

건국봇: 검색모델과 생성모델을 결합한 챗봇

이현우*, 민덕기**1)

*건국대학교 컴퓨터공학과

e-mail:hyunwoo9301@naver.com, dkmin@konkuk.ac.kr

KU-Bot: Chatbot combining Retrieval-based model and Generative Model

Hyunwoo Lee* Dugki Min**

*Computer Engineering, Konkuk University

요 약

최근 AI 스피커를 비롯한 지능형 비서 서비스들이 빠르게 등장하고 있으며, AI 시장에서도 특히 챗봇 구축이 가장 활발하게 진행되고 있다. 건국봇은 건국대학교 학생들에게 필요한 정보를 제공하는 대화형 서비스이다. 본 논문에서는 대표적인 챗봇 구현 방법인 검색모델과 생성모델의 장단점을 분석하고, 건국봇에 적용한 사례를 소개한다. 궁극적으로, 질의문의 의도를 단어의 가중치를 고려해 추론함으로써 Unknown 추론을 강화하고 의도되지 않은 문장의 처리 관점에서 성능을 향상시키는 방법을 제안한다.

1. 서론

챗봇이란 정해진 응답 규칙 또는 인공지능 기반의 응답으로 사용자와 소통하는 서비스이다. 챗봇에 대한 연구는 크게 두 가지 접근방식으로 연구되어 왔다.[1] 첫째로, 검색모델 (Retrieval-based Model)[2,3]은 사용자의 발화를 미리 정의된 문장들과의 유사도에 따라 적절한 응답을 선택하는 방식이다. 정확한 답변을 제공할 수 있는 장점이 있으나, 정의되지 않은 범위의 응답은 할 수 없다. 둘째로, 생성모델 (Generative Model)[4,5]은 미리 정의된 응답에 의존하지 않고 새로운 응답을 생성하는 방식이다. 채팅이나 영화자막 텍스트를 학습데이터로 사용하여 새로운 응답을 유기적으로 만들어 내는 방식이다. 정의되지 않은 응답을 생성할 수 있는 장점이 있으나, 문법이나 맥락에서 벗어난 응답을 생성하기도 한다. 최근 연구 동향은 생성모델에 대한 연구가 활발하다. 하지만 생성모델은 실용화하기에는 여전히 부족함이 있어 기업의 솔루션 구축에는 검색모델을 주로 활용하고 있다.

글로벌 벤더들은 앞다투어 특화된 도메인에 실제로 적용할 수 있는 검색모델 위주의 솔루션을 제시하고 있다. IBM의 Wason[6], Microsoft의 LUIS[7]는 Intent, Entity 분석을 제공하는 대표적인 검색모델 솔루션이다. 한편, 한국에서도 네이버의 Clova Extension Kit과 같은 검색모델 기반 솔루션 개발이 활발히 진행되고 있다.

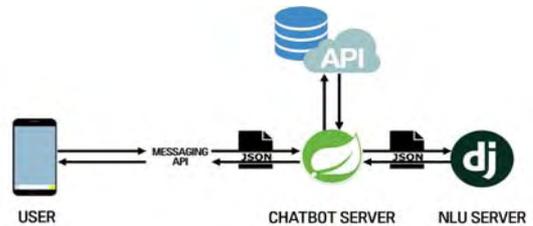
본 논문에서는 국내에서 가장 대중적으로 사용하는 메신저인 카카오톡을 기반으로 건국대학교 학생들에게 필요한 정보를 제공하는 대화형 서비스인 건국봇을 구축한 사례를 소개한다. 건국봇은 정확한 답변이 필요한 도메인에

대해서는 미리 정의된 문장들과의 유사도에 따라 DNN

기반으로 발화의도(Intent)를 추론하는 검색모델을 사용하고, 정의된 발화의도에 속하지 않는 Unknown 타입의 질의에 대해서는 생성모델인 Seq2Seq기반으로 응답을 생성하는 하이브리드 방식을 사용한다. 특히, TF-IDF²⁾와 n-제곱을 적용해 의도되지 않은 문장을 Unknown으로 따로 분류하는 방법과 이를 생성모델에 다시 결합하여 응답을 생성하는 방법을 제시한다.

