

Open Domain Question Answering Task

3등 솔루션 공유



1조 : 강범서, 오피훈, 이예진, 한기백, 정유리

목차

1. 대회 개요 & 외부데이터

2. Retriever

a. Sparse Retriever

b. Dense Retriever

3. Reader

a. Data Augmentation

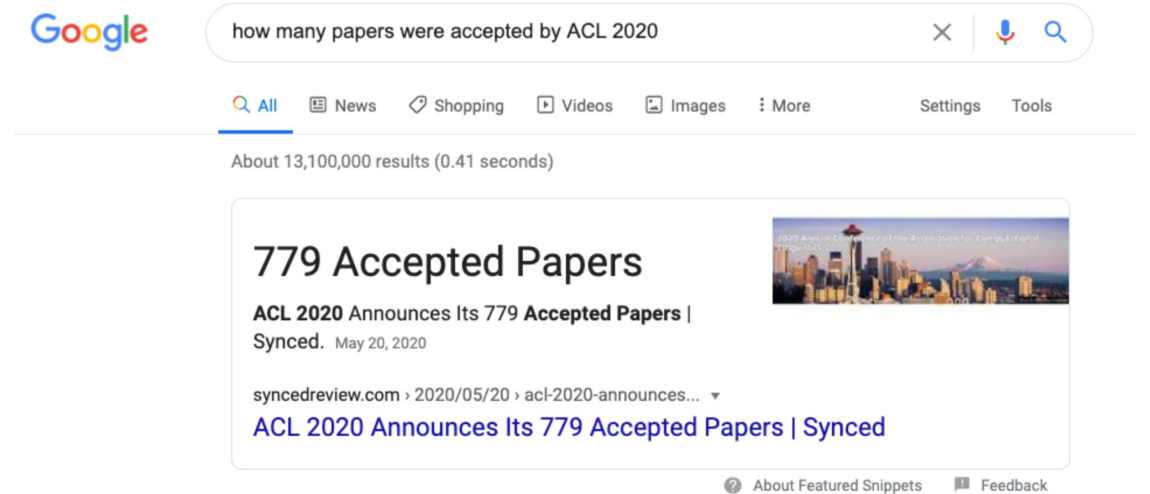
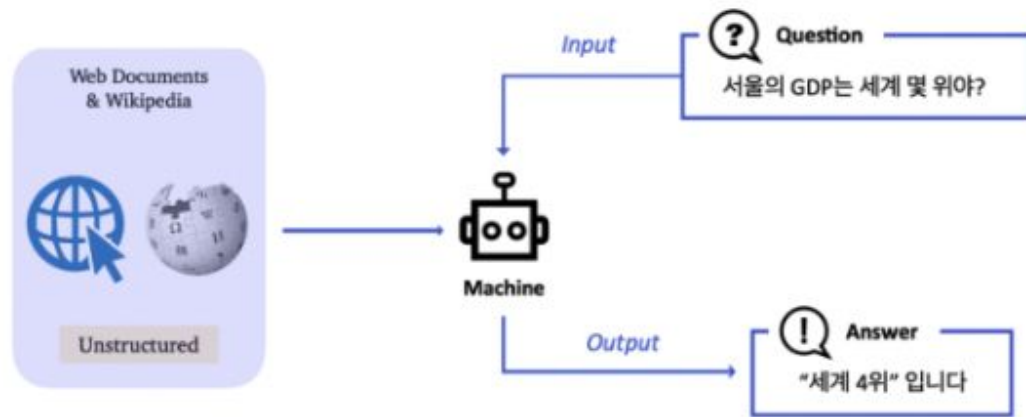
b. Hyper-parameter Tuning

4. Retriever + Reader

5. Ensemble

6. Conclusion

대회 개요



<https://github.com/danqi/acl2020-openqa-tutorial>

- 방대한 World Knowledge (Knowledge Base, Tables, Web documents & Wikipedia) 을 바탕으로 사용자의 질문에 답하는 Task
- 대회는 주어진 한국어 wikipedia documents를 바탕으로 retriever, reader two-stages 단계로 답변을 추출
- 이전 대회와의 차이점 - **KLUE-MRC 데이터셋을 제외한 모든 외부 데이터 사용 허용**

외부 데이터

1. **KorQuAD 1.0 - 한국어 Machine Reading Comprehension을 위해 만든 데이터셋** (출처: [KorQuAD 1.0](#))
 - a. KorQuAD_v1.0_dev.json
 - i. context : 964, question : 5774
 - b. KorQuAD_v1.0_train.json
 - i. context : 9681, question : 60407
2. **일반상식 위키백과 데이터셋 질문 - 정답 쌍** (출처: [AIHUB 일반상식](#))
 - a. ko_wiki_v1_squad.json
 - i. context : 68538, question : 100268
3. **기계독해 위키백과 데이터셋 질문 - 정답 쌍** (출처: [AIHUB 도서자료 기계독해](#))
 - a. ko_nia_normal_squad_all.json
 - i. context : 47314, question : 243425
 - b. ko_nia_noanswer_squad_all.json : 답변 불가능한 질문 추가
 - i. context : 20030, question : 100244
 - c. ko_nia_clue0529_squad_all.json : factoid 뿐만아니라 why, how 등 설명가능 질문 추가
 - i. context : 34500, question : 96663

너무 많은 Dataset, 어떤 데이터를 선택해야 할까?

기준1. annotation bias 를 줄이자

질문하는 사람이 답을 알고 질문!

-> 어휘 중복

context 없이 질문이 성립 안함!

-> 마지막 소토마요르를 포함 몇 명의 선수들이 결승을 다투었나?

여러분, 혹시 그거 아시나요? 소토마요르는 결승전만 적어도 24번 진출한 높이뛰기 선수입니다...

기준2. DPR 의 in-batch-negative sampling 을 고려하자

in-batch-negative sampling 시

서로 다른 question 에 대해 정답 context 가 일치하고 해당 질문들이 같은 배치에서 훈련시

negative context 와 positive context 가 같아져 훈련이 어려워짐.

