

# 대화 시스템의 개체 생략 복원을 위한 유효 발화문 인식

소찬호<sup>†0</sup>, 왕지현<sup>‡</sup>, 이충희<sup>‡</sup>, 이연수<sup>‡</sup>, 강재우<sup>†\*</sup>

고려대학교<sup>†</sup>, ㈜엔씨소프트<sup>‡</sup>

chanhoso@korea.ac.kr, {KorJhwang, forever73, yeonsoo}@ncsoft.com, kangji@korea.ac.kr

## Valid Conversation Recognition for Restoring Entity Ellipsis in Chat Bot

Chan Ho So<sup>†0</sup>, Ji Hyun Wang<sup>‡</sup>, Chunghee Lee<sup>‡</sup>, Yeonsoo Lee<sup>‡</sup>, Jaewoo Kang<sup>†\*</sup>  
Korea University<sup>†</sup>, NCSOFT Corp.<sup>‡</sup>

### 요약

본 논문은 대화 시스템인 챗봇의 성능 향상을 위한 생략 복원 기술의 정확률을 올리기 위한 유효 발화문 인식 모델을 제안한다. 생략 복원 기술은 챗봇 사용자의 현재 발화문의 생략된 정보를 이전 발화문으로부터 복원하는 기술이다. 유효 발화문 인식 모델은 현재 발화문의 생략된 정보를 보유한 이전 발화문을 인식하는 역할을 수행한다. 유효 발화문 인식 모델은 BERT 기반 이전 분류 모델이며, 사용된 BERT 모델은 한국어 문서를 기반으로 새로 학습된 한국어 사전 학습 BERT 모델이다. 사용자의 현재 발화문과 이전 발화문들의 토큰 임베딩을 한국어 BERT를 통해 얻고, CNN 모델을 이용하여 각 토큰의 지역적인 정보를 추출해서 발화문 쌍의 표현 정보를 구해 해당 이전 발화문에 생략된 개체값이 있는지를 판단한다. 제안한 모델의 효과를 검증하기 위해 유효 발화문 인식 모델에서 유효하다고 판단한 이전 발화문만을 생략 복원 모델에 적용한 결과, 생략 복원 모델의 정확률이 약 5% 정도 상승한 것을 확인하였다.

주제어: 한국어 BERT, Text Classification, 생략 복원, 챗봇

### 1. 서론

야구는 국내에서 가장 많은 사람들이 즐기고 있는 프로 스포츠 중 하나로, 최근 ㈜엔씨소프트에서는 프로 야구 커뮤니티 겸 종합 서비스 어플 PAIGE를 개발하여 배포하였다. AI 야구 챗봇은 PAIGE의 주요 기능 중 하나로 많은 사용자들이 팀, 선수, 경기 기록 등 본인이 필요한 야구 관련 정보를 얻어 가고 있다. 이런 챗봇이 사용자가 만족하면서 지속적으로 서비스가 되려면 챗봇은 사람과의 연속적인 대화 안에서 사용자의 질의를 정확하게 파악하고 적절한 대답을 할 수 있어야한다. 그러나 현재 대부분의 챗봇들은 사용자가 원하는 대답을 주지 못하는 경우가 자주 발생한다. 이 상황의 대표적인 원인은 사용자의 발화문 안에서 ‘생략’이 이루어지기 때문이다. 예를 들어 A: “한화 이글스의 김진욱 선수가 제일 어려워요.” B: “정말? 몇 년생인데?” 라는 A, B의 발화를 보면 B의 발화에는 ‘김진욱 선수’ 라는 내용이 생략되어 있다. 사람과 사람 간의 대화에서는 이런 대화 문맥상 반복되는 정보는 생략하고 말을 한다. 같은 표현을 말하지 않아도 사람은 대화 문맥상 생략이 일어난 상대의 발화를 이해할 수 있고 알맞은 답변을 줄 수 있다. 하지만 일부 중요한 정보가 생략된다면 챗봇은 답변 모델에 입력으로 들어갈 정보가 부족해 적절한 답변을 줄 수 없을 가능성이 높다. 이를 방지하고자 사용자와 챗봇의 이전 발화문들을 통해서 현재 발화문에 생략된 개체를 복원하는 기술을 생략 복원 기술이라 한다.

