

중간보고서

전기컴퓨터공학부 정보컴퓨터공학전공

신우창(201424479)

이태오(201424513)

김윤호(201424427)

과제명 : 딥러닝과 검색엔진을 활용한 질의응답시스템

지도교수 : 권 혁 철

목차

1. 과제의 목표	3
2. 요구 조건 및 제약사항 분석.....	3
2.1. 요구조건.....	3
2.2. 제약사항	3
2.3. 대책	3
3. 설계 상세화 및 변경내역	4
4. 갱신된 과제 및 추진계획.....	7
5. 데이터셋 수정사항.....	8
6. 구성원 별 진척도	8
7. 과제 수행 내용 및 중간 결과.....	9

1. 과제의 목표

딥러닝과 검색엔진을 활용한 질의응답 시스템

- 제한된 도메인에서 정보검색 기술을 바탕으로 사용자의 질문에 응답하는 한국어 질의응답 시스템을 개발한다.
- 검색 엔진으로 사용자의 질문의 주제에 해당하는 위키피디아 및 나무위키 문서를 검색하고, 딥러닝을 이용하여 위키피디아 및 나무위키 문서 내에서 해당하는 답변을 추출하는 것을 목표로 한다.

2. 요구 조건 및 제약사항 분석

2.1. 요구조건

1. User Interface(질의)

클라이언트 프로그램은 사용자가 질문을 하고 답변을 받는 과정에 있어서 여러가지 편의성을 제공하고, 사용자는 클라이언트 프로그램의 UI를 통해 제한 없이 질문을 입력 할 수 있어야 한다.

2. 검색엔진

검색엔진은 사용자의 질의에 따라 한국어 위키피디아의 모든 문서에서 관련 정보가 있는 텍스트를 고속으로 검색하여 출력하여야 한다.

3. 질의응답 모델 구현

질의 응답 모델은 자연어 처리 모델의 자연어 처리 결과를 입력으로 받아 사용자의 질의에 대해 가장 높은 응답을 출력해야 한다.

4. User Interface(응답)

사용자는 클라이언트의 UI를 통해 질의에 대한 응답을 확인 할 수 있어야 한다.

5. 성능개선

Open 도메인 질의응답 시스템의 정확도를 개선시킨다.

2.2. 제약사항

1. 한국어 질의 응답 시스템

과제의 목표에 따라 질의 응답 시스템의 언어처리는 한국어로 제한된다.

2. 검색엔진

연구실에서 제공하는 미리내 검색엔진을 개량하여 사용한다.

3. 성능

기존의 Bert 모델보다 더 높은 정확도를 요구한다.

2.2. 대책

1. 제약사항 1. 에 대한 대책

한국어 Machine Reading Comprehension을 위해 만든 dataset Korquad를 사용한다.

2. 제약사항 2. 에 대한 대책

대용량 검색 처리를 위한 inverted index 알고리즘과 검색 키워드에 가장 부합하는 문서를 검색 결과 최상위에 배치하는 알고리즘 TF-IDF을 사용하여 구현한다.

3. 제약사항 3. 에 대한 대책

추출한 답변을 질문과 concatenation하여 다시 검색데이터로 사용한다.

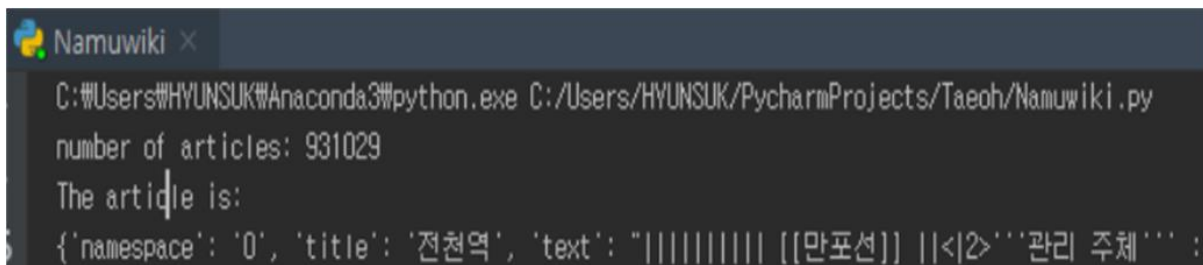
3. 설계 상세화 및 변경내역

3.1 설계 상세화

a. 데이터 전처리

1. Read Dump file

Reading DB Dump File Python의 json 라이브러리를 이용해 NamuWiki의 dump파일을 읽어온다.



```
Namuwiki x
C:\Users\HYUNSUK\Anaconda3\python.exe C:/Users/HYUNSUK/PycharmProjects/Taoh/Namuwiki.py
number of articles: 931029
The article is:
{'namespace': '0', 'title': '전천역', 'text': "||||||| [[만포선]] ||<|2>''관리 주체'' :
```