-> 이상적으로 context 과 question의 비율이 같으면 좋음

검증. top-20 elasticsearch accuracy

korquad v1 : 80.39, AI HUB 일반상식 : 59.88 -> 어휘 중복 낮음

위의 기준에서 1) 질문과 context의 어휘 중복이 낮고, 2) 비교적 질문이 context에서 독립적이고, 3) context와 question의 비율이 비슷한

일반상식 위키백과 데이터셋 질문 - 정답 쌍 (출처: [AI HUB 일반상식](#))

dpr, mrc에서 대회 제공 데이터와 함께 baseline data로 선택하여 진행

Sparse Retriever

다양한 Retriever & Tokenizer 종류

Retriever

- TF-IDF, BM25, BM25L, etc
- TF-IDF, BM25

Tokenizers

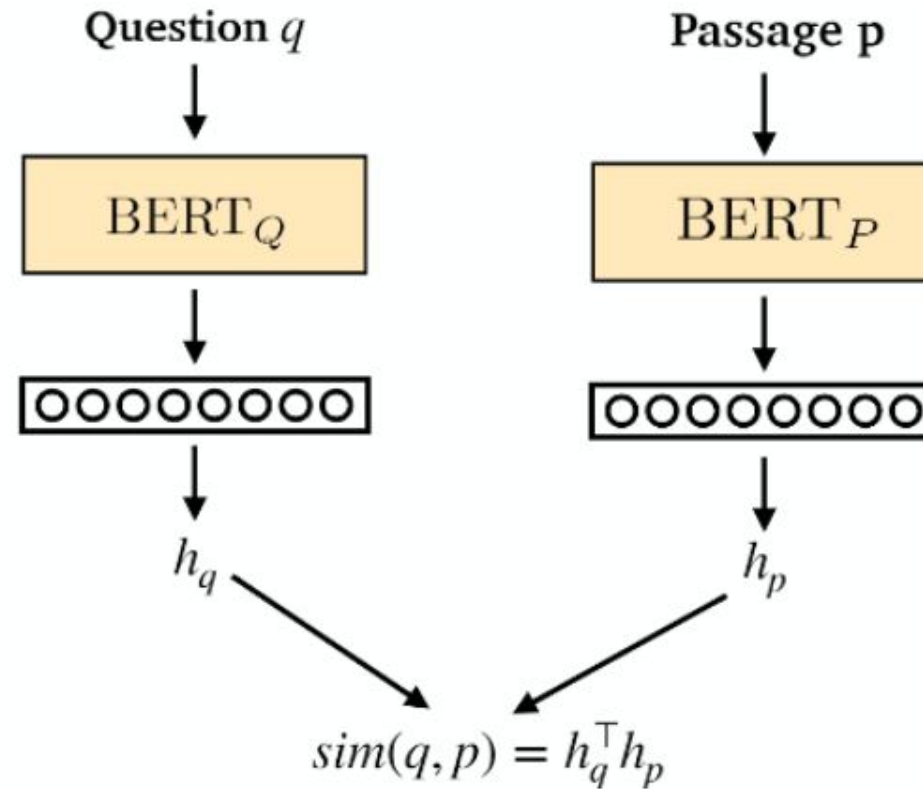
- Transformer Tokenizers, Mecab, Okt, etc
- Mecab, bert-base-multilingual-cased, klue/roberta-large, monologg/koelectra-base-v3-discriminator

Elasticsearch (BM25 + nori-tokenizer)

결과 Elasticsearch - top20 **accuaray** : 0.950381679389313

Dense Retriever

Dense Passage Retrieval for Open-Domain Question Answering(2020)



Dense Retriever

Dense Passage Retrieval for Open-Domain Question Answering(2020)

많은 걸 바꿔야 했음

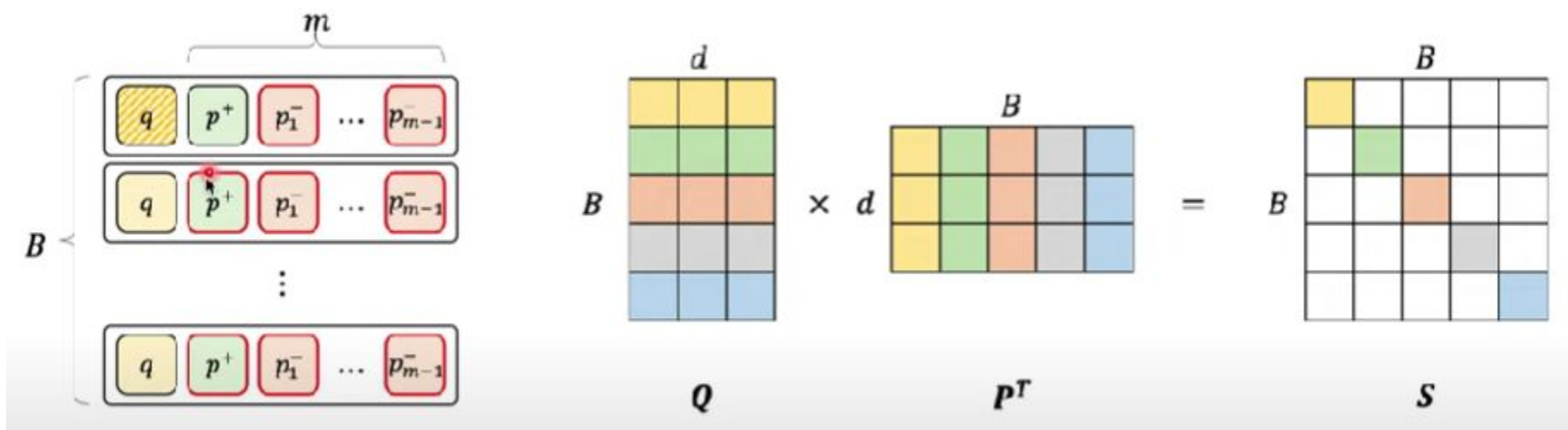
- 지금까지 다룬 모델들과 달리 Encoder를 2개를 사용하는 구조이므로 모델을 새로 정의해 주었어야 했으며, 2개를 결합하여 훈련하여야 하므로 그에 따라 loss도 새로 만들어줬어야함.
- 또한, in-batch negative training 방식은 batch 내에서 negative를 활용하는 방식으로, input에는 negative에 대한 label이 따로 존재하지 않았음. 따라서 label 없이 훈련 가능하도록 trainer도 새로 만들어줬어야함.

Dense Retriever

Dense Passage Retrieval for Open-Domain Question Answering(2020)

훈련

In-batch Negative



Dense Retriever

Dense Passage Retrieval for Open-Domain Question Answering(2020)

Model klue/roberta-large

Loss 다음과 같은 NLL loss를 활용함.

$$L(q_i, p_i^+, p_{i,1}^-, \dots, p_{i,n}^-) = -\log \frac{e^{\text{sim}(q_i, p_i^+)}}{e^{\text{sim}(q_i, p_i^+)} + \sum_{j=1}^n e^{\text{sim}(q_i, p_{i,j}^-)}}$$

Dense Retriever

Dense Passage Retrieval for Open-Domain Question Answering(2020)

결과

한 batch 내에서의 정확도는 기대감을 부풀게 하였음.