현재 발화문에서 생략된 개체를 복원하려면 챗봇과 사용자 대화의 이전 발화문안에서 복원 대상을 추출하여 복원해야 한다. 현재 발화문에서 생략된 개체가 이전 발화문에 있다고 한다면 생략 복원 모델은 이전 발

화문에 출현한 개체들 중 현재 발화문에서 생략된 개체를 인식해서 복원해야 한다. 하지만 모든 이전 발화문에 항상 복원할 대상이 있는 것이 아니기 때문에 복원 대상이 없는 발화문들은 생략 복원 모델의 성능을 떨어뜨리는 원인 중 하나이다. 또한 사용할 딥러닝 기반 언어 모델들은 모델이 한번에 입력 받을 수 있는 이전 발화문들의 길이가 한정 되어있는데, BERT 기반 생략 복원 모델의 입력으로 복원에 도움이 안되는 발화문들까지 고려한다면 생략 복원 모델의 효율성이나 정확도가 떨어질 수 있다. 이런 점들을 보완하고자 본 논문에서는 현재 발화문의 생략 정보를 가지고 있는 이전 발화문을 인식하는 유효 발화문 인식 기술을 연구하였다. 본 논문에서는 “유효 발화문”을 현재 발화문에서 생략된 개체를 가지고 있는 이전 발화문이라고 정의한다. 예를 들어 “오늘 경기 일정 알려줘.” 라는 발화문은 ‘NC 다이노스’ 라는 개체가 생략되어 있다고 가정하자. 이 발화문에 대한 “유효 발화문”은 ‘NC 다이노스’ 라는 개체를 포함하고 있는 이전 발화문들을 의미한다.

최근 NLP 연구에 가장 많이 사용되고 있는 BERT[1]의 한국어 사전 학습 모델을 기반으로 유효 발화문을 분류하는 모델을 개발하였다. 본 논문은 유효 발화문 인식 연구와 생략 복원 기술에 유효 발화문 인식 모델을 적용한 방법에 대해서 설명한다. 논문 구성은 2장에서 관련된 연구를 설명하고, 3장은 유효 발화문 인식 모델의 세부적인 내용을 상세히 설명하였다. 4장에서는 학습과 평가에 사용한 데이터와 실험을 설명하고, 5장에서 유효 발화문 인식 모델의 성능과 모델을 실제 생략 복원 모델에 적용한 전, 후 결과를 분석하였다.

### 2. 관련 연구

본 논문에서 해결하려는 유효 발화문 인식 문제는 현재 발화문과 이전 발화문의 생략 복원 관계를 인식하는 분류 문제라고 할 수 있다. 자연어에 대한 분류를 수행하려면 우선적으로 해당 자연어 시퀀스의 토큰 별 혹은 시퀀스 전체의 표현 정보(representation)를 얻어야 하며, 현재 딥러닝을 사용한 언어 모델들이 좋은 성능을 보이고 있다. 이중 구글이 개발한 BERT와 XLNet[2], Allen 연구소가 개발한 ELMO[3] 등이 좋은 성능을 보이고 다양한 자연어처리 분야에서 활용되고 있다. 본 논문에서는 그 중 BERT를 한국어 말뭉치에 대해서 사전 학습한 한국어 BERT를 이용하여 발화문들의 임베딩 정보를 얻었다. 이처럼 구글이 공개한 BERT를 사용하고자 하는 언어나 도메인에 맞춰 사전 학습을 진행하는 연구도 있다. 의료, 생명 분야 논문 코퍼스로 사전 학습을 거쳐 의료, 생명 텍스트를 활용한 태스크들에서 전반적으로 좋은 성능을 거둔 BioBERT[4]와 과학, 기술 도메인에 사전 학습 시켜 과학, 기술 텍스트 전문 언어 모델 SciBERT[5]가 대표적이다.

