user comments. The 'koBERT' model, pretrained on approximately 50 million sentences from Korean Wikipedia text, represents written language and thus differs slightly from the colloquial user review data. It is utilized to validate our proposed model across various pretrained models. For training the decoder layers, the tagged TripAdvisor data were randomly split into a training dataset (70%) and a test dataset (30%).

We used precision, recall, and *F*1-score as evaluation metrics in our experiments. Precision is calculated as the ratio of the number of correctly recognized modification relationship tags to the total number of predicted tags. Recall is the ratio of the number of correctly recognized tags to the total number of actual true tags within the test dataset. The *F*1-score is the geometric mean of *precision* and *recall*, as shown in Equation (5).

$$\begin{aligned} Precision &= \frac{\#\ of\ correctly\ predicted\ tags}{\#\ of\ predicted\ tags} \\ Recall &= \frac{\#\ of\ correctly\ predicted\ tags}{\#\ of\ true\ tags} \\ F1 &= \frac{2 \times Precision \times Recall}{Precision + Recall} \end{aligned} \tag{5}$$

3.2. Experimental Results and Discussion

To validate our proposed transfer learning model, we compared its performance with that of the standard fine-tuning model and a domain-specific model. The domain-specific model relies exclusively on labeled training data and does not utilize a pre-training based on a large-scaled unlabeled corpus. Both the fine-tuning model and our proposed model employed 'kcbert-large' and 'koBERT' models, which were pre-trained on a large-scale corpus. The domain-specific model, however, was trained solely on our labeled TripAdvisor review dataset. As shown in Table 3, our proposed model achieved the highest precision, demonstrating an approximate 5% improvement over the simple fine-tuning model based on both the koBERT and kcbert-large pretrained models, and a roughly 4% enhancement compared to the domain-specific model.

Table 3. Performance of the transfer learning-based NER Algorithm for the modification-relationships tags.

Model	Precision	Recall	*F*1-Score
Domain-specific Model	82.1	79.5	80.8
koBERT based			
Fine-tuning Model	79.2	81.2	80.2
Our proposed Model	83.1	80.7	81.9
kcbert-large based			
Fine-tuning Model	81.2	81.8	81.5
Our proposed Model	85.3	81.1	83.1

Our proposed model achieved the highest precision, as shown in Table 3. This is because it can leverage both the general linguistic features learned from the large corpus and the sequence and symmetric-dependency features of the named-entity tags. We specifically tagged features that have predicates for travel destinations and attractions within a sentence, as only features with predicates are valid for their representation. Additionally, the subword tokens used to represent predicates can vary based on their relative positions and the positions of the features they modify. A simple fine-tuning model cannot learn these types of dependency and sequence features; hence, our proposed model significantly improves precision.

The recall is slightly lower compared to the fine-tuning model, which might be due to our model applying sequence and dependency rules more strictly. However, the decrease in recall is not substantial compared to the domain-specific model, as it is not pre-trained on a large-scaled corpus.

4. Conclusions

In this paper, we presented an end-to-end transfer learning-based framework for pairwise information extraction, focusing mainly on extracting characteristic information about products, services, and places from user reviews. Initially, we introduced a tagging scheme for applying the Named Entity Recognition (NER) task to extract such information, based on the BIO tagging scheme and considering the linguistic features of Korean natural language.

We also presented a transfer learning-based model using BERT in this paper. This model capitalizes on the strengths of BERT by applying general linguistic features pre-trained on a large-scale corpus, and it also utilizes the sequence and symmetric-dependency features inherent in the named entities. Therefore, we do not compromise performance due to either the lack of utilization of pre-training on a large-scale unsupervised corpus or the failure to utilize the innate features of the named-entity tags. The experiments and results demonstrated that our proposed model is more effective in domains where obtaining large amounts of tagged training data is generally challenging. However, in situations where large amounts of labeled training data are available, the domain-specific models might achieve better performance. This is because they train sub-word tokens and their associated entity tokens together from the pre-training stage.

Additionally, this paper presents a pairwise information extraction algorithm. With this algorithm, we can precisely extract information by pairing features with their modifying word(s). This entire framework is applicable across various domains for extracting information from user reviews and comments in Korean natural language.

Funding: This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2021-0058).

Data Availability Statement: Data are contained within the article.

Acknowledgments: The author expresses the acknowledgment to National Research Foundation of Korea (NRF) for the funding, Korea government (MSIT) (No. 2021-0058).

Conflicts of Interest: The author declares no conflicts of interest.