(약 **97%**의 정확도를 보임. batch_size=8 기준)



Dense Retriever

Dense Passage Retrieval for Open-Domain Question Answering(2020)

그러나...

Top-1 : 33%
Top-5 : 61%
Top-10 : 70%
Top-20 : 79%
Top-30 : 83%
Top-50 : 89%
Top-70 : 92%
Top-100 : 94%



전체 training data(약 3952개)에서의 Top-20을 기준으로 79%정도의 정확도를 보임

당시의 Sparse Retriever는 wikipedia_documents.json(56737개) Top-20을 기준으로 93%정도의 정확도임에 비해 성능이 많이 떨어지는 것으로 관찰되었음..

예측 실패 원인 분석

▼ '덜', '더' 등의 비교

```
<prediction is in context, but not answer!>
index: 27
(context)
포르투갈어와 유사한 스페인 갈리시아 지방의 갈리시아어가 존재하나, 포르투갈어의 표준규범은 브라질 포르투갈어와 유럽 포르투갈어를 들 수 있다.
두 포르투갈어 표준은 발음 면에서 현저한 차이의 변화가 있으며, 문법 면에서는 대명사 체계에 차이점이 존재한다. 또한 브라질 포르투갈어는, 문
법 면에서 덜 보수적으로 간주된다. 그 결과 두 포르투갈어 화자 사이에 이전의 노출 경험 없이 커뮤니케이션을 취하려 할 경우, 특히 브라질 포
르투갈어 화자가 유럽 포르투갈어를 접할때에 어려움이 발생한다. 브라질의 텔레노벨라의 오랜 기간 동안의 유럽 지역의 보급으로, 포르투갈의 포르
투갈어 화자는 브라질 포르투갈어의 엑센트 및 특정 단어의 이해 면에서 덜 어려움을 겪는다.날짜=2017-11-09\n\n브라질 포르투갈어와 유럽 포
르투갈어는 현재 별도의 유사한 철자 표준을 각각 두고 있다. 이 두 표준의 통합 표기법(제한된 수의 단어는 이중 맞춤법을 허용)에 브라질 및 포
르투갈 정부에 의해 승인되어, 현재는 공식적인 표준으로 존재한다. 공식 표기 표준은 문법적으로 서로 유사하나 일부 사소한 문법적 차이점은 존재
한다.\n\n이 외에 아프리카의 포르투갈어 사용 지역에서 사용되는 아프리카 포르투갈어 와 마카오, 동티모르 등의 아시아 포르투갈어 는 유럽 표준
을 바탕으로 하고 있으나, 각자 특유의 발음 및 문법이 발전되었다.
(question) 문법 측면에서 더 보수적인 포르투갈어 표준은?
(answer) {'answer_start': [61], 'text': ['유럽 포르투갈어']}
(prediction) 브라질 포르투갈어
```

- “또한 브라질 포르투갈어는, 문법 면에서 덜 보수적으로 간주된다.” → context 전체가 '유럽 포르투갈어'와 '브라질 포르투갈어'의 차이에 대해 논하고 있고, 브라질 포르투갈어가 '덜 보수적'이라는 문구가 있으므로 '더 보수적인 포르투갈어'는 '유럽 포르투갈어'로 예측해야 함.
- 아마도 문장 내에 '(유럽 포르투갈어보다) 덜' 등의 표현이 생략되었기 때문에 단순히 '브라질 포르투갈어'라는 문구와 '보수적'이라는 단어만 연결해서 잘못된 예측을 하지 않았나 싶음.
 - '덜', '더'를 포함한 비교급 표현들이 포함된 context, question, answer를 구해서 data augmentation 후 학습을 진행하면 잘 맞출 수 있지 않을까 싶습니다. 혹은 함축적인 의미를 잘 파악하는 모델을 사용..?

```
<prediction is in context, but not answer!>
index: 115
(context)
《뒤러의 코뿔소》(Dürer's Rhinoceros)는 알브레히트 뒤러가 르네상스 시대인 1515년에 제작한 목판화이다. 그림은 1515년 초 리스본에 도착한 인도코뿔소를 보고 어느 화가가 묘사한 글과 간단한 스케치를 바탕으로 제작되었다. 뒤러의 목판화는 코뿔소를 정확하게 표현하지 못했다. 뒤러는 코뿔소의 목에는 고지트, 가슴 쪽에는 흉갑, 그 사이를 잇는 리벳이 있는 갑옷 같이 딱딱한 판으로 몸이 둘러싸인 동물로 묘사하였다. 또, 코뿔소의 등에 뒤틀린 작은 불을 그렸고, 비늘로 뒤덮어진 다리와 톱과 같이 뾰족한 엉덩이도 그렸다. 하지만 실제로 이러한 특징은 코뿔소에서 찾아볼 수 없다. 이런 구조적 오류에도 불구하고, 뒤러의 코뿔소는 유럽에서 매우 유명해졌고, 그 후 3세기 동안 많은 복사본들이 판매되었다. 서양인들은 18세기 후반이 될 때까지 그것을 진짜 코뿔소의 모습으로 생각했다. 결국은 1740년대에서 1750년대 사이에 유럽을 돌아다닌 클라라는 코뿔소를 그린 그림과 같은 더 현실적이고 세밀한 그림들이 뒤러의 코뿔소를 대신하게 되었다. 뒤러의 목판화는 "아마 어떤 동물 그림도 예술에 이 정도로 큰 영향을 끼친 적이 없을 것이다."라고 평가된다. \n\n한편, 로마 제국 시절 이후 유럽에서 살아 있는 코뿔소가 목격된 것은 처음이었는데, 뒤러는 그려진 코뿔소를 실제로 보지 못했다. 그려진 코뿔소는 1515년 말에 포르투갈의 왕인 마누엘 1세가 교황 레오 10세에게 선물로 보냈던 것이나, 1516년 초에 이탈리아의 해안에서 난파를 당해 죽었다. 그 이후로 1577년 인도에서 포르투갈의 세바스티앙의 궁중으로 와 1580년 경에 스페인의 펠리페 2세에게 상속된 아바다라는 코뿔소가 올 때까지 유럽에서 살아있는 코뿔소를 다시 볼 수 없었다.
(question) 뒤러는 어디에 코뿔소를 그렸는가?
(answer) {'answer_start': [61], 'text': ['목판']}
(prediction) 목판화

<prediction is in context, but not answer!>
index: 120
(context)
서울 시내에서 고급 병원을 운영하고 있는 이인국 박사는 자신의 병원에서 가난한 환자들은 받지 않고 병원비도 다른 병원의 두 배를 받으면서 부유층과 권력층 등 돈 있는 환자만 받는다. 이박사는 막 수술을 마치고 수술이 성공한 것 같지 않은 개운치 않은 상태에서 미국 대사관의 브라운씨를 만나러 간다. 그는 광복 전까지 힘 있는 일본인 만을 치료하면서 부유한 생활을 해왔지만 1945년 일본이 패망하고 한반도가 일제로부터 해방되자 평양에는 북쪽에서 내려온 소련군이 진주한다. 이후 이박사는 자신이 치료를 거절했던 춘석의 고발로 친일 혐의로 잡혀간다. \n\n감옥에서 이인국 박사는 매를 맞아 아픈 몸으로도 노어(러시아어) 회화책을 우연히 얻어 러시아어를 공부한다. 감방 안에서 이질이 발생했다는 것을 알고 간수들에게 알린다. 이후 의술을 인정받아 의무실에서 근무하게 된 그는 의무관이었던 스텐코프의 혹을 치료해주고, 그 대가로 감옥에서 풀려나 집으로 돌아가게 된다. 스텐코프의 주선으로 이인국은 아들을 모스크바로 유학까지 보내지만 1950년 한국전쟁이 발발하자 아들과 연락이 끊어진다. \n\n이인국 박사는 전쟁 와중에 아내를 잃고 아들의 생사를 모른 채 1.4 후퇴 때 남한으로 내려온다. 이후 그는 서울 시내에 병원을 차리고 자신의 의술로 부유층과 권력층들만을 상대하면서 돈을 번다. 딸 나미는 미국으로 유학가지만, 외인 교수와 결혼할 예정이다. 미 대사관으로부터 국무부 초청장을 받는 데 성공하자, 그는 자신이 미국에 가서도 반드시 성공하리라는 자신감을 가지고 비행기표를 사러 반도호텔로 간다.
(question) 이인국 박사가 석방되는 결과를 가져온 병은?
(answer) {'answer_start': [444], 'text': ['혹']}
(prediction) 스텐코프의 혹
```