이 논문의 구성은 다음과 같다. 먼저, 제 2절에서는 건국봇 구축에 적용한 시스템 구조에 역할에 대해 설명한다. 제 3절에서는 적용한 알고리즘 및 로직에 대해서 소개한다. 제 4절에서는 실험 결과와 성과에 대해서 소개하고, 제 5절에서는 논문의 결론을 내린다.

2. 건국봇 시스템 구조



[그림 1] 건국봇 시스템 구조도

[그림 1]은 본 연구팀이 구현한 건국봇의 시스템 구조도이다. 챗봇 서버와 자연어 이해 서버가 핵심 구성요소이다.

MESSAGING API는 사용할 대화형 인터페이스에서 제시하는 스펙을 따른다. 건국봇에서는 카카오톡 자동응답 API를 사용하였다. 사용자가 카카오톡을 통해 입력한 질

1) 교신저자

2) 어떤 단어가 특정 문서 내에서 얼마나 중요한 것인지를 나타내는 통계적 수치

의문은 카카오 서버를 통해 JSON 데이터 타입으로 변환되어 HTTP GET/POST 방식으로 챗봇 서버에 전달된다.

챗봇 서버는 자연어 이해 결과를 통해 최종응답을 생성하는 역할을 담당한다. JSON 형태로 질의문을 받은 챗봇 서버는 자연어 이해 서버로 질의문을 전달한다. 자연어 이해 서버의 검색모델을 통해 추론된 Intent 정보와 이전 대화를 참조해 최종 Intent를 결정하고, Entity 정보를 바탕으로 데이터베이스와 연동해 동적으로 응답을 생성한다. 만약 최종 Intent가 Unknown이라면 자연어 이해 서버의 생성모델에 요청해 응답을 제공받는다. 챗봇 서버는 반복 실패, 필수 Entity 요청 등의 예외처리 기능도 포함한다.

자연어 이해 서버의 구조는 크게 두 가지로 나누어진다. 첫째, 질의문을 분석해 Intent 및 Entity 정보를 제공하는 검색모델 구현부이다. 둘째, 생성모델을 사용해 응답을 생성하는 생성모델 구현부이다.

제 3절에서는 본 절에서 언급한 시스템의 구성요소에 대한 개발환경과 구체적인 구현 방법을 단계적으로 소개한다.

3. 구현 방법

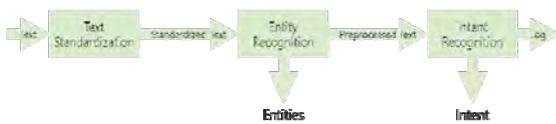
자연어 이해 서버와 챗봇 서버의 구현에 사용된 개발 환경 및 구현 방법에 대해 소개한다.

3.1 자연어 이해 서버

본 연구에서 자연어 이해 서버는 Django Framework를 사용하여 웹 서버를 구축하였다. 자연어 처리는 KoNLPy 패키지를 사용하였고, 학습 모형 개발은 Tensorflow를 사용해 구현하였다.

3.1.1 검색모델 구현부

자연어 이해 서버의 첫 번째 기능인 Entity 인식 및 Intent를 추론에 대한 연구와 개발 경험을 소개한다. 검색 모델 구현부는 아래 [그림 2]와 같이 구성되어 있다.



[그림 2] Main Components

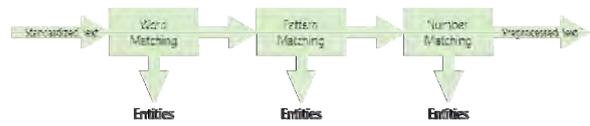
3.1.1.1 텍스트 표준화(Text Standardization)

표준단어와 다양하게 표현될 수 있는 동의어 목록을 정의하고 전처리 과정에서 변환한다.