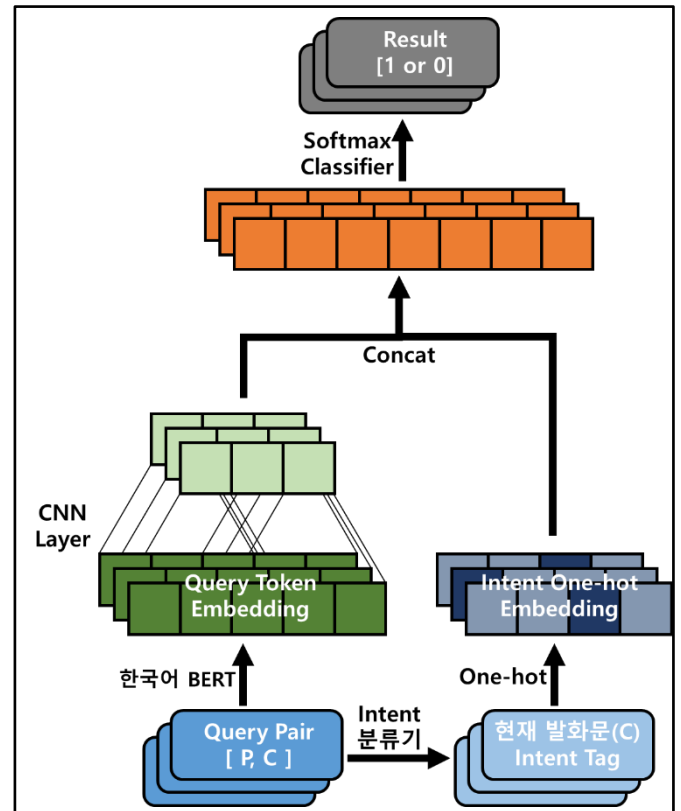
분류 문제 해결을 위한 딥러닝 연구들 중에는 Convolutional Neural Network(CNN)을 기반으로 텍스트를 분류한 Yoon Kim의 연구[6]가 대표적이며, 이를 한국어에 적용하여 텍스트 분류를 진행한 연구[7,8]들도 발표되었다. 이밖에 신경망 모델을 개선해서 분류 문제를 해결하려는 연구[9]도 있다. BERT 공개 이후 BERT를 활용한 텍스트 분류에 관한 연구들은 다수 이루어졌으며 계속해서 새로운 연구들이 발표되고 있다. BERT를 서로 다른 텍스트 분류 데이터셋에 적용하여 태스크 맞춤형 layer를 튜닝한 감정 분석 연구[10], 문서 전체를 분류하는 연구[11] 등이 발표되었다.

한국어 생략어복원의 가이드라인을 제시한 연구[12]나 문서 안에서의 전역적인 관계 추출을 위해 생략된 개체를 고려하는 연구[13] 등 한국어 생략어에 대한 연구가 있으나, 본 논문에서 다루는 형태와 같은 한국어 대화시스템에서의 생략 복원 연구는 국내외에서 많이 연구되거나 발표되지는 않고 있다. 한국어가 아닌 영어 문장에서의 생략 복원 연구들은 발표되고 있지만, 영문장에서의 생략 현상은 동사(구)가 생략되거나, 일반 명사를 대명사로 치환한 경우가 많아[15] 한국어에 적용하기에는 무리가 있거나 생략 복원보다는 상호 참조를 강조한 면이 있다. 또한 도메인에 따라 문법을 정의해야 하여[16] 구어체를 대상으로 하는 본 논문의 연구와 차이가 있다. 본 논문에서는 이전에 선행되었던 연구에 기반을 두어 개선점을 찾기 보다 비슷한 분류 문제를 해결하는 기존 연구들을 참고하였다.

### 3. 한국어 BERT를 이용한 유효 발화문 인식 모델

본 논문이 제안하는 유효 발화문 인식 모델의 전체적인 형태는 [그림 1]과 같다. 본 논문에서 제안하는 유효 발화문 인식 모델은 기본적으로 BERT를 기반으로 하고 있으며, BERT 모델의 입력 정보로는 현재 발화문(C)과 이전 발화문(P) 정보를 사용한다. 해당 토큰 임베딩을 CNN layer를 거쳐 얻은 벡터와 C의 질문 유형(intent) 정보 임베딩을 합쳐 분류기(classifier)에 입력으로 사용한다. 이때 분류기는 단순한 소프트맥스(softmax) layer로 구성된다. 분류기 결과를 통해, 해당 P가 C에 대하여 유효한지(C의 생략 정보를 P가 포함하는지) 예측 결과(1,0)을 얻을 수 있다. 인식 모델의 결과는 최종적으로 생략 복원 모델의 입력에 포함되어

사용하게 된다.



[그림 1] 유효 발화문 인식 모델 구조

#### 3.1 Input layer

학습 및 평가에 사용되는 한 사용자와 연속적으로 주고받은 대화 전체(Query set)에서 현재 시점의 사용자 질문 쿼리(C)와 각각의 이전 발화문(P)와 하나씩 쌍(P, C), Query pair)을 만들어 사용한다. 또한 각 C에 대한 질문 유형 정보를 One-hot 임베딩하여 사용한다. P와 C는 BERT에서 서로 다른 문장으로 구분되는 하나의 입력 시퀀스(input sequence)로 사용한다.