- 뒤러가 코뿔소를 그린 곳은 목판'화'가 될 수 없음.
 - 이러한 시나리오에 대한 data를 더 넣어보면 되지 않을까 싶습니다. (목판과 목판화의 구분)
- 스텐코프의 혹 때문에 이인국 박사가 석방되었지만 석방을 유발한 '병'을 묻는다면 일반적인 병명인 '혹'을 정답으로 보는 것이 맞을 것임.

Reader, Data Augmentation

예측 실패 원인 분석

“특정 시나리오에 대해 예측을 잘하는 모델을 추가하여 앙상블?”
“질문을 수정?”



가장 단순하고 강력한 방법은 ‘데이터 증강’

Reader, Data Augmentation

일반상식 - wiki 본문에 대한 질문 - 답 쌍 (train set : 약 68,600개)

일반상식 소개

<div>소개</div> <div>다운로드</div>			
데이터셋명	일반상식		
데이터 분야	음성/자연어	데이터 유형	텍스트
구축기관	마인즈랩	담당자명	안준환(마인즈랩)
가공기관		데이터 관련 문의처	전화번호 031-625-4349
검수기관		이메일	pworks@mindslab.ai
구축 데이터량	15만	구축년도	2017년
버전	1.0	최종 수정일자	2018.01.02
소개	한국어 위키백과 내 주요 문서 15만 개에 포함된 지식을 추출하여 객체(entity), 속성(attribute), 값(value)을 갖는 트리플 형식의 데이터 75만 개를 구축한 지식베이스 제공.		
주요 키워드	한국어, 위키백과, 일반상식, 지식베이스, WIKI 본문, 질의응답, 챗봇, 지능형 QA 서비스, 위키백과 데이터		
저작권 및 이용정책	본 데이터는 과학기술정보통신부가 주관하고 한국지능정보사회진흥원이 지원하는 '인공지능 학습용 데이터 구축사업'으로 구축된 데이터입니다. [데이터 이용정책 상세보기]		
데이터 설명서	구축활용가이드		
샘플 데이터	교육활용 동영상		
저장도구	AI 모델		

데이터 변경이력

버전	일자	변경내용	비고
1.0	2018.01.02	데이터 최초 개발	

WIKI 본문에 대한 질문-답 쌍	WIKI 본문내용과 관련한 질문과 질문에 대응되는 WIKI 본문 내의 정답 쌍	파일 다운로드
-----------------------	---	------------

```
{'paragraphs': [{'qas': [{'question': 'ave;new 본거지 어디야',  
  'answers': [{'answer_start': 22, 'text': '도쿄 치요다구'}]}  
  'id': '9_f2_wiki_3091-1'}],  
  'context': 'ave;new(아베;뉴, アベニュー)는 도쿄 치요다구에 본 거처를 둔 일본의 음악 제작 그룹이다. ave;new의 프로듀서인 a.k.a.dRESS가 학창시절부터 친구인 네모토 히데미·마츠시타 미유키를 불러 사운드 팀을 시작하자고 제안한 것이 결성의 계기이다. 2003년 7월에 결성된 ave;new는, 주로성인 게임의 주제가나 BGM를 제공하고 있다. 주로 파트너 브랜드의 BGM 및 예능 프로덕션도 담당하고 있으며 CD는 그룹명과 같이 ave;new의 라벨로 판매되고 있으나 다른 라벨에 비해 시장에 약간 유통 하기 어렵다. 또한 일부의 CD는 d;VIRTU(주식회사 디바트)가 판매하고 있기 때문이다. 음악은테크노계나 트랜스계와 같이 밝은 곡으로부터, 재즈나발라드와 같이 조용한 곡까지 폭넓게 제작할 수 있는 것이 특징이다. 특히, 신디사이저를 이용한 테크노계의 곡은 높은 평가를 얻고있다. ave;new의 프로듀서인 a.k.a.dRESS가 학생시절부터의 친구인 네모토 히데미·마츠시타 미유키를 불러, 사운드 팀을 시작하려고 제안한 것이 결성의 계기다. ave;new라고 하는 브랜드명은 "avenue"(길·수단)과 "new"(새롭다)이라고 하는 2개의 단어를 조합한 것으로, 항상 창조의 새로운 길과 수단을 모색한다라고 하는 의미를 담고, a.k.a.dRESS가 이름을 붙였다.'}],  
  'title': 'Ave;new'}
```


Reader, Data Augmentation

KorQuAD 1.0 (train set: 약 65000개)

KorQuAD

1.0 (ENG) 1.0 (한국어)

KorQuAD 1.0

The Korean Question Answering Dataset

What is KorQuAD 1.0?

