# Node.js에서 MeCab 라이브러리와 Kakao API를 이용한 메신저 기반 개인화 채팅 봇 시스템

심대수*, 박두순*
*순천향대학교 컴퓨터소프트웨어공학과
e-mail : tlaeotn123@naver.com

# A Personalized Messenger Chat Bot System using MeCab Library and Kakao API in Node.js

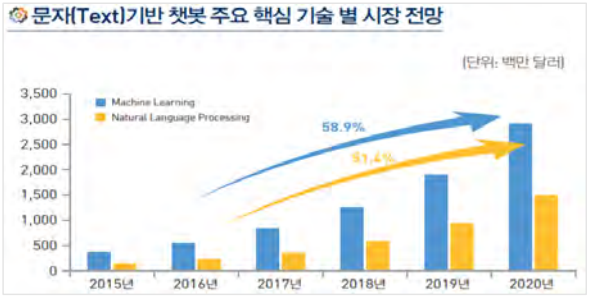
Dae-Soo Sim*, Doo-Soon Park*
*Dept. of Computer Software Engineering, SoonChunHyang University

요 약

하드웨어의 속도 발전과 데이터의 누적으로 생성된 수많은 빅 데이터의 활용을 통해 인공지능에 대한 무수히 많은 알고리즘과 시스템이 발전되고 있다. 그중 자연어 처리는 각 국가별 언어적 독립성으로 인해 국가별로 많은 연구가 이루어지고 있으며 앞으로 계속하여 발전해야 하는 분야이다. 이러한 현대 추세에 맞추어 본 논문에서는 사용자의 개인별 누적된 데이터를 통해서 개인에게 맞추어진 개인화 채팅 봇 시스템을 AWS EC2 Instance로 Node.js환경에서 MeCab 라이브러리와 Kakao API, Word Embedding 기법을 이용하여 보다 개인에게 맞춤화된 채팅 봇 시스템을 개발한다.

## 1. 서론

정보화 시대에 데이터 누적과 더불어 하드웨어의 성능 향상과 많은 연구자들의 노력으로 인해 인공지능 시장은 날이 갈수록 확장되어 가고 있다. 이세돌' 바둑기사와의 알파고, 아이폰의 Siri, Watson, 자율주행 자동차 등 시간이 지남에 따라 인공지능이 많은 영역을 대체하고 있는 추세이다. 이러한 추세는 외국에서만 멈추지 않고 국내에서도 발전하고 있으며 2018 ICT 산업전망컨퍼런스 에서는 2018년 ICT 10대 이슈 중 1위로 인공지능(AI)을 뽑았다[1]. 이러한 수많은 인공지능 분야에서 문자(Text)기반 채팅 봇은 고객 센터 무인화 등 많은 분야를 대체할 수 있으며, 각 기술별 채팅 봇시장 또한 커지고 있다. 국가과학기술정보센터(NDSL)에서 16년도 발표한 채팅 봇 핵심 기술 별 시장 전망은 (그림 1)과 같다.




(그림 1) 2015-2020 채팅 봇 핵심 기술 별 시장 전망[2]

※ 본 연구는 NRF-2017R1A2B1008421에 의해 지원되었음

이처럼 성장해 가는 채팅 봇 시장에서 사용자의 요구 사항에 따라 '심심이', '가짜톡' 등 이미 기존에 여러 시스템들이 만들어 졌으나, 각 개인에게 맞춤화된 채팅 봇을 개발하기란 쉬운 일이 아니다. 또한 기존의 채팅 봇 시스템들의 문제점인 수동적 응답과, 사용자에 따른 대화 개별성에 대해서 본 시스템에서는 사용자에 따른 개별 대화 학습 테이블을 두어 수동적 응답 및 대화 개별성 문제를 완화시켰다.