#### 3.2 한국어 BERT(Bidirectional Encoder Representations from Transformers) layer

본 논문의 모델은 구글에서 코드와 사전 학습 모델을 함께 공개한 언어 모델 BERT를 한국어 코퍼스에 새롭게 학습하여 (주)엔씨소프트에서 사용중인 한국어 BERT를 사용하였다. 한국어 BERT를 통해 쿼리(P, C)의 토큰 임베딩 및 쿼리 전체의 시퀀스 임베딩을 얻었다. Transformer[14] 구조를 활용하여 학습한 언어 모델인 BERT는 각 토큰 별 3가지(포지션, 세그먼트, 토큰) 임베딩 값의 합으로 이루어져 있다.

- 포지션 임베딩: 해당 토큰의 위치에 관한 임베딩
- 세그먼트 임베딩: 하나의 입력 시퀀스에서 여러 개의 문장이 들어왔을 때, 각 문장 별 임베딩
- 토큰 임베딩: BERT의 tokenizer인 wordpiece tokenizer를 한국어에 확장시킨 tokenizer의 임베딩

기존 구글의 BERT와의 차이점은 다음 [표 1]과 같으며, 특히 tokenizer와 학습 코퍼스에서 차이를 보인다.

[표 1] 구글 BERT와 (주)엔씨소프트 한국어 BERT 비교

	BERT[1]	한국어 BERT
종류	Base, Large	Base
Tokenizer	Wordpiece	형태소, 음절, 한국어 Wordpiece
학습 코퍼스	BookCorpus (800M words) English Wikipedia (2,500M words)	한국어 Wikipedia (1.9M sentences) 야구 뉴스 (23M sentences) 인터넷 뉴스 (108M sentences)
언어	English, Multilingual	한국어

한국어 BERT는 형태소, 음절, 한국어 Wordpiece 등 세가지 tokenizer를 지원한다. 그 중 제안 모델은 한국어 Wordpiece tokenizer를 이용한 한국어 BERT를 사용하였다. 한국어 Wordpiece tokenizer도 기존 BERT의 tokenizer처럼 세 가지 형태의 쿼리 임베딩을 얻을 수 있다. 입력 레이어의 입력 정보로는 tokenizer로 쪼개진 입력 시퀀스의 토큰들을 사용하고 토큰 임베딩, 포지션 임베딩, 세그먼트 임베딩 정보를 합한 한국어 BERT 모델의 표현 정보를 얻는다. 또한 Wordpiece tokenizer를 이용하면 '[CLS]' 토큰과 '[SEP]' 토큰과 같은 특수한 형태의 토큰을 갖게 되는데, [CLS] 토큰의 경우 입력 시퀀스에서 가장 처음 들어가 있는 토큰이며 BERT의 학습과정을 거치면 입력 시퀀스 전체의 표현 정보를 나타낸다. 제안 모델에서도 한국어 Only BERT 모델은 BERT만을 사용해서 입력 시퀀스의 임베딩을 얻기 위해 [CLS] 토큰의 출력 벡터를 사용한다. [SEP]토큰은 현재 발화문과 이전 발화문들 사이를 구분하기 위한 토큰으로 사용하였으며 제안 모델에서는 P와 C를 같은 BERT모델에 입력할 때, 두 발화문 사이에 [SEP]토큰을 넣어 현재와 이전 발화문이 서로 다른 세그먼트 임베딩을 갖도록 구성하였다.

### 3.2 CNN (Convolutional Neural Network) layer

CNN은 이미지의 지역적인 정보를 보존하여 학습하는 방법을 위해 만들어진 신경망 모델이지만, Y. Kim(2014)에서 제안한 모델처럼 텍스트의 순서, 문맥 정보 등의 지역적인 정보를 보존하여 학습하는 것이 가능하다. 텍스트의 각 토큰들의 임베딩을 이미지의 한 픽셀처럼 생각하여 window size(n)에 따라 n-그램 토큰 표현 정보를 추출하여 지역적 정보를 보존해서 Convolutional layer의 입력으로 사용한다. Convolutional layer의 최종 결과는 해당 텍스트의 표현 정보로 사용한다.