KorQuAD 1.0은 한국어 Machine Reading Comprehension을 위해 만든 데이터셋입니다. 모든 질의에 대한 답변은 해당 Wikipedia article 문단의 일부 하위 영역으로 이루어집니다. Stanford Question Answering Dataset(SQuAD) v1.0과 동일한 방식으로 구성되어 있습니다.

KORQUAD 1.0 소개 (SLIDE)

KORQUAD 1.0 소개 (PAPER)

Getting Started

KorQuAD 1.0의 전체 데이터는 1,560 개의 Wikipedia article에 대해 10,645 건의 문단과 66,181 개의 질의응답 쌍으로, Training set 60,407 개, Dev set 5,774 개의 질의응답 쌍으로 구분되었습니다.

KorQuAD 1.0의 데이터셋은 CC BY-ND 2.0 KR 라이선스를 따릅니다.

또한 CodaLab을 통한 모델 제출시 테스트 스코어 계산 및 리더보드를 통한 스코어 공개에 동의한 것으로 간주합니다. 참고로 제출한 모델 및 소스 코드 등에 대해서는 참가자가 직접 라이선스를 부여하고 이를 명시할 경우 그에 따릅니다.

TRAINING SET (37MB)

DEV SET (3.9MB)

Leaderboard

KorQuAD 1.0의 Test set으로 평가한 Exact Match(EM) 및 F1 score입니다.

Rank	Reg. Date	Model	EM	F1
-	2018.10.17	Human Performance	80.17	91.20
1	2020.08.24	SDS-XFormer+ (single model) Samsung SDS AI Research	88.10	95.57
2	2022.03.18	h_model (single model) Anonymous gh	87.71	95.39
3	2020.07.13	LGSP-LM-Large V2.0 LG AI NLP Team	87.46	95.39
4	2021.11.03	SKERT-Large 2.0.0 (ensemble) Skelter Labs	87.94	95.25
5	2021.12.02	InfoLab KorLM v0.4 (single model) KAIST InfoLab	88.17	95.24
6	2021.12.07	SKERT-Large 2.0.1 (ensemble) Skelter Labs	87.58	95.18
7	2021.09.09	InfoLab KorLM v0.3 KAIST InfoLab	87.79	95.16
8	2020.01.08	SKERT-Large (single model) Skelter Labs	87.66	95.15
9	2020.11.09	Americano (single) SK Planet RB Dialogue Team(JunSeok Kim)	86.81	95.13

```
▼ data: [] 1420 items
▼ 0:
▼ paragraphs: [] 3 items
▼ 0:
▼ qas: [] 8 items
▼ 0:
  ► answers: [] 1 item
    id: "6566495-0-0"
    question: "바그너는 괴테의 파우스트를 읽고 무엇을 쓰고자 했는가?"
▼ 1:
  ► answers: [] 1 item
    id: "6566495-0-1"
    question: "바그너는 교향곡 작곡을 어디까지 쓴 뒤에 중단했는가?"
► 2:
► 3:
► 4:
► 5:
► 6:
► 7:
context: "1839년 바그너는 괴테의 파우스트를 처음 읽고 그 내용에 마음이 끌려 이를 소재로 해서 하나의 교향곡을 쓰려는 뜻을 갖는다. 이 시기 바그너는 1838년에 빛 독촉으로 산전수전을 다 겪은 상황이라 좌절과 실망에 가득했으며 메피스토펠레스를 만나는 파우스트의 심경에 공감했다고 한다. 또한 파리에서 아브네크의 지휘로 파리 음악원 관현악단이 연주하는 베토벤의 교향곡 9번을 듣고 깊은 감명을 받았는데, 이것이 이듬해 1월에 파우스트의 서곡으로 쓰여진 이 작품에 조금이라도 영향을 끼쳤으리라는 것은 의심할 여지가 없다. 여기의 라단조 조성의 경우에도 그의 전기에 적혀 있는 것처럼 단순한 정신적 피로나 실의가 반영된 것이 아니라 베토벤의 합창교향곡 조성의 영향을 받은 것을 볼 수 있다. 그렇게 교향곡 작곡을 1839년부터 40년에 걸쳐 파리에서 착수했으나 1악장을 쓴 뒤에 중단했다. 또한 작품의 완성과 동시에 그는 이 서곡(1악장)을 파리 음악원의 연주회에서 연주할 파트보까지 준비하였으나, 실제로는 이루어지지 않는 않았다. 결국 초연은 4년 반이 지난 후에 드레스덴에서 연주되었고 재연도 이루어졌지만, 이후에 그대로 방치되고 말았다. 그 사이에 그는 리엔치와 방황하는 네덜란드인을 완성하고 탄호이저에도 착수하는 등 분주한 시간을 보냈는데, 그런 바쁜 생활이 이 곡을 잊게 한 것이 아닌가 하는 의견도 있다."
```

```
► 1:
► 2:
title: "파우스트_서곡"
```


Reader, Data Augmentation



1	Finished	39.5800	51.6700
---	-----------------	---------	---------



Finished	45.8300	60.2400
-----------------	---------	---------

Reader, Hyper-parameter Tuning

RoBERTa: A Robustly Optimized BERT Pretraining Approach

bsz	steps	lr	ppl	MNLI-m	SST-2
256	1M	1e-4	3.99	84.7	92.7
2K	125K	7e-4	3.68	85.2	92.9
8K	31K	1e-3	3.77	84.6	92.8

특정 지점까지는 **batch size**를
늘리는 것이 성능 향상에 도움이 됨

Table 3: Perplexity on held-out training data (*ppl*) and development set accuracy for base models trained over BOOKCORPUS and WIKIPEDIA with varying batch sizes (*bsz*). We tune the learning rate (*lr*) for each setting. Models make the same number of passes over the data (epochs) and have the same computational cost.