본 논문에서는 AWS(Amazon Web Service)의 리눅스 기반 EC2 Instance, Node.js를 이용해 서버를 구성하였으며, MeCab 형태소 분석기 라이브러리를 이용하여 문장의 형태소 분석을 진행하고 문장의 각 단어의 유사도를 측정하기 위한 Word Embedding 방법으로는 Co-Occurrence Matrix를 이용해 단어를 벡터로 변환하여 코사인 유사도(Cosine Similarity)로 각 벡터의 각도로 유사도를 측정하였다, 이렇게 측정된 단어들의 유사도를 통해 문장 유사도를 구했으며, 사용자 회화 DB는 MySQL로 구성하였고 Kakao API를 이용해 실제 메신저를 연동하고 각 개인마다 학습 테이블을 두어, 기존의 채팅 봇 시스템과 다르게 개인에게 맞춤된 채팅 봇 시스템을 구현하였다.

## 2. 메신저 기반 채팅 봇 시스템의 구성

메신저 기반 채팅 시스템의 동작 과정은 크게 형태소 분석, Word Embedding, 코사인 유사도(Cosine Similarity)를 통한 단어, 문장 유사도 계산, 응답 단계로 나눠볼 수

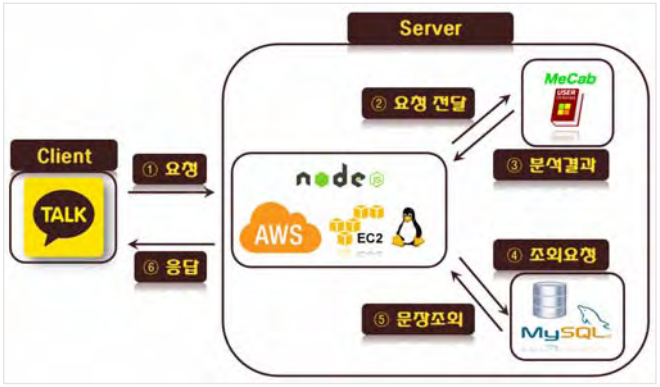
있으며, 이러한 과정을 자세히 살펴보면 다음과 같다.

(1) 형태소 분석(Morphology Analysis): 메신저 기반 채팅 시스템에서의 입력 데이터는 1개의 문장에 대해 n개의 단어로 표현될 수 있다. 각 문장의 단어는 형태소라는 단위로 분석되어 사용된다.

(2) 문장 유사도 계산(Sentence Similarity Calculation): 각 문장에 사용된 단어의 유사도를 계산하여 유사 문장을 탐색하는 과정이다. 두 단어의 유사도를 측정하는 방법으로써는 단어를 그 중에서 벡터 공간에 단어를 표현하는 단어 임베딩(Word Embedding)하여 하나의 벡터로 표현한 뒤 각 단어 벡터를 코사인 유사도(Cosine Similarity)를 이용한다[3].

(3) 응답(Response): 각 문장의 유사도를 측정한 뒤 가장 근접한 유사도를 가지는 문장에 대한 응답을 돌려준다.

따라서 본 논문에서 구현한 채팅 봇 시스템의 알고리즘은 (그림 2)와 같다.




(그림 2) 채팅 봇 시스템 알고리즘

(그림 2)의 채팅 봇 시스템에 대한 알고리즘은 다음과 같다.

**[알고리즘 1]** 채팅 봇 시스템에 대한 알고리즘

[방법] 1. 사용자는 Kakao 메신저를 통해 채팅 봇에게 회화 요청
2. 요청 회화 데이터를 형태소 분석을 위해 MeCab 형태소 분석기에 요청
3. MeCab 형태소 분석기는 완료된 형태소 분석 결과를 서버에게 전달
4. DB에 현재까지 누적된 Q&A 테이블 의 모든 질문을 Query로 요청
5. 서버는 각 모든 문장에 대해 유사도를 검사
6. 결정된 응답을 Kakao API 에 맞추어 전송

여기서, 각 문장의 유사도 판별은 문장에 속해있는 단어의 유사도를 통해 연산하며, 각 단어의 유사도는 Co-Occurrence Matrix를 이용해 Word Embedding을 한 벡터 결과를 코사인 유사도(Cosine Similarity)를 통해 연산하며 1에 가까울수록 유사한 단어라는 의미를 지닌다. Co-Occurrence Matrix는 영상분석 등에 사용되며, 본 논문에서는 단어의 벡터를 구하기 위해 사용되었으며 만들어지는 단어 벡터는 (표 1)과 같다.