제안 모델은 P+C 입력 시퀀스에 대한 BERT 모델의 출력 시퀀스 값을 CNN Layer의 입력으로 사용하고, CNN layer를 통해 각 토큰의 지역적인 정보를 보존한 벡터를 얻는다. 이렇게 얻어진 벡터에 추가적인 자질들을 추가하여 최종 분류 레이어의 입력으로 사용한다.

### 3.3 Intent 임베딩 layer

제안 모델은 입력 발화문에 대한 파인 튜닝된 BERT 모델 결과와 CNN 결과뿐 아니라 추가적인 자질도 분류 layer의 입력으로 사용하였다. 자질들 중에는 입력으로 쓰인 Query pair가 속한 Query set의 문맥 정보, 현재

발화문(C)에서 실제로 복원해야 할 복원값, 혹은 현재 발화문의 질문 유형 정보 등이 있다. 추가 자질 중 현재 발화문의 질문 유형(Intent) 자질만 성능 향상에 도움이 되어 최종 제안 모델에서는 추가적인 자질로 질문 유형만 사용하였다.

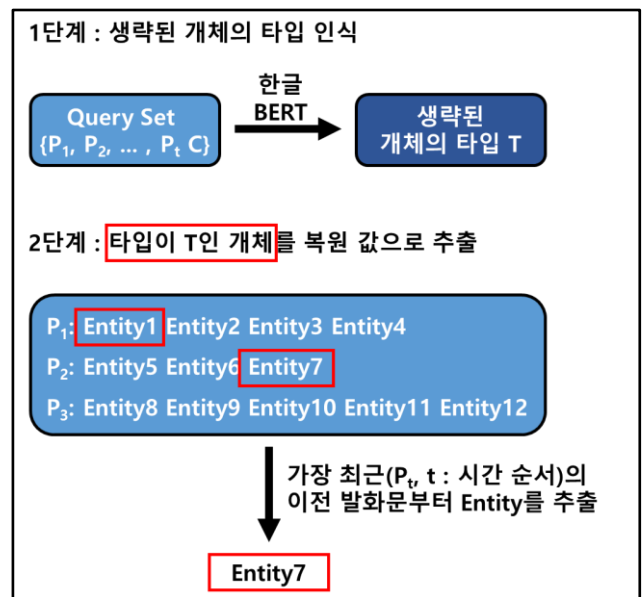
현재 발화문의 질문 유형은 53개의 Intent로 구성되며, 각 Intent는 야구 분야 대화의 질문 의도를 나타내는 경기일정, 구장, 매치업, 기관\_팀, 인물\_선수 등이 있다. 제안 모델에서 사용한 (주)엔씨소프트의 BERT 기반 Intent 분류기는 F1 95.0% 정도의 성능을 내는 모델을 사용했다. 그 외 Intent 분류기의 세부적인 내용은 본 논문에서 중점적으로 다룰 내용이 아니므로 생략한다. 현재 발화의 Intent 정보는 One-hot 임베딩 방식으로 표현된다. 이렇게 구한 현재 발화문의 Intent 임베딩을 앞서 3.2의 CNN layer의 최종 벡터와 연결하여 분류기의 입력 벡터로 사용하였다.

### 3.4 생략 복원 적용

본 논문에서 제안하는 유효 발화문 인식기는 야구 도메인 대화 시스템의 생략 복원 기술의 정확도를 개선하기 위해서 개발되었으므로 제안 모델의 효용성을 보기 위해서 생략 복원 모델에 적용하였다. 생략 복원 모델은 두가지 전제조건을 바탕으로 개발 되었다.

- 야구 도메인에서 현재 발화문에서 정답을 제약하는 생략된 주된 타입은 [시간, 날짜, 장소, 팀, 사람]이 대부분이기 때문에 이 5가지로 한정한다.
- 대화의 특성상 현재 발화문과 가장 가까운 최근의 이전 발화문의 복원 후보가 복원 정답이 될 가능성이 높다.

제안 모델을 적용할 생략 복원 모델은 위 두가지 전제조건을 나누어 [그림 2]와 같이 2단계로 구성된다.