Reader, Hyper-parameter Tuning

```
training_args.gradient_accumulation_steps = 16  
training_args.per_device_train_batch_size = 8
```

‘gradient_accumulation_steps’를 이용하여
다양한 batch size 실험(8, 16, 32, 64, 128, 256)

batch size 128 선택

Finished	55.8300	68.9000
Finished	58.3300	71.2800
Finished	57.5000	69.4500

klue/roberta-large

Finished	41.2500	54.0600
Finished	35.4200	49.7500
Finished	30.4200	42.9700

koelectra-base

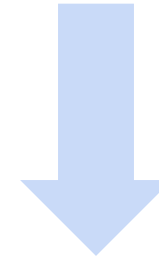
Reader, Hyper-parameter Tuning



Finished

48.7500

61.5200

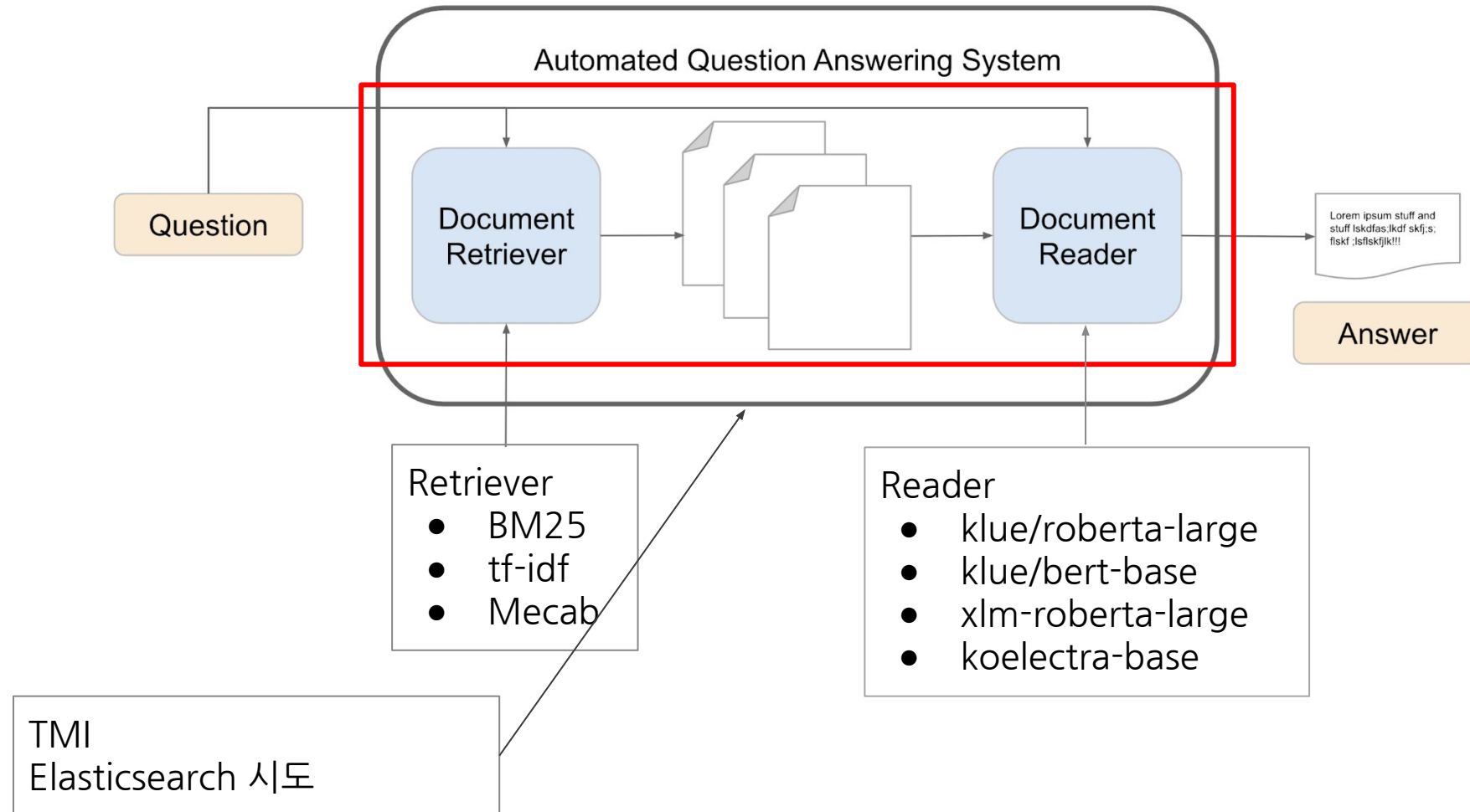


Finished

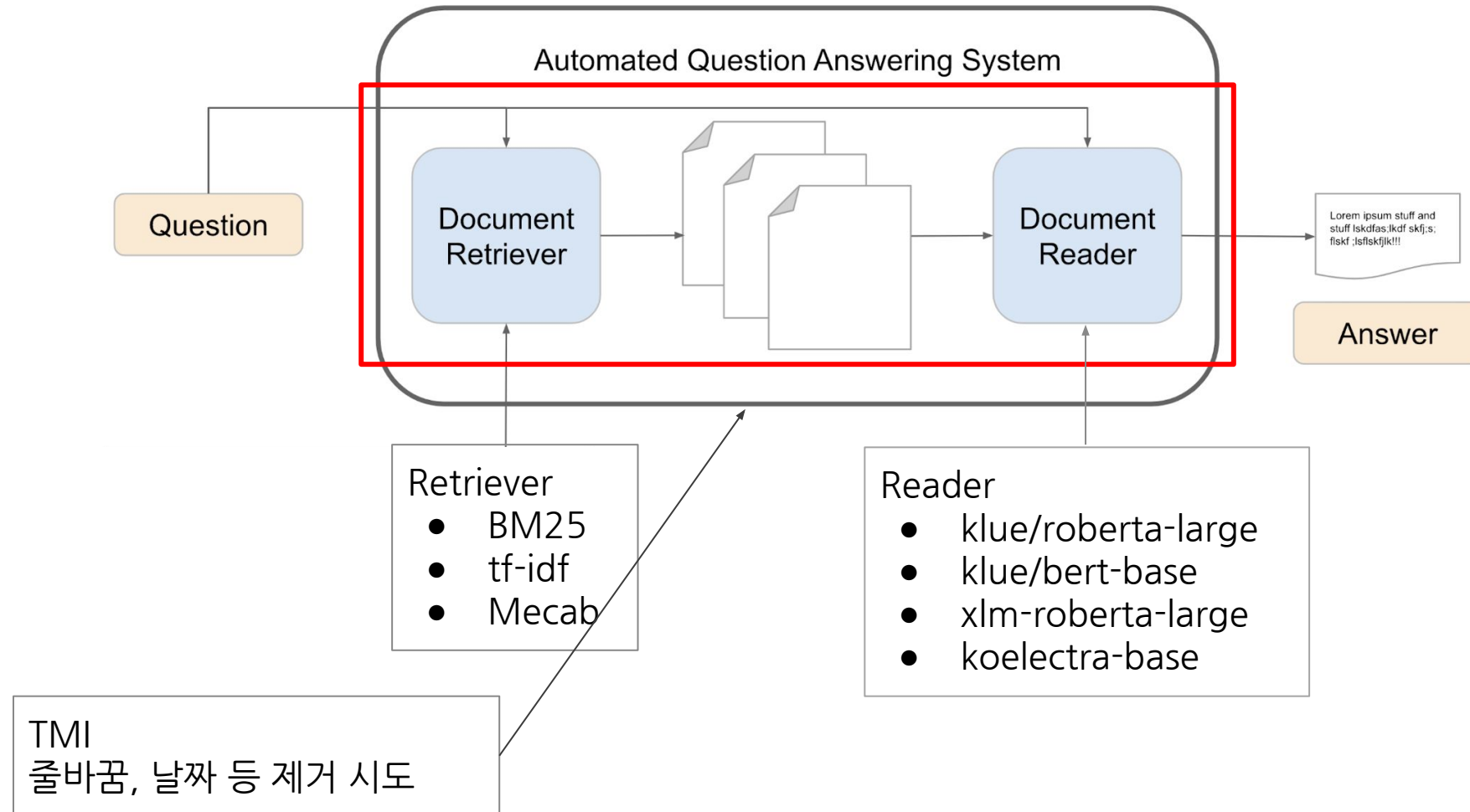
58.3300

71.2800

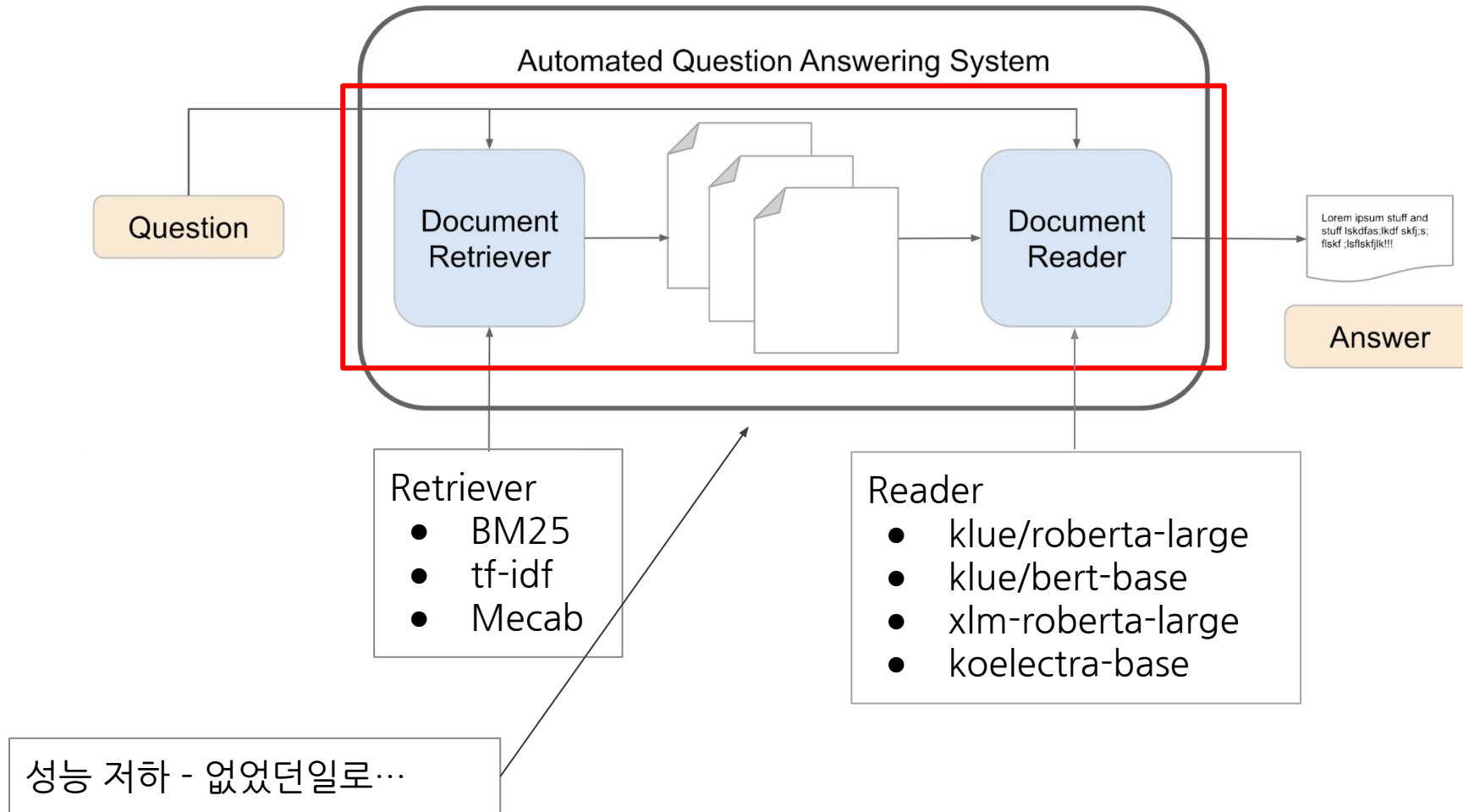
Retriever + Reader



Retriever + Reader



Retriever + Reader



Ensemble

soft-voting

- 각 prediction에 대한 probability를 모두 더한 후 가장 높은 probability를 갖는 prediction text 제출
- '특정 text에 대해 각 모델이 얼마나 확신을 가지고 있는지', '몇 개의 모델이 해당 text를 예측했는지'를 모두 반영

```
def preprocessing(nbests):
    ids = nbests[0].keys()
    total_nbest = {}

    for id_ in ids:
        total_nbest[id_] = {}
        for nbest in nbests:
            for info in nbest[id_]:
                if info['text'] in total_nbest[id_].keys():
                    total_nbest[id_][info['text']] += info['probability']
                else:
                    total_nbest[id_][info['text']] = info['probability']
    return total_nbest
```

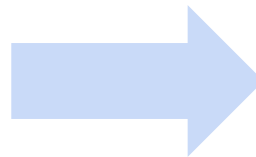
```
def get_preds(total_nbest):
    preds = {}
    for id_ in total_nbest.keys():
        best = sorted(total_nbest[id_].keys(), key=lambda x: total_nbest[id_][x])[-1]
        preds[id_] = best
    return preds
```


