**Weekly report**

날짜 : **2023-03-30**

연구원 : **최윤석**

* **이전 수행계획**
  + **관계형 테이블 임베딩 :**
    - Value overlap + Semantic overlap + Embedding(EmbDI) 을 사용하여 테이블 융합이 가능한지 확인하는 나만의 아키텍처 연구
    - 테이블 간 조인 / 유니온 하기위한 컬럼을 찾는 문제에 관해 생각 (Elmo / Bert / Word2Vec) 등도 생각 / 내가 손으로 하는 것을 자동화하는 방법으로 생각
    - SemProp(Attribute overlap + Value overlap + Embedding) 문제점을 찾고 문제점을 타개할 해결방안 찾기, 큰 부분은 모방해도 되지만 독창성 있는 아이디어로 나의 연구를 할 것
    - Contextual word embeddings for tabular data search and integration 저자가 올린 소스코드 설치 및 실행
    - Efficient Joinable Table Discovery in Data Lakes: A High-Dimensional Similarity-Based Approach 읽고 정리
    - DeepJoin: Joinable Table Discovery with Pre-trained Language Models 읽고 정리
    - 테이블 융합 관련 논문 추가 검색
    - 연구 논문에 사용할 만한 아이디어 구상하기

* **수행결과**
* **관계형 테이블 임베딩** : (임베딩 / 융합 가능성 파트 분류해서 작성)
  + 논문 아이디어 구상
    - Tag embedding(W2V) + column name embedding(EmbDI) + column value embedding(EmbDI)을 활용해 조인이 가능한 테이블을 찾아내는 연구

도표이(가) 표시된 사진

자동 생성된 설명

Figure . 내가 구상한 융합 가능한 테이블을 찾는 모델의 구조

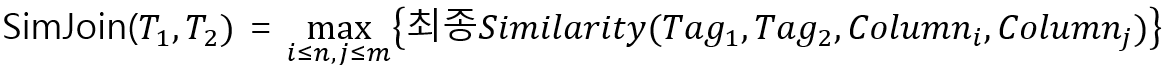
* + - 조인이 가능한지 판단하는 기준 : 두 테이블의 컬럼쌍의 최종 similarity 중 최대값이 Threshold 값보다 높은가?

Figure . 조인이 가능한지 판단하는 기준식

* + - 최종 Similarity의 계산식은 각각의 similarity 값에 가중치를 더해서 계산하는 것으로 함.
      * Ex. 최종Similarity = Tag similarity \* 0.2 + column name similarity \* 0.3 + column value similarity \* 0.5
    - Tag는 각 테이블의 메타데이터로 W2V를 사용해 embedding함. 두 테이블이 비슷한 도메인의 테이블인지를 확인하는 용으로 사용함.
    - Column name은 각 컬럼의 이름으로, EmbDI를 활용해 embedding함. 두 컬럼명을 매개로 두 테이블이 조인할 수 있는지 확인하는 용으로 사용함.
    - Column value는 각 컬럼에 포함된 값들로, EmbDI를 활용해 embedding함. Embedding이 완료되면 각 컬럼에 속한 Unique value들끼리 코사인 유사도를 계산하고 계산한 값을 전부 더하고 그 수만큼 나누어 최종 Column value끼리 유사도 값을 구함. 두 컬럼에 포함된 값을 매개로 두 테이블이 조인할 수 있는지 확인하는 용으로 사용함
  + Contextual word embeddings for tabular data search and integration 저자가 올린 소스코드 설치 및 실행
    - 소스코드에서는 Join / Union 둘 다 가능하지만 Join에 관심이 있기에 Join만 진행하였음.
    - 총 3단계로 구성되어 있음.
      * Table similarity : 서로 다른 테이블들의 각각의 컬럼쌍에 대해 similarity를 계산하고 파일로 출력함
      * Table retrieval : 앞서 출력된 similarity 파일을 활용하여 각 테이블과 가장 관련된 n개의 테이블과의 similarity를 출력함. 이 때 출력하는 similarity는 각각의 컬럼쌍에 대한 similarity 중 최대값임.
      * Table integration : threshold 값을 넣어주면, 앞서 출력된 similarity 파일을 활용해 두 테이블과 두 테이블을 조인할 컬럼쌍을 제시해줌.
    - Europe.csv, tripadvisor\_european\_restaurants.csv, android\_games\_224games.csv 3개의 테이블을 입력으로 소스코드를 실행해 봄
      * Europe.csv는 유럽 지역별 인구에 관한 테이블
      * tripadvisor\_european\_restaurants.csv는 유럽 전역에 위치한 음식점 위치 정보
      * android\_games\_224games.csv는 안드로이드에서 즐길 수 있는 224개의 게임에 대한 정보
      * Europe.csv와 tripadvisor\_european\_restaurants.csv는 국가명으로 조인이 가능하다고 직접 판단한 테이블이고, android\_games\_224games.csv는 위의 두 테이블과 조인이 안된다고 판단한 테이블임.
    - 최종 실행 결과



* + - Europe.csv와 tripadvisor\_european\_restaurants.csv를 국가명을 활용해 Join 할 수 있다고 보여줌.
    - 기존의 나의 연구에서는 두 테이블(Europe.csv와 tripadvisor\_european\_restaurants.csv)이 융합 가능하다고 판단하여 활용하였지만, 테이블 레코드 전체를 임베딩하는 방식을 사용했기에, 두 테이블의 코사인 유사도가 높게 나오지 않았음.
    - 이 논문의 저자가 사용한 각 컬럼별로 임베딩하는 방식을 사용하면 Join 가능한 테이블을 훨씬 높은 퀄리티로 찾을 수 있을 것으로 보임.
  + DeepJoin: Joinable Table Discovery with Pre-trained Language Models 읽는 중
* **수행계획**
  + **관계형 테이블 임베딩 :**
    - Value overlap + Semantic overlap + Embedding(EmbDI) 을 사용하여 테이블 융합이 가능한지 확인하는 나만의 아키텍처 연구
    - 테이블 간 조인 / 유니온 하기위한 컬럼을 찾는 문제에 관해 생각 (Elmo / Bert / Word2Vec) 등도 생각 / 내가 손으로 하는 것을 자동화하는 방법으로 생각
    - SemProp(Attribute overlap