- Before Text Standardization
- [근입]이랑 [어대]에 지하철 언제와? 12 010-1234-5678
- + After Text Standardization
- + [근대입구]이랑 [어린이대공원]에 지하철 언제와? 12 010-1234-5678

3.1.1.2 엔터티 인식(Entity Recognition)

문장에서 개체를 추출한다. 고유명사는 데이터가 다양해질수록 의미를 잃는다. 이를 하나의 개체로 연관시켜 중요도를 유지하고, 궁극적으로 Intent 추론에 의미 있는 정보를 제공하는 역할을 한다.



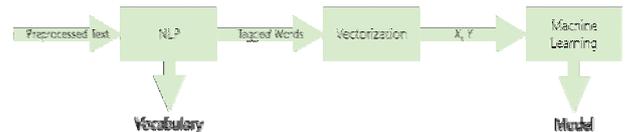
[그림 3] Entity Recognition

[그림 3]과 같이 엔터티의 인식은 세단계로 구성하였다. 첫째, 엔터티로 인식할 단어를 정의하고 이를 참조하여 인식하는 과정이다. 둘째, 날짜, 시간, 전화번호 등의 엔터티를 인식하기 위해 정규식을 이용해 패턴을 인식하는 과정이다. 셋째, 앞의 과정을 거친 후 남겨져 있는 숫자들을 인식하는 과정이다.

- Before Entity Matching
- [근대입구]이랑 [어린이대공원]에 지하철 언제와? [12] [010-1234-5678]
- + After Entity Matching
- + [장소]이랑 [장소]역에 지하철 언제와? [숫자] [전화번호]

3.1.1.3 발화의도 인식(Intent Recognition)

검색모델 구현부의 핵심 기능으로 본 연구에서는 미리 정의된 문장들과의 유사도를 분석하여 발화 의도를 추론한다.



[그림 4] Intent Machine Learning

발화 의도 학습은 위 [그림 4]와 같이 세단계로 진행되며, 학습 결과 생성된 Model을 통해 발화 의도를 예측한다.

1) 자연어 처리(Natural Language Processing)

KoNLPy를 활용해 학습된 문장을 자연어 처리하였다.

- Before NLP
- 장소이랑 장소역에 지하철 언제와? 숫자 전화번호
- + After NLP
- + 장소/Noun 장소/Noun 역/Noun 지하철/Noun 언제/Noun 오다/Verb 숫자/Noun 전화번호/Noun

2) 벡터화(Vectorization)

기본적으로 Classification은 Unknown이라는 결과를 내지 않는다. 다시 말해서, 학습되지 않은 데이터가 입력되어도 학습된 결과 중에서 하나를 선택하게 된다. 이를 방지하기 위해 모든 Unknown 케이스를 학습시키는 것은 현실적으로 불가능하다. 하지만, Unknown은 예외처리를 위해 매우 중요한 부분이다.

본 연구는 미학습된 데이터 뿐만 아니라 학습되었더라도 의미를 가지지 못하는 데이터의 조합까지 Unknown으로 분류하는데 기여한다. 모형의 입력데이터인 X는 문장의 단어들에 TF-IDF와 n제곱을 적용한 벡터다. TF-IDF를 통해 단어에 가중치를 부여하고 n제곱의 과정을 거쳐 중요한 단어의 값은 유지하고, 나머지 값은 0에 수렴시킨

것이다. 출력데이터인 Y는 Intent를 One Hot Encoding³⁾한 벡터로 이후 Decoding을 통해 Intent로 복원된다.