(표 1) Co-Occurrence Matrix Word Embedding

| | He | is | not | lazy | intelligent | smart |
|---|---|---|---|---|---|---|
| He | 0 | 4 | 2 | 1 | 2 | 1 |
| is | 4 | 0 | 1 | 2 | 2 | 1 |
| not | 2 | 1 | 0 | 1 | 0 | 0 |
| lazy | 1 | 2 | 1 | 0 | 0 | 0 |
| intelligent | 2 | 2 | 0 | 0 | 0 | 0 |
| smart | 1 | 1 | 0 | 0 | 0 | 0 |

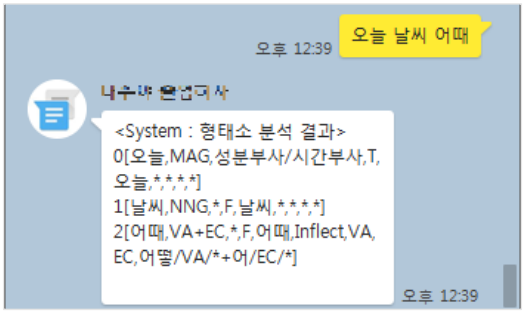
위와 같은 형태로 만들어진 두 단어(Vector)의 유사도를 구하는 방식은 (그림 3) 코사인 유사도(Cosine Similarity)를 사용하여 유사도를 측정한다.

$$sim(A,B) = \frac{\sum_{i \in I_{AB}} R_{A,i} \; R_{B,i}}{\sqrt{\sum_{i \in I_{AB}} R_{A,i}^{\,2}} \sqrt{\sum_{i \in I_{AB}} R_{B,i}^{\,2}}}$$

(그림 3) 코사인 유사도(Cosine Similarity)

여기서 R은 n x n의 Co-Occurrence Matrix이며, $I_{AB}$는 A 단어, B단어의 벡터이다, $R_{A,i}$와 $R_{B,i}$는 단어 벡터 A, B 와 공통으로 나타난 단어 i의 Count를 뜻한다[4]. 즉 A, B와 공통으로 나타난 단어의 Count 벡터를 이용해 A,B 단어 벡터 사이의 각을 구하여 두 단어 사이의 유사도를 측정한다.

또한 채팅 봇 시스템을 만들기 전 Kakao 메신저 기반의 생활 회화 데이터 3,000건의 데이터를 기반으로 학습데이터 및 Word Embedding에 사용하였으며, 학습에 쓰인 회화 데이터의 형태소 분석 결과 데이터는 (그림 4)와 같다.




(그림 4) 생활회화 학습에 사용된 학습데이터 일부

## 3. 메신저 기반 채팅 봇 시스템의 구현

본 논문에서 구현한 채팅 봇 추천 시스템은 기본적으로 Kakao Talk에서 제공하는 플러스친구 API를 사용하기 때문에 관리자로 등록하여야 하며 등록은 다음 (그림 5)와

같이 할 수 있다.


(그림 5) Kakao API를 사용하기 위한 관리자 등록[5]

관리자를 등록하게 되면 Kakao API를 확인할 수 있으며 기본적인 API를 상세하게 살펴보면 다음과 같이 두 가지로 볼 수 있다.

(1) Keyboard API : Keyboard API는 기본적으로 사용자가 채팅방에 입장했을 시 호출되는 API로서 다음 (그림 6)과 같은 API 규격을 가지고 있으며, 이를 이용해 사용자의 Input 범위를 버튼Input, Keyboard Input등으로 조절할 수 있다.


(그림 6) Kakao Keyboard API 규격[5]

(2) Message API : Message API는 사용자의 응답으로 실질적인 회화가 이루어지기 위해 사용되는 API로서 Request API 규격은 (그림 7)과 같으며 Response API 규격은 (그림 8)과 같다.