[그림 2] 생략 복원 모델 구조

생략 복원 1단계에서는 현재 발화에서 생략된 복원 유형을 인식한다. 복원 유형은 야구 대화에서 자주 생략되는 중요 정보인 [시간, 날짜, 장소, 팀, 사람]의 5가지 유형이다. 복원 유형 인식기는 한국어 BERT 기반 분류 모델이며, 현재 발화와 이전 발화문을 입력으로 사용한다. 만약 복원할 타입이 없다면 'not'으로 인

식한다. 만약 복원 타입이 2개 이상일 경우에는 복원해야 할 모든 타입을 제시한다. 2단계에서는 1단계에서 인식한 복원 유형에 대한 복원값을 이전 발화문으로부터 추출한다. 대화에서의 생략 복원값은 대부분 가장 가까운 발화문에 있는 값인 경우가 많다. 그러므로 본 논문에서 사용한 생략 복원기에서도 단순히 가장 가까운 최근 이전 발화문부터 복원 대상 복원값을 찾는다. 복원 타입이 복수인 경우에는 복수의 각 복원 타입별 복원값을 찾는다.

본 논문에서는 제안 모델을 생략 복원기의 2단계 과정에 적용하였다. 2단계에서는 가장 최근의 이전 발화문부터 가장 오래된 이전 발화문까지 역방향으로 1단계에서 찾은 생략된 개체의 타입과 동일한 타입의 개체를 찾아 복원값으로 추출한다. 제안 모델을 2단계에 적용하는 방법은 이전 발화문들에 대해서 제안 모델이 생략 복원 대상으로 유효한지를 판단하고, 유효한 발화문들 중 가장 가까운 발화문에서부터 생략 복원값을 추출함으로써 생략 복원 모델의 정확률을 높이는 것이 목표이다.

## 4. 데이터 및 실험 방법

### 4.1 데이터

생략 복원 및 유효 발화문 인식 모델을 실험하기 위해서 야구 챗봇과 사람이 대화하는 가상 대화 시나리오 데이터를 수작업으로 구축하였다. 해당 시나리오 데이터는 다수의 이전 발화문과 하나의 현재 발화문으로 이루어져 있다(Query Set). 유효 발화문 인식은 하나의 현재 발화문에 대해서 여러 이전 발화문들 중 어느 발화문이 유효한지 판단해야 하므로 현재 발화문과 이전 발화의 (P, C) 쌍(Query Pair)으로 학습 데이터 및 평가 데이터를 구성하였다. 각 쌍에는 C에서 복원할 복원값 정답과 C의 질문 유형(Intent) 정보가 부착 되어있다.

[표 2] 학습 및 평가 데이터 상세

	'1' label	'0' label
학습	9810 pair	9810 pair
평가	1000 pair	1000 pair

데이터 상세는 [표 2]와 같으며, 특정 label에 모델이 편중되어 학습하는 것을 방지하고자, 데이터의 '0' label과 '1' label의 Query pair 비율은 1대 1로 맞추어 구성하였다. 또한 공식적인 평가 데이터가 있는 데이터나 문제가 아니고, 샘플링한 데이터 마다 성능의 차이가 큰 점을 감안하여 평가 데이터는 전체 데이터에서 같은 크기로 랜덤 샘플링(비복원 추출) 한 5개의 평가 데이터를 만들어 평가하였다.

### 4.2 실험 모델

실험을 수행한 모델은 총 3가지로 상세는 [표 3]과 같다. Baseline으로 사용한 Only BERT 모델의 경우, 별도의 쿼리 표현 정보를 사용하지 않고 한국어 BERT의 sentence 표현 정보 결과를 그대로 사용하였으며, 이전 발화문과 현재 발화문을 제외한 다른 자질들은 전혀 사용하지 않았다. BERT-CNN 모델들은 한국어 BERT에서 각

토큰 별 표현 정보를 가져와 토큰 임베딩으로 사용하였으며, 해당 토큰 임베딩을 CNN layer에 통과시켜 쿼리 표현 정보를 얻었다. BERT-CNN+ 모델은 BERT-CNN 모델에 Intent 정보를 추가 자질로 사용한 모델이다.

[표 3] 실험 모델 상세

Model	Only BERT	BERT-CNN	BERT-CNN+
BERT emb	Sentence	Token	Token
Query rep	BERT result	CNN result	CNN result
Intent emb	X	X	One-hot emb
Note	Baseline		

### 4.3 실험 환경 및 모델 hyper parameter

실험 환경은 다음과 같다.