- Dictionary (Key: 단어, Value: 벡터의 인덱스 번호)
- {언제 : 0, 장소 : 1 ... 지하철 : 52}
- + TF-IDF Vectorization
- + [0.6, 0.4, 0, 0 ... 0, 0, 1.0]
- + TF-IDF + POW(n) Vectorization
- + [0.36, 0.16, 0, 0 ... 0, 0 1.0] (n=2)
- + [0.07776, 0.0256, 0, 0 ... 0, 0, 1.0] (n=4)

마지막으로 X:[0, 0 ... 0, 0]에 대해 Unknown을 출력하도록 학습데이터를 추가하는 것으로 벡터화를 마무리한다.

3) Deep Learning기반 Intent 추론 모델 생성

본 연구에서 Intent를 분류하는 모델 Deep Neural Network 모형으로 구현한다. 신경망의 첫 번째 히든 레이어의 차원의 개수는 [단어사전에 등재된 단어의 수, 다음 레이어의 차원의 개수]로 한다. 이후, 각 히든 레이어의 차원의 개수는 [이전 레이어의 차원의 개수, 다음 레이어의 차원의 개수]로 한다. 마지막 히든 레이어의 차원의 개수는 [이전 레이어의 차원의 개수, 분류 개수(=인텐트의 개수)]로 한다. 활성화함수로는 소프트맥스, 손실함수로는 크로스 엔트로피를 사용한다. 추가적으로, learning_rate, epoch, hidden layer 개수 등의 옵션을 조정하며 성능 튜닝을 진행한다.

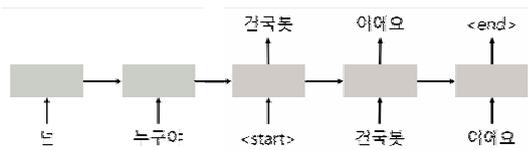
4) 발화의도(Intent) 예측

예측은 학습과 마찬가지로 동일한 전처리 과정을 거친 후 학습된 Model을 Classifier로 사용한다. 소프트맥스의 결과로 요소의 합이 1인 확률을 얻게 된다. 이 값을 가공해 Probability를 제공하고, 가장 확률이 높은 인덱스에 매칭되는 Intent를 찾음으로써 추론을 마무리한다.

- Classification
- [0.8, 0.1, 0.0, 0.0, 0.1]
- + Decode
- + 지하철조회

3.1.2 생성모델 구현부

본 연구에서는 생성모델을 구현하는 대표적인 기술인 Seq2Seq를 사용하였다. Seq2Seq은 2개의 RNN Cell인 Encoder와 Decoder로 구성된다. 아래 [그림 5]과 [그림 6]은 Cell의 순차적인 입력을 풀어서 표현한 것이다.

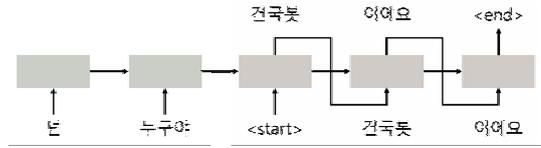


[그림 5] Seq2Seq Machine Learning

[그림 5]는 Seq2Seq의 학습방법을 나타낸다. 입력문장의 단어 개수만큼 Encoder에 입력을 반복한다. 마찬가지로

3) 고유한 N개의 범주형을 한 요소만 1이고 나머지는 0인 N차원 벡터로 표현하는 방법

로, 출력문장의 단어 개수만큼 Decoder에 입력을 반복한다. 각 입력은 단어벡터와 상태정보를 포함한다.

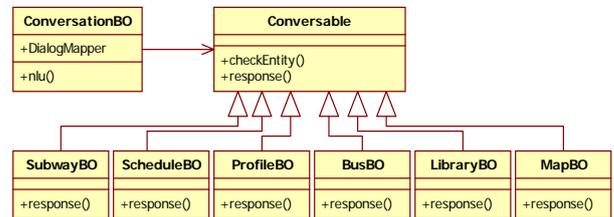


[그림 6] Seq2Seq Prediction

[그림 6]은 Seq2Seq의 예측방법을 나타낸다. [그림 5]에 서와 달리 Decoder의 출력이 다시 Decoder의 입력으로 사용된다. <end>가 출력될 때 까지 반복하며, 각 출력에서 <end>를 제외한 것이 응답 문장이 된다.