Specification

- **Method** : POST
- **URL** : http(s)://:your_server_url/message
- **Content - Type** : application/json; charset=utf-8
- **Parameters**

| 필드명 | 타입 | 필수여부 | 설명 |
|---|---|---|---|
| user_key | String | Required | 메시지를 발송한 유저 식별 키 |
| type | String | Required | text, photo |
| content | String | Required | 자동응답 명령어의 메시지 텍스트 혹은 미디어 파일 uri |

(그림 7) Kakao Message Request API 규격[5]

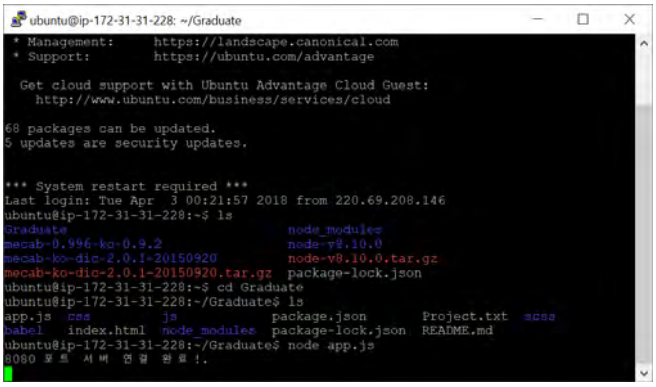
Response

| 필드명 | 타입 | 필수여부 | 설명 |
|---|---|---|---|
| message | Message | Required | 자동응답 명령어에 대한 응답 메시지의 내용.<br>6.2에서 상세 기술 |
| keyboard | Keyboard | Optional | 키보드 영역에 표현될 명령어 버튼에 대한 정보.<br>생략시 text 타입(주관식 답변 키보드)이 선택된다.<br>6.1에서 상세 기술 |

(그림 8) Kakao Message Response API 규격[5]

또한 기본적으로 API 통신을 하기위한 서버로는 AWS (Amazon Web Service)의 Linux 기반의 EC2 Instance를 사용하기 때문에 다음 (그림 9)와 같이 대여한 뒤 (그림 10)과 같이 Putty 프로그램을 이용하여 접속 및 관리 한다.


(그림 9) Amazone에서 대여한 EC2 Instance




(그림 10) Putty를 이용해 EC2 Instance에 접속한 모습

추가적으로 사용자의 응답을 형태소 분석할 때 사용되는 MeCab 라이브러리를 설치하기 위해 MeCab 홈페이지에서 다운받아 사용해야 하며 Node.js를 통한 서버환경을 만들기 위해 Node.js 홈페이지에서 추가적으로 다운받아야 한다.

그리고 학습데이터로 사용될 생활회화 데이터를 전처리하여 (그림 11)과 같이 형태소단위로 분석 DB에는 하나의 문장에 대해 한 단어씩 형태소로 변환된 데이터, 단어의 품사, 단어 누적 Count, 질문유형 등을 저장한다.


(그림 11) 누적된 학습 DB

그 뒤 단어 간 유사도 측정을 위해 사용될 벡터를 구현하기 위하여 Word Embedding을 진행하기 위하여Co - Occurrence Matrix를 사용해서 Matrix를 구성한다. 결과로 얻은 Matrix를 이용하여 단어 간 유사도를 측정, 문장의 유사도를 판별할 수 있으며 다음(표 3)은 '오늘 날씨 어때?'라는 문장 Input과 '날씨 어때?' 라는 문장의 문장 유사도를 측정한 결과와 (그림 12)는 실제 사용 결과이다.

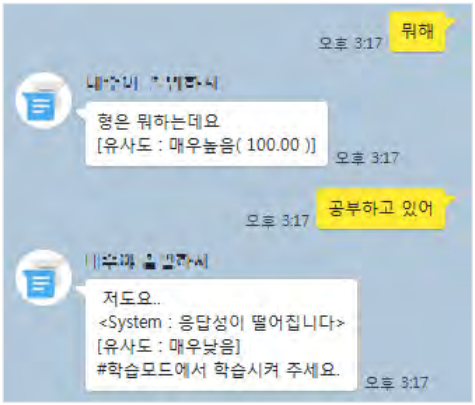
(표 3) 문장 유사도 비교 결과 샘플

| Sentence A | Sentence B | Similarity |
|---|---|---|
| 오늘 날씨 어때? | 날씨 어때? | 0.7 |
| 채팅 봇 시스템 유사도 분석 예 | 채팅 봇 유사도 분석 시스템 | 0.8571 |
| 공부 하는건 재밌어? | 공부 재밌어? | 0.7684 |
| 시험 준비는 좀 했어? | 기분 어때 | 0.167 |