- CPU: Intel Xeon E5-2630 v3 @ 2.4GHz
- Memory: 128GB
- GPU: Titan Xp(12GB) \* 2
- OS: Ubuntu 18.04

각 모델 별 hyper parameter는 [표 4]와 같으며, 기록하지 않은 기타 parameter들은 BERT base와 동일하다.

[표 4] 모델 hyper parameter

Model	Only BERT	BERT-CNN	BERT-CNN+
CNN layer	-	3	3
CNN window	-	5	5
Seq length	128	128	1280
Learning rate	5e-5	5e-5	5e-5
Batch Size	16	16	16
Epoch	10	10	10

## 5. 평가 및 분석

### 5.1 유효 발화문 인식 성능

성능 평가는 precision, recall, F1, accuracy를 사용하였고, 각각은 다음과 같다.

- TP = True Positive, FP = False Positive, TN = True Negative, FN = False Negative
- Precision =  $n(TP) / (n(TP)+n(FP))$
- Recall =  $n(TP) / (n(TP)+n(FN))$
- F1 =  $2 * Precision * Recall / (Precision + Recall)$
- Accuracy =  $n(TP) + n(TN) / (n(TP) + n(FP) + n(TN) + n(FN))$

유효 발화문 인식 모델의 모델 별 성능은 [표 5]와 같으며, BERT-CNN 모델에 Intent 임베딩을 추가한 모델이 가장 좋은 성능을 보인 것을 확인할 수 있다. F1 성능 기준으로 Only BERT 모델 대비 4.4%의 성능 향상을 보였으며, 토큰 임베딩에 CNN을 사용한 모델이 단순 한국어 BERT의 문장 표현 정보([CLS] 토큰의 임베딩 벡터)를 사용하는 것보다 우수한 것을 확인했다.



[표 5] 유효 발화문 인식 성능

Model	Prec	Recall	F1	Acc
Only BERT	0.8805	0.8859	0.8818	0.8828
BERT-CNN	0.9202	0.9159	0.9177	0.9195
BERT-CNN+	<b>0.9210</b>	<b>0.9165</b>	<b>0.9184</b>	<b>0.9202</b>

## 5.2 유효 발화문 모델의 생략 복원 적용 시 성능 변화

생략 복원의 2단계에서 유효 발화문 인식 모델 결과를 적용한 모델 별 상세 성능은 [표 6, 7]과 같다. 상단에는 유효 발화문 인식 모델의 효과를 검증하기 위해 모든 이전 발화문을 사용한 유효 발화문 인식 모델을 사용하지 않은 생략 복원 모델의 성능을 함께 제시하였다. 복원값을 정확히 맞췄는지 여부에 따른 Precision, Recall, F1 성능이며, 쿼리 기준[표 6]과 개체 기준[표 7] 성능을 나누어 제시하였다. 쿼리 기준의 경우 하나의 현재 발화문에 1개 이상의 복원값이 정답일 경우 이를 모두 맞춰야 정답으로 인정한 기준이며, 개체 기준의 경우 복원값 각각을 맞췄는지 틀렸는지를 평가한 기준이다.

생략 복원 2단계에서 제안 모델을 사용했을 때, 복원할 최근접 발화문들을 한정했기 때문에 잘못된 복원을 할 확률이 줄어들었다. 이를 통해 유효 발화문 인식 모델 효과로 상승하길 기대하였던 Precision이 약 5%정도 상승한 것으로 확인하였다. 유효 발화문 인식 모델을 적용함으로써 생략 복원에 실패하는 경우도 있지만, 개선된 효과가 더 커서 가장 좋은 성능을 보인 BERT-CNN 모델의 경우 Recall(재현율)도 소폭 상승하여 F1 score도 상승한 것을 볼 수 있다. 5.1의 유효 발화문 인식과는 다르게 생략 복원 모델에 적용했을 경우, intent 임베딩을 추가한 경우 적용 성능이 추가하지 않은 경우에 비해 떨어진 점을 확인할 수 있었다.