Ensemble

soft-voting

- 각 prediction에 대한 probability를 총 합한 후 가장 높은 probability를 갖는 prediction text 제출

```
"mrc-0-003951": {
  "뉴질랜드": 3.7963509206401795,
  "뉴질랜드에서": 0.002903978049289435,
  "바누아투": 0.25237240876140277,
  "싱가포르": 0.053681404217059026,
  "뉴질랜드에서 마오리언어는 세 개의 공식 언어 중 하나이고, 다른 언어는 영어와 뉴질랜드 수화이다.": 0.00023766442427586298,
  "뉴질랜드 북쪽에 사는 마오리족은 골에 내륙이고 있다는 느낌을 받았고, 뉴질랜드": 3.571222350728931e-05,
  "뉴질랜드에서 마오리언어": 0.00014560615613845584,
  "싱가포르에서는 영어, 중국어, 말레이어, 타밀어가 공용어로 사용되기 때문에 싱가포르": 1.7501615729997866e-05,
  "바누아투는 현재 예전 식민지 시절의 영향으로 3가지 공용어를 사용하고 있는데 바누아투": 0.00456370169672482,
  "뉴질랜드에서 마오리언어는 세 개의 공식 언어 중 하나이고, 다른 언어는 영어와 뉴질랜드": 7.52786224893498e-05,
  "예를 들어, 뉴질랜드": 7.173551011874224e-06,
  "": 2.4701073471078416e-06,
  "뉴질랜드에서 마오리언어는 세 개의 공식 언어 중 하나이고, 다른 언어": 6.965573140860215e-05,
  "뉴질랜드에서 마오리언어는 세 개의 공식 언어 중 하나이고, 다른 언어는 영어와 뉴질랜드 수화": 9.167216182959237e-05,
  "뉴질랜드에서 마오리언어는 세 개의 공식 언어 중 하나": 9.693712399894139e-06,
  "뉴질랜드에서 마오리언어는 세 개의 공식 언어": 1.6219872804867919e-06,
  "정착자들은 토착 언어들을 보존하기 위해 많은 노력을 기울이고 있다. 예를 들어, 뉴질랜드": 7.334261181313195e-05,
  "토착 언어들을 보존하기 위해 많은 노력을 기울이고 있다. 예를 들어, 뉴질랜드": 1.1982880323557765e-06,
  "유럽연합": 0.049595541262533516,
  "바누아투는": 0.001550592376588611,
  "뉴질랜드 초대 총독 윌리엄 홀슨은 1842년에 사망했다. 뉴질랜드": 1.2190180314064492e-05,
  "바": 6.150511126179481e-05,
  "케냐": 8.219446499424521e-06,
  "아일랜드": 0.3857541102170944,
  "미국": 0.2873286511749029,
  "필리핀": 0.07816744036972523,
  "멕시코": 0.03817123919725418,
  "영국": 0.027642924338579178,
  "필리핀": 0.02621247060596943,
  "유럽연합": 0.024054352194070816,
  "미국인": 0.023081675171852112,
  "유럽": 0.06519082374870777,
  "ENG": 0.02201148308813572,
  "뉴질랜드, 통가 등이 주변국이다.\n\n바누아투": 5.105580203235149e-05,
  "솔로몬 제도, 오스트레일리아, 뉴질랜드, 통가 등이 주변국이다.\n\n바누아투": 4.945573164150119e-05,
  "오스트레일리아, 뉴질랜드, 통가 등이 주변국이다.\n\n바누아투": 3.393581573618576e-05,
  "뉴질랜드에서 마오리언어는 세 개의 공식 언어 중 하나이고, 다른 언어는 영어": 3.387605829630047e-05,
  "반드시": 0.25916697084903717,
  "심지어": 0.0706534013151225,
  "쪽": 0.16849475353956223,
  "직어도": 0.11683658510446548,
  "직어도 하나 이상의 언어 변이를 공유하는 집단"으로 정의하였는데 이 두 가지 모두가 쪽": 0.11683658510446548,
  "비록": 0.10564098879694939,
  "금속화": 0.04727673903107643,
  "오직": 0.03234316408634186,
  "너무": 0.011543489061295986,
  "대략": 0.01154347788542509,
  "바누아투": 5.60479702471639e-06,
  "바누": 5.053680069977418e-06
}
```



```
"mrc-0-003951": "뉴질랜드",
```

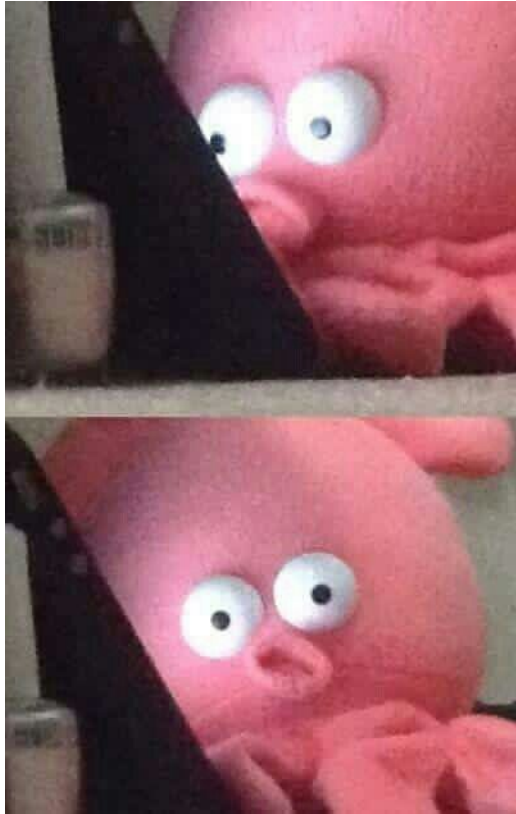
Ensemble

- 'klue/roberta-large' 4개
- 'klue/roberta-large' 4개 + 'koelectra' 1개 + 'xlm-roberta-large' 1개






















```
4
description.txt
{..} klue-roberta_gw_bs128_nbest_predictions.json
{..} klue-roberta_gw_bs256_nbest_predictions.json
{..} klue-roberta_gwq_bs128_nbest_predictions.json
{..} klue-roberta_gwq_bs256_nbest_predictions.json
{..} predictions.json
{..} total_nbest.json
```

```
5
description.txt
{..} klue-roberta_gw_bs128_nbest_predictions.json
{..} klue-roberta_gw_bs256_nbest_predictions.json
{..} klue-roberta_gwq_bs128_nbest_predictions.json
{..} klue-roberta_gwq_bs256_nbest_predictions.json
{..} koelectra_nbest.json
{..} predictions.json
{..} total_nbest.json
{..} xlm-roberta-large_nbest_predictions.json
```

Conclusion























바.. 발표...?

6	NLP_10조	     	63.7500	73.5800	52	6d
7	NLP-01조	    	63.3300	75.8500	40	3d
8	NLP_06조	    	59.1700	69.0300	31	2d
9	NLP-03조	    	57.5000	68.5400	31	19h



Public

1	NLP_09조	    	69.7200	81.7400	81	15h
2	NLP_11조	    	66.1100	78.0400	97	18h
3	NLP-01조	    	64.4400	76.9100	40	16h
4	NLP-07조	    	63.0600	74.4900	77	1d

Private

감사합니다!