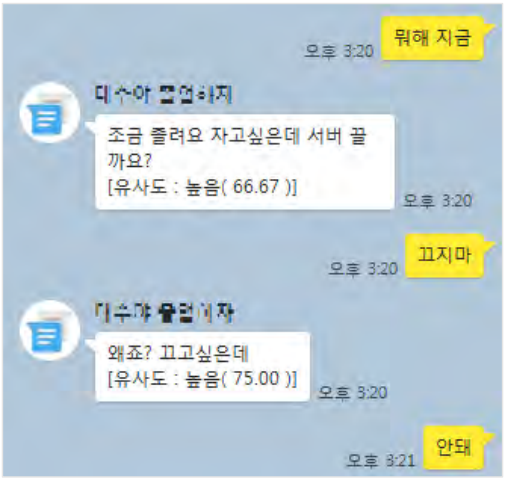

(그림 12) 날씨 요청에 대한 응답

위처럼 Front End로서 Kakao Talk을 사용하기 때문에 기본적으로 신규 메신저를 개발할 필요가 없으며, 사용자의 접근성이 높다는 것이 장점이며, 사용자 별 학습 테이블을 개별로 두어, 개별 학습이 진행되기 때문에 보다 사용자에게 맞춤화된 학습이 진행된다.

최종적으로 임의 질문에 대한 챗봇 시스템과의 회화 예시는 (그림13)과 (그림14)와 같다.




(그림 13) 사용자 요청에 대한 챗봇 시스템 응답




(그림 14) 사용자 요청에 대한 챗봇 시스템 응답

## 4. 결론

본 논문에서는 데이터의 누적으로 인한 많은 데이터를 이용하는 최신 인공지능 분야 중 채팅 봇 시스템의 문제점인 사용자 개별성 부분과, 수동적 질문에 대해 개선하기 위해 각 사용자 별로 학습 테이블을 구성하여 개별 학습이 진행되게 하였으며, Kakao Talk API를 이용해 사용자에게 보다 친숙하고 접근성 높은 채팅 봇 시스템을 개발 하였으며 Node.js를 통해 서버를 구성해 통신을 쉽게 하도록 하였다 형태소 분석은 MeCab 라이브러리를 통해 진행하였고, 또한 본 회화 시스템을 이용하여 추가적으로 고객센터 Q&A 센터 무인화와 같은 시스템에 사용될 수 있을 것 이며, 보다 높은 보안과, 의료 전문가들의 조언을 통해 상담형 채팅 봇 시스템으로의 확장도 가능할 것이다. 결과적으로 기존의 시스템보다 개인에게 집중되며, 접근성 높은 채팅 봇 시스템을 개발하였으나, 본 채팅 봇 시스템의 문제점인 학습데이터 부족으로 인한 희박성 문제(Sparsity Problem)에 따라 학습되지 못한 응답에 대해서는 잘 응답하지 못하는 낮은 응답성을 보인다. 따라서 이 부분을 보안하기 위해 더욱 연구가 진행되어야 할 것이다.

## 참고문헌

[1] 2018년 ICT 10대 이슈, 정보통신기술진흥센터, 2017.11
[2] 챗봇 핵심 기술별 시장전망(한국), KISTI MARKET REPORT, 2016.06
[3] 최영관, "영상처리:이미지 검색을 위한 색상 성분 분석", 한국정보처리학회 논문지, pp.403-410, 2004.08
[4] 이재식, "장르별 협업필터링을 이용한 영화추천 시스템의 성능 향상", 한국 지능정보 시스템 학회 논문지, pp. 66-78, 2007.12.
[5] Kakao API Specifications, 다음 카카오, 2017.05