2. Data Exploration

NamuWiki의 dump파일의 크기와 각 문서의 최소 길이와 최대길이를 구한다.

```
min text size: 0
max text size: 426441
!
#redirect 느낌표
```

3. Test Parsing

Dump파일이 잘적재되었는지 몇몇 기사의 title, text를 출력한다.

```
#redirect 노필요

!!아앗!!
[[파일:3444050440.jpg]]
{{{신 세계수의 미궁 2}}}에서 뜬 !!아앗!!
{{{+! ! ! ああっと ! ! }}}

[[세계수의 미궁 시리즈]]에 건통으로 등장하는 대사, [[세계수의 미궁 2 제왕의 성배|2편 제왕의 성배]]부터 등장했으며, 훌륭한 [[사양 플러그]]의 예시이다.

세계수의 모험가들이 탐험하는 던전인 수해의 구석구석에는 채취/별채/채굴 포인트가 있으며, 이를 위한 채집 스킬에 투자하면 제한된 채집 기회에 보다 큰 이득을 얻을 수 있다.

1. 채집용 캐릭터들로 이루어진 약한 파티(ex: [[레인저<세계수의 미궁 2>|레인저]] 5명)가 수해에 입장한다.
1. 월드 전투를 회피하면서 채집 포인트에 도착해 열심히 아이템을 캐는 중에...
1. ...!!아앗!!... 수라클레시아가 나타났다!--
1. 이때 등장하는 것은 [[F0E<세계수의 미궁 시리즈>|F0E]]는 아닌 일단 월드 출제지만, 훨씬 위 층에 등장하는 강력한 출제이며 선 텀을 빼앗긴다!...
1. ...[[오알 죽음|죽음]]...(hage)

작품마다 !!아앗!!의 세세한 모습은 다르다. 그 약탈함은 첫 등장한 작품이자 시리즈 중에서도 불친절하기로 정평이 난 2편이 절정이었는데, 그야말로 위의 [[세계수의 미궁 3 성배의 내방자|3편]], [[세계수의 미궁 4 견승의 거신|4편]]에는 승룡이 트이게도 채집 중 낮은 확률로 "좋은 아이템을 얻을 수 있을 것 같아"
[[신 세계수의 미궁 밀레니엄의 소녀|신 세계수의]] [[신 세계수의 미궁 2 파프니르기사|미궁 시리즈]], 그 이후에 나온 최신판 [[세계수의 미궁 5 오렌 신화]]

세계수 시스템을 기반으로 한 [[페르소나 시리즈]]와의 콜라보 작품 [[페르소나 0 새도우 오브 더 레버런스]]에도 물론 등장. 3, 4편과 같이 파워 스팟에서 채

여담으로, 2편에서 채집 도중 !!아앗!!이 볼 확률은 [[http://www.atlusnet.jp/topic/detail/310]고작 1%였다고 한다.]] 낮아보이는 확률이어도 플레이 중 한

그 기원은 1인칭 던전 크롤러의 원조 [[워저드리]]에서 합정을 건드렸을 때 나오는 대사 Oops!(<ああっと !>)라고 한다.

[[분류:세계수의 미궁 시리즈]]
```

4. Preprocessing with RegEx

Regular Expression을 이용하여 불필요한 문자, 기호들을 제거해준다.

```
!!아앗!!

신 세계수의 미궁 2에서 뜬 아앗

세계수의 미궁 시리즈에 건통으로 등장하는 대사, 2편 제왕의 성배부터 등장했으며, 훌륭한 사양 플러그의 예시이다.

세계수의 모험가들이 탐험하는 던전인 수해의 구석구석에는 채취별채굴 포인트가 있으며, 이를 위한 채집 스킬에 투자하면 제한된 채집 기회에 보다 큰 이득을 생길 수 있다. 그러나 분배할 수 있는 스킬

1. 채집용 캐릭터들로 이루어진 약한 파티ex 레인저 5명이 수해에 입장한다.
1. 월드 전투를 회피하면서 채집 포인트에 도착해 열심히 아이템을 캐는 중에...
1. 아앗
이때 등장하는 것은 F0E는 아닌 일단 월드 출제지만, 훨씬 위 층에 등장하는 강력한 출제이며 선 텀을 빼앗긴다
1. 퍽알hage

작품마다 아앗의 세세한 모습은 다르다. 그 약탈함은 첫 등장한 작품이자 시리즈 중에서도 불친절하기로 정평이 난 2편이 절정이었는데, 그야말로 위의 아앗 시리즈 그대로, 문지도 따지지도 않고 채집한
3편, 4편에는 승룡이 트이게도 채집 중 낮은 확률로 좋은 아이템을 얻을 수 있을 것 같지만... 주변에서 몬스터들의 기척이 느껴진다.는 메시지가 뜨고 이때 윤이 풍으면 레어 아이템을 얻을 수 있지만
신 세계수의 미궁 시리즈, 그 이후에 나온 최신판 5편에서는 채집 방식이 한 턴으로 끝나는 구조로 바뀐 덕분인지 강제 조우로 다시 회귀해버렸다.... 그나마 위험감지 역통과 같은 버그성 난점들은 수정

세계수 시스템을 기반으로 한 페르소나 시리즈와의 콜라보 작품 페르소나 0 새도우 오브 더 레버런스에도 물론 등장. 3, 4편과 같이 파워 스팟에서 채집 도중 메시지가 뜨고, 실패하면 파티에 참가하고

여담으로, 2편에서 채집 도중 아앗이 볼 확률은 고작 1%였다고 한다. 낮아보이는 확률이어도 플레이 중 한번이라도 일어나는 것을 경합하는 채강 확률을 고려해서 확률을 설정한다고.

그 기원은 1인칭 던전 크롤러의 원조 워저드리에서 합정을 건드렸을 때 나오는 대사 Oops라고 한다.
```