[표 6] 생략 복원 2단계 적용 시 성능 (쿼리 기준)

Model	Prec	Recall	F1
X	0.7724	0.9223	0.8407
Only BERT	0.7863	0.8932	0.8363
BERT-CNN	<b>0.8083</b>	<b>0.9238</b>	<b>0.8622</b>
BERT-CNN+	0.7833	0.9216	0.8468

[표 7] 생략 복원 2단계 적용 시 성능 (개체 기준)

Model	Prec	Recall	F1
X	0.8592	0.7922	0.8243
Only BERT	0.8837	0.7403	0.8057
BERT-CNN	<b>0.9044</b>	<b>0.7987</b>	<b>0.8483</b>
BERT-CNN+	0.8815	0.7727	0.8235

## 6. 결론

본 논문에서는 야구 도메인 대화 시스템의 성능 향상을 위한 생략 복원의 정확률을 올리기 위해서 현재 발화문과 이전 발화문의 생략 복원 관계를 분류하는 유효 발화문 인식 기술을 개발하였다. 개별 및 적용 실험을

통해 제안 모델이 생략 복원 모델의 정확률 향상에 도움이 되는 것을 확인할 수 있었다. 유효 발화문 인식은 꼭 필요한 발화문을 인식하는 것도 중요하지만, 방해가 될 수 있는 발화문을 사전에 차단하여 오류를 방지하는 역할이 더 중요하기 때문에 재현율에 비해 정확률이 올라간 점은 긍정적으로 보인다. 하지만 실제 사용자의 데이터가 아닌 사용자와 챗봇의 대화를 가정하고 제작한 데이터로 학습 및 평가가 이루어졌다는 한계가 있다. 차후 연구로는 서비스 되고 있는 챗봇의 실제 사용자 대화 데이터를 대상으로 학습하고 챗봇에 적용해 봄으로써 본 논문에서 제안한 모델이 서비스되고 있는 실제 환경에서도 도움이 되는지 실험해 보는 방법을 검토 중이다.

## 참고문헌

- [1] DEVLIN, Jacob, et al. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805, 2018.
- [2] YANG, Zhilin, et al. XLNet: Generalized Autoregressive Pretraining for Language Understanding. arXiv preprint arXiv:1906.08237, 2019.
- [3] PETERS, Matthew E., et al. Deep contextualized word representations. arXiv preprint arXiv:1802.05365, 2018.
- [4] LEE, Jinhyuk, et al. Biobert: pre-trained biomedical language representation model for biomedical text mining. arXiv preprint arXiv:1901.08746, 2019.
- [5] BELTAGY, Iz; COHAN, Arman; LO, Kyle. Scibert: Pretrained contextualized embeddings for scientific text. arXiv preprint arXiv:1903.10676, 2019.
- [6] KIM, Yoon. Convolutional neural networks for sentence classification. arXiv preprint arXiv:1408.5882, 2014.
- [7] KIM, Min, et al. Multi-channel CNN을 이용한 한국어 감성분석. 제 30회 한국어 및 한국어 정보처리 학술대회 논문집. pp.79-83. 2018.
- [8] CHOI, Kyungho, et al. CNN을 이용한 발화 주제 다중 분류. 제 29회 한국어 및 한국어 정보처리 학술대회 논문집. pp.56-59. 2017.
- [9] KIM, Wonwoo. and PARK, Kwang-Hyun. 합성곱 신경망을 이용한 한국어 텍스트 감성 분류기 설계. 한국컴퓨터종합학술대회 논문집. pp.642-644. 2017.
- [10] ZHENG, Jianzhong, et al. Short Text Sentiment Analysis of Micro-blog Based on BERT. In: Advanced Multimedia and Ubiquitous Engineering. Springer, Singapore, 2019. p. 390-396.
- [11] ADHIKARI, Ashutosh, et al. DocBERT: BERT for Document Classification. arXiv preprint arXiv:1904.08398, 2019.
- [12] RYU, Jihee, et al. 한국어 생략어복원 가이드라인. 제 29회 한국어 및 한국어 정보처리 학술대회 논문집. pp.213-219. 2017.
- [13] KIM, Kuekyeng, et al. 문서 내 전역 관계 추출: 생략된 개체의 고려. 제 30회 한국어 및 한국어 정보처리 학술대회 논문집. pp.47-49. 2018.
- [14] VASWANI, Ashish, et al. Attention is all you need. In: Advances in neural information processing systems. 2017. pp. 5998-6008.

- [15] SAHA, Amrita, et al. Complex sequential question answering: Towards learning to converse over linked question answer pairs with a knowledge graph. In: Thirty-Second AAAI Conference on Artificial Intelligence. 2018.
- [16] GUO, Daya, et al. Dialog-to-action: conversational question answering over a large-scale knowledge base. In: Advances in Neural Information Processing Systems. 2018. p. 2942-2951.