References

1. Tikayat Ray, A.; Fischer, O.J.; Mavris, D.N.; White, R.T.; Cole, B.F. aeroBERT-NER: Named-Entity Recognition for Aerospace Requirements Engineering using BERT. In Proceedings of the AIAA SCITECH 2023 Forum, National Harbor, MD, USA, 23–27 January 2023; p. 2583.
2. Zhang, Y.; Zhang, H. FinBERT–MRC: Financial Named Entity Recognition Using BERT Under the Machine Reading Comprehension Paradigm. *Neural Process. Lett.* **2023**, *55*, 7393–7413. [CrossRef]
3. Lv, X.; Xie, Z.; Xu, D.; Jin, X.; Ma, K.; Tao, L.; Qiu, Q.; Pan, Y. Chinese named entity recognition in the geoscience domain based on bert. *Earth Space Sci.* **2022**, *9*, e2021EA002166. [CrossRef]
4. Akhtyamova, L. Named entity recognition in Spanish biomedical literature: Short review and BERT model. In Proceedings of the 2020 26th Conference of Open Innovations Association (FRUCT), Yaroslavl, Russia, 20–24 April 2020; pp. 1–7.
5. Kim, Y.M.; Lee, T.H. Korean clinical entity recognition from diagnosis text using BERT. *BMC Med. Inform. Decis. Mak.* **2020**, *20*, 242. [CrossRef] [PubMed]
6. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is all you need. In Proceedings of the 31st International Conference on Neural Information Processing Systems, Red Hook, NY, USA, 4–9 December 2017; p. 30.
7. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv* **2018**, arXiv:1810.04805.
8. Syed, M.H.; Chung, S.T. MenuNER: Domain-adapted BERT based NER approach for a domain with limited dataset and its application to food menu domain. *Appl. Sci.* **2021**, *11*, 6007. [CrossRef]
9. Yang, R.; Gan, Y.; Zhang, C. Chinese Named Entity Recognition Based on BERT and Lightweight Feature Extraction Model. *Information* **2022**, *13*, 515. [CrossRef]
10. Agrawal, A.; Tripathi, S.; Vardhan, M.; Sihag, V.; Choudhary, G.; Dragoni, N. BERT-based transfer-learning approach for nested named-entity recognition using joint labeling. *Appl. Sci.* **2022**, *12*, 976. [CrossRef]
11. Li, W.; Du, Y.; Li, X.; Chen, X.; Xie, C.; Li, H.; Li, X. UD_BBC: Named entity recognition in social network combined BERT-BiLSTM-CRF with active learning. *Eng. Appl. Artif. Intell.* **2022**, *116*, 105460. [CrossRef]
12. Zhang, Z.; Wu, S.; Jiang, D.W.; Chen, G. BERT-JAM: Maximizing the utilization of BERT for neural machine translation. *Neurocomputing* **2021**, *460*, 84–94. [CrossRef]
13. Wu, X.; Xia, Y.; Zhu, J.; Wu, L.; Xie, S.; Qin, T. A study of BERT for context-aware neural machine translation. *Mach. Learn.* **2022**, *111*, 917–935. [CrossRef]
14. Yan, R.; Li, J.; Su, X.; Wang, X.; Gao, G. Boosting the Transformer with the BERT Supervision in Low-Resource Machine Translation. *Appl. Sci.* **2022**, *12*, 7195. [CrossRef]
15. Zhang, Z.; Han, X.; Liu, Z.; Jiang, X.; Sun, M.; Liu, Q. ERNIE: Enhanced language representation with informative entities. *arXiv* **2019**, arXiv:1905.07129.
16. Yamada, I.; Asai, A.; Shindo, H.; Takeda, H.; Matsumoto, Y. Luke: Deep contextualized entity representations with entity-aware self-attention. *arXiv* **2020**, arXiv:2010.01057.
17. Jeong, H.; Kwak, J.; Kim, J.; Jang, J.; Lee, H. A Study on Methods of Automatic Extraction of Korean-Language Modification Relationships for Sentiment analysis. In Proceedings of the 2020 International Conference on Artificial Intelligence in Information and Communication (ICAIIC), Fukuoka, Japan, 19–21 February 2020; pp. 544–546.
18. Wu, Y.; Schuster, M. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv* **2016**, arXiv:1609.08144.
19. Graves, A. Generating sequences with recurrent neural networks. *arXiv* **2013**, arXiv:1308.0850.
20. Wolf, T.; Debut, L.; Sanh, V.; Chaumond, J.; Delangue, C.; Moi, A.; Cistac, P.; Rault, T.; Louf, R.; Funtowicz, M.; et al. Huggingface's transformers: State-of-the-art natural language processing. *arXiv* **2019**, arXiv:1910.03771.
21. Huggingface Tokenizers: Fast State-of-the-Art Tokenizers optimized for Research and Production. Available online: https://github.com/huggingface/tokenizers (accessed on 26 December 2023).
22. Lee, J. Kcbert: Korean comments bert. In Proceedings of the Annual Conference on Human and Language Technology, Lisboa, Portugal, 3–5 November 2020; pp. 437–440.
23. Huggingface Model Hub. Available online: https://huggingface.co/models (accessed on 26 December 2023).
24. Naver News. Available online: https://news.naver.com/ (accessed on 26 December 2023).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.