b. 딥 러닝 기반 자연어 처리모듈

1. KorQuAD Open-Domain Setting & Evaluation



“KTX의 호남선이 개통 된 날짜는 언제인가?”

-> “2004년 4월 1일”

검색 엔진 결과(상위 K=5개의 문서)

-한국고속철도 -> KorQuAD에서 포함된 문서

-한국철도공사

-경부선

-경부고속철도

-시흥연결선

한국고속철도 문서의 연관성 라벨을 1로 설정, 나머지 4개의 문서에 만약 정답에 해당하는 단어가 존재한다면, 그 문서들의 연관성 라벨도 1로 설정

랜덤으로 K개의 문서를 뽑아내고, 그 문서들의 연관성 라벨을 1로 설정 => 학습 데이터 구성

Evaluation에서는 위키백과에서 검색 된 상위 K개의 문서와, 나무위키 문서에서 검색 된 상위 K개의 문서를 입력으로 사용하여 정답을 추출하고 정확도를 평가

=> Evaluation에서는 위키백과 이외에 나무위키의 문서를 함께 사용하는 이유는 학습된 Open-Domain QA 모델이 학습 때 사용되지 않은 처음 보는 문서(Knowledge)도 잘 활용하여 추론을 할 수 있는지 확인하고, 실제 시연에서 더 다양한 주제에 답변할 수 있는 시스템을 구축하기 위해서이다

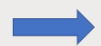
2. Re – Ranking passage Aggregation



이병헌이 주연으로 출연한 사극 영화 제목이 뭐야?

-생명의 위협을 느끼게 되는 왕 광해(**이병헌**)는
도승지 허균(**류승룡**)에게 지시하여 자신의 대역을 찾게 하였다.
그리고 허균은 ... (생략)

추출 된 정답



광해
(3번 Paragraph와 합침)

-**광해군** 치세기 당시 실존 인물을 빌려와 **승정원일기**에서
지워진 15일간의 빈 시간 사이에 광해군으로 위장한 대역이
조선을 다스렸다고 가정한 **팩션** 영화이다 (생략)



광해군
(그대로 사용)

-궁의 혼란이 극에 달한 시절. 암살과 역란의 공포에
괴박해져 가던 **왕 광해**는 급기야 자신의 대역을 구하기에
이르고, 그렇게 저잣거리 천민의 아찔한
왕 노릇이 시작된다..... (생략)



왕 광해
(1번 Paragraph와 합침)



이병헌이 주연으로 출연한 사극 영화 제목이 뭐야?광해



추출된 정답
후보와 질의를 합침

-생명의 위협을 느끼게 되는 왕 광해(이병헌)는
도승지 허균(류승룡)에게 지시하여 자신의 대역을 찾게 하였다.
그리고 허균은 ... (생략)
궁의 혼란이 극에 달한 시절. 암살과 역란의 공포에
괴박해져 가던 왕 광해는 급기야 자신의 대역을 구하기에
이르고. 그렇게 저잣거리 천민의 아찔한
왕 노릇이 시작된다..... (생략)



Aggregation이 된 Paragraph

3.2 변경 내역

- 검색엔진을 구현하는 대신 연구실에서 제공하는 검색 엔진을 가공하여 사용한다.
- 기존 Bert의 기계독해 성능보다 높은 정확도를 위해 추출한 답변을 질문과 concatenation하여 검색데이터로 사용한다.
- Re - Ranking passage Aggregation을 적용

4. 갱신된 과제 및 추진계획

월 ↙ 진행 항목 ↘	8	9	10
검색엔진 사용법 익히기			
중간보고서 작성	완료		
자연어 처리모델			
데이터 정제	완료		
최종보고서 작성			
Test 및 보완			
졸업과제 발표			
결과물 제출/ SW등록			