3.2 챗봇 서버

챗봇 서버는 Spring Framework을 사용하여 구현하였다. [그림 8]은 챗봇 서버의 다이얼로그 생성과 관련된 로직을 담은 클래스 다이어그램이다.



[그림 7] 챗봇 서버 다이얼로그 클래스 다이어그램

ConversationBO의 DialogMapper는 Intent별로 상세 로직을 담은 클래스를 매핑한다. nlu() 메소드는 자연어 이해 서버에 분석을 요청하는 메소드이다. 이전 대화를 참조해 Intent를 결정하고 DialogMapper를 통해 Intent에 매핑된 다이얼로그 클래스로 로직을 분기한다. ConversationBO는 Conversable를 상속하고 response() 메소드를 구현한 다이얼로그 클래스를 Intent 추론 결과에 따라 주입받음으로써 런타임 의존관계를 가지게 된다. Conversable의 checkEntity()는 필수 엔티티를 존재 여부를 검사하고, 없을 경우 자동으로 사용자에게 요청하는 메소드이다.

챗봇 서버는 각 사용자마다 intent, entity, subject, tryCount 등의 정보를 관리한다. subject는 다이얼로그 클래스 내에서 절차적인 대화를 구성하기 위해 사용한다. tryCount는 같은 subject에서 대화를 시도한 횟수를 나타내는 값으로, 반복실패에 대한 예외처리에 사용한다.

4. 구현 환경 및 실험 결과

본 절에서는 건국봇의 구현에 사용된 오픈 소스 소프트웨어를 포함한 구현 환경과 구현된 UI 모습 및 실험결과를 제공한다.

4.1 구현 환경 및 UI

본 연구에서 자연어 이해 서버는 Python 3.5 버전, Django Framework 1.11 버전, Tensorflow 1.3.0 버전을 사용하여 구현하였다. 챗봇 서버는 Spring Framework 4.3.11 버전을 사용하여 구현하였고, WAS는 Apache Tomcat 7.0 버전을 사용하였다.

건국대학교 학사 서비스와 관련된 기능은 건국대학교 앱에서 사용하는 API를 사용하였다. 대중교통에 관련된 기능은 공공데이터 포털에서 제공하는 API를 사용하였다. 버스나 지하철의 경우 조회하는 대상이 많으면 카카오톡의 5초 타임아웃 제약조건에 위배될 수 있다. 건국봇은 성능을 위해 Spring의 AsyncRestTemplate를 사용한 비동기 요청으로 문제를 해결하였다. 아래 [그림 10]은 최종적으로 건국봇이 서비스되고 있는 화면이다.



[그림 8] 카카오톡 플러스 친구의 건국봇

다음은 본 연구에서 사용한 학습데이터와 실험결과를 예시를 통해 소개한다.

4.2 학습데이터

초기 학습데이터는 예상되는 질문을 본 연구 팀이 직접 작성하는 것으로 시작하였다. 입력과 응답에 대해 log를 남기도록 하고 서비스를 통해 log에 쌓인 표현들을 바탕으로 다양한 표현을 학습시키고 동의어 목록을 만들었다.

| Intent | Sentence |
|--------|--------------------------------|
| 위치조회 | 새천년관 위치 가르쳐줘, 새천년관이 어디야, ... |
| 시간표조회 | 오늘 내 일정 가르쳐줘, 시간표 가르쳐줘, ... |
| 프로필조회 | 민덕기 교수님 전화번호 가르쳐줘, ... |
| 지하철조회 | 건입에 지하철 언제와, 지하철 시간표 가르쳐줘, ... |

[표1] 건국봇 학습데이터 샘플

[표1]은 건국봇 학습데이터의 일부이다. “가르쳐줘”를 포함해 “알려줘”, “말해줘”등은 여러 문서에서 나타났다. 다시 말해서, 언급된 단어들은 다양한 문서군에서 나타나는 가중치가 낮은 단어이다. 실험에서는 이처럼 가중치가 낮은 단어 또는 단어들의 조합을 대상으로 실험을 진행하였다.

