<https://littlefoxdiary.tistory.com/39>

**Recipes for building an open-domain chatbot(2020)**

**BLENDER**

페이스북 AI  
대화에 적절히 개입, 지식과 강세, 페르소나를 나타내면서 멀티턴 대화에서 일관적인 성격 유지에 초점  
최대 9.4B 파라미터 크기의 모델 버전  
호응도와 사람다움(engageness, humanness)에서 다른 모델 능가  
미나와 비교할 때 25~75%, 35~65% 개선

기존 연구: 파라미터 수 & 데이터 수 중요

모델 크기 스케일링 할 때 두 가지 추가 접근법이 있다.

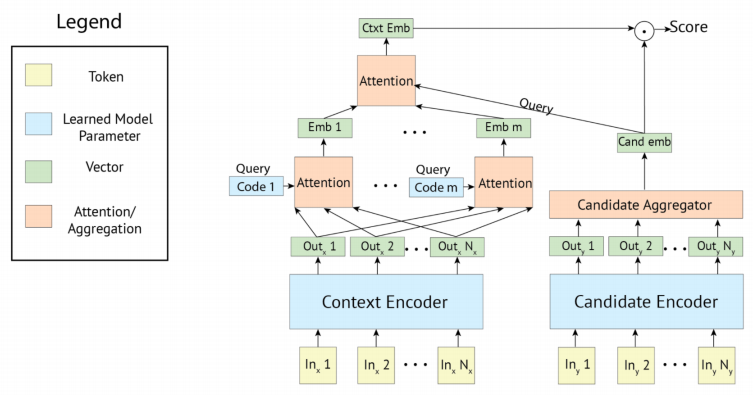
**블렌딩 스킬 – Blended Skill Task(BST) 셋업**

성격(페르소나), 개입, 지식, 강조 등을 BST로 훈련 = 초기대화 컨텍스트(페르소나&주제)를 통해 학습시키고자 하는 특성에 초점 맞추는 법  
-> 작은 모델 + BST가 큰 모델보다 성능 좋음  
-> 바람직한 측면 강조 -> 나쁜 측면의 코퍼스 억제

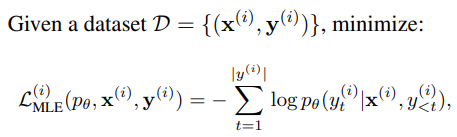
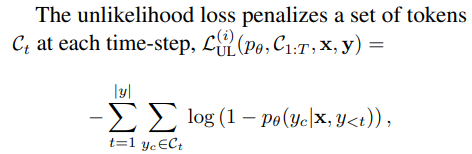
**생성 전략**

빔서치 파라미터를 잘 조절하면 더 좋은 성능

모델 아키텍쳐, not end-to-end 컴포넌트 결합식

1. Retrieval  
     
   대화 이력 인풋 -> 다음 대화 발화  
   답변 후보들이 기존에 주어지고(?? 이런 task가 있나, 따로 구축한 대형 답변 후보 데이터셋 인듯) 각 답변에 score를 하는 방식  
     
   poly-encoder 구조 = 다양한 대화 모델에서 SOTA
2. Generator: 일반적인 S2S 트랜스포머  
     
   ParlAI 버전, 사전학습 데이터에 대한 BPE, 세부 세팅은 참조
3. Retrieve and Refine  
     
   대화 retrieval과 지식 retrieval 두 가지를 고려  
   위의 생성 모델 직전에 무언가 추가 정보를 받고(KG도 활용 가능성 존재??) 답변 생성  
   일반 생성 모델은 외부 지식 접근 힘듦 = 모델에 내포돼 있는 정보 부족이기 때문  
     
   Dialogue Retrieval: (1)에서 구한 답변 + 스페셜 토큰 + 대화 이력이 (2)의 인풋  
   -> copy도 필요하다면 이용  
     
   Knowledge Retrieval: 거대한 지식 베이스에서 외부 지식 이용  
   Wizard of Wikipedia task 풀기 위해 고안된 방식 -> Wizard Generative Model  
   대화에서 지식 어떻게 사용? 위키 덤프에서 TF-IDF 기반으로 지식 찾고 대화 후보 랭킹을 매기는데 사용, 필요 없다면 안 사용하는 모델

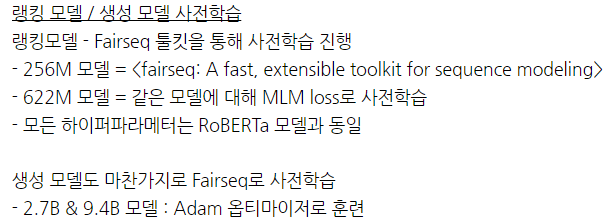
**목적 함수**

1. Retrieval  
     
   CEL 이용, (n-1)개의 표본 줄이는 negative sample, 큰 배치
2. 생성 과정  
     
   일반적인 likelihood 방식(MLE)  
     