5. 데이터셋 수정사항

데이터셋 이름	사용 될 모듈	데이터 구성	데이터크기
위키피디아	검색엔진	위키피디아 문서 제목 및 내용	1320446개 문장
KorQuad	딥러닝 자연어 처리 모듈	질문 – 질문의 주제에 해당하는 위키피디아 문서 제목 – 해당 위키피디아 문서내에서 정답에 해당하는 단어	10,645건의 문단 66,181개의 질의응답
나무위키	검색엔진	나무위키 문서 제목 및 내용	931,029 문서

데이터셋	타입	특징
SQuAD	Machine Comprehension	
Quasar-S (Quasar-T)	Open-domain	[질문,답]과 함께 50개의 관련 문서들을 제공
WikiMovies	Factoid (Open-domain)	[질문, 답]을 제공
CuratedTREC	Factoid (Open-domain)	[질문, 답]을 제공
WebQuestion	Factoid (Open-domain)	[질문, 답]을 제공
SQuAD open[5]	Factoid (Open-domain)	SQuAD 데이터셋을 Open-Domain으로 세팅한 데이터셋
KoQuAD	Machine Comprehension	영문 데이터를 기계 번역한 데이터셋(노이즈가 많음)
KorQuAD	Machine Comprehension	사람이 한국어 위키를 통해 직접 태깅한 데이터셋

6. 구성원 별 진척도

구성원	역할	진행상황
신우창	- 검색엔진 개량 - 클라이언트 개발	검색엔진 학습 및 트레이닝 데이터 구동 중
김윤호	- 자연어 처리 모델 구현 - 질의 응답 모델 구현 - 모델 학습 및 최적화	BERT 모델 학습 및 구현 중
이태오	- 서버 개발 - 나무위키 데이터 정제	데이터 정제 완료

7. 과제 수행 내용 및 중간 결과

7.1. 데이터 전처리

	데이터 정제 전	데이터 정제 후
문서의 양(개수)	931029	655452
파일 크기	7,422,121KB	1,491,224KB

7.2. Bert 실험 결과

7.2.1. korquad

Google의 **Multilingual Model**(음절기반)과
연구실에서 제공받은 **형태소 기반**으로 학습 된 BERT 모델 2가지로 실험1

토큰화 할 문장: 나는 집으로 갔다
구글 모델: [나, ##는, 집, ##으로, 갔, ##다]
형태소 모델: [나, 는, 집, 으로, 갔, 다]

두 모델의 모델에 대한 parameter는 **Vocabulary Size**를 제외하고는 모두 같다

구글 모델: 98791개 (다른 언어의 문자도 다수 포함되어 있다)
형태소 모델: 128000개 (한글의 형태소로만 이루어져 있다)

7.2.2 korquad 성능측정

실험 세팅

모델	배치 사이즈	최대 길이	Clip Gradient
구글 모델	5	400음절	X
형태소 모델	7	300형태소	O

실험 결과

	구글 모델	형태소 모델
Baseline	85.5%	87.3%
형태소 정보 사용	87.1%	89.2%

<Evaluation은 F1 Score를 기반으로 측정>

System	Seq Length	Max Batch Size
BERT-Base	64	64
...	128	32
...	256	16
...	320	14
...	384	12
...	512	6

<입력 길이에 따른 최대 배치 사이즈 개수>
GPU Memory 12G

7.2.3. Qausar-T

R3 Net, BERT Model(google English baseline Model)을 사용하여 Fine-Tuning하여 실험을 진행
(R3 Network는 Match LSTM 기반의 모델)

SQuAD를 학습한 방법과 동일하게 Start Index와 Stop Index를 예측하는 Classifier와 함께,
해당 문장에 정답이 포함되어 있을지 아닐지(연관성)를 예측하는 Rank Classifier를 추가함
(Rank Classifier는 [CLS] 토큰의 임베딩을 통해 0 ~ 1의 확률 값을 출력: 1이면 연관 문단)

방법: Rank Classifier의 확률 값에 라벨을 부여하고, Start, Stop, Rank의 3가지의

Loss를 줄이도록 학습

7.2.4 성능측정

	EM	F1
Match-LSTM	29.1%	35.1%
BERT-baseline	41.5%	43.9%

<실험 결과>

Rank	Method	EM (Quasar-T)	F1 (Quasar-T)
1	Evidence Aggregation via R ³ Re-Ranking	42.3	49.6
2	Denoising QA	42.2	49.3
3	DecaProp	38.6	46.9

<State-of-Art 결과물들의 실험결과>

현재 실험결과는 Re-Ranking 기법이 전혀 적용되지 않았기 때문에 추후 학습 데이터를 더 잘 정제하고 Re-Ranking 기법을 적용한다면 SOTA에 더 가까운 결과도 가능할 것 이다.