4.3 문장 추론 실험 결과

실험은 건국봇에 개발에 사용한 학습데이터를 IBM Watson, Microsoft LUIS에 동일하게 학습시키고, 각 서비

스의 Web UI에서 제공하는 테스트 도구를 활용해 추론 결과를 바탕으로 한다. 실험은 의도되지 않은 문장을 Unknown으로 추론하는 관점에서 진행하였다.

| | Intent Prediction |
|----------------|-------------------|
| IBM Watson | 연락처 조회 |
| Microsoft LUIS | 위치 조회 |
| TF-IDF + POW | Unknown |

[표2] 상용플랫폼 / 건국봇 알고리즘 예시 문장 추론 결과

위의 [표 2]는 “가르쳐줘”를 입력해 Intent를 추론한 결과이다. 이처럼 가중치가 낮은 단어의 조합으로만 입력하는 경우 각자 다른 추론 결과를 보인다. 입력 자체가 의도되지 않은 문장이므로 Unknown으로 분류하는 것이 적합하다.

본 논문에서 제시하는 방법을 사용했을 때, Unknown으로 추론이 가능한 것을 확인할 수 있다.

5. 결론

본 프로젝트는 검색모델과 생성모델의 장점을 취한 하이브리드 모델 개발방법을 제시하고 있다. 생성모델은 분명 이상적이지만 현재 기술적으로 실용화하기에는 부족한 부분이 많았다. ‘정확한 답변’은 기업의 솔루션 구축에 필수적이므로 검색모델에 기반한 모델링을 먼저 진행하는 것이 바람직해 보인다. 하지만, 검색모델에서 사전에 모든 의도를 정의하는 것이 불가능하므로 부차적인 기능은 생성모델을 통해 응답하는 방식으로 개발했다.

다시 말해서, 우리는 비즈니스 업무를 분석해 사전에 의도를 정의하고 대화를 구성하였다. TF-IDF와 n제곱을 적용해 의도되지 않은 문장을 따로 분류할 수 있게 하였고, 이를 생성모델에 다시 결합하여 응답을 생성하였다. 결과적으로, 핵심 업무를 부차적인 업무 분리해 개발할 수 있게 함과 동시에 부차적인 업무는 특별한 모델링 없이 응답을 생성할 수 있게 되었다.

Acknowledgement

본 연구는 과학기술정보통신부 및 정보통신기술진흥센터의 대학ICT연구센터육성지원사업의 연구결과로 수행되었음 (IITP-2018-2016-0-00465)

참고문헌

- [1] <https://brunch.co.kr/@gentlepie/18>
- [2] Alexander Bartl, A retrieval-based dialogue system utilizing utterance and context embeddings, 2017
- [3] Ellis Pratt, Artificial Intelligence and Chatbots in Technical Communication
- [4] Ilya Sutskever, Sequence to Sequence Learning with Neural Networks
- [5] Ilya Sutskever, Generating Text with Recurrent Neural Networks
- [6] <https://console.bluenix.net/docs/services/conversation/index.html#about>
- [7] <https://docs.microsoft.com/en-us/azure/cognitive-services/LUIS>
- [8] 양민철, 예제 기반 챗봇을 위한 기계 학습 기반의 발화 간 유사도 측정 방법
- [9] Alan Nicho, Understanding intents and entities
- [10] Dan Jurafsky, Conversational Agents, 2012