   
3. Retrieve & Refine  
     
   알파-블렌딩 사용  
     
   받아온 답변 그냥 이어 붙여 사용 x  
   받아온 답변을 무시하는 경향이 많다(gold label과 관련성을 확인할 수 없기 때문 = not end-to-end의 한계)  
   dialogue retrieval에서 일부는 무조건 사용하도록 한다. = not end-to-end의 컴포넌트간 간극을 스무딩
4. Unlikelihood loss를 추가하여 반복형사, overrepresent 문제 줄임 = regularization = 각 스텝에서 어떤 토큰에 대한 페널티 부과  
     
     
     
   likelihood = 전체 시퀀스의 확률 분포 모델링 (GT와 비슷하게)  
   UL = 바이어스 수정(GT와 다르게, 다양하게?)  
   또한 동어 반복 문제 해결을 위해 n-gram 트래킹을 통해 negative candidate 만듦

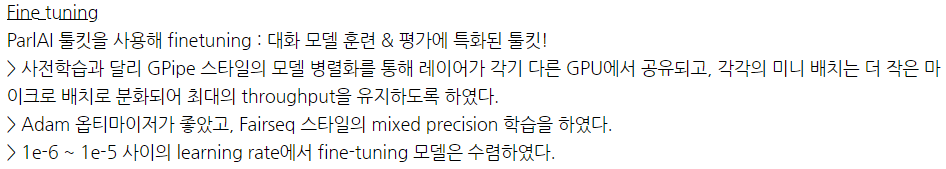
**디코딩**

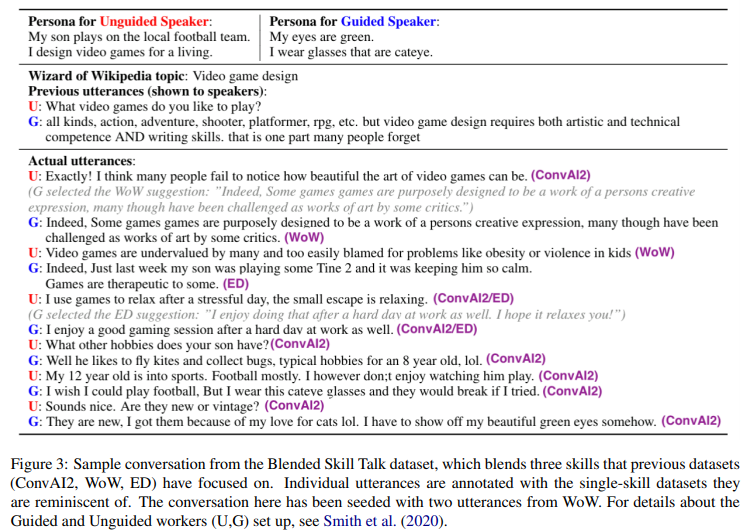
빔서치 -> 샘플링 -> response length: 짧게 생성 방지 위해 미니멈 length & predictive length 고려 -> subsequence blocking = 동어 반복 문제 해결 n-gram = 위에서 말한 것

**훈련 세팅**



데이터는 pushshift.io Reddit으로 사전학습



데이터셋을 혼합해 만듬, 페르소나가 있음(이래서 블렌딩인듯)  


즉 윗 부분이 pre-context로 주어지면 actual utterance가 실제 golds가 된다.

아주 큰 답변 후보 데이터셋도 그러면 blended dataset에서 오나??

**나쁜말 막는 분류기도 있음**

**평가 방법**

ACUTE eval: 두 개의 대화를 보고 어떤 챗봇과 대화하고 싶은지, 어떤게 더 사람처럼 느껴지는지 토너먼트 형식으로

Self-Chat ACUTE-Eval: 고르는 것은 사람이 하되, 대화는 봇끼리!

Discussion

1. 실수 여전히  
     
   반복, 모순, 지식 부족으로 인한 지엽적 발화 해결 못함
2. 14턴 이내의 짧은 대화에서 성능 좋음  
     
   긴 대화는 역시 long term 문제 -> 긴 메모리 연구 이용하면 좋을 듯  
   하지만 긴 데이터셋도 필요한데 이건 오버헤드 큼, 또한 긴 대화일수록 핵심적인 것만 뽑아내는 기능도 중요해질 것  
   BST는 흥미로운 발화 유도 강함(파인튜닝 시 초기에 대화 주제와 페르소나가 정해지니까)  
   즉, 이 논문에서 가장 독특한 것은 파인튜닝 데이터셋인데 긴 대화일수록 이것을 상세하게 해야된다.
3. 바른 행동 문제
4. PPL도 중요하지만 디코딩 알고리즘이 중요(같은 PPL이라도 다른 결과) = 재밌는 발화, 공감하는 발화?(SSA에서 놓치는 부분이 이걸로 해결될 수도), 또한 데이터 선택도 중요 => 결론적으로 사람다움과 호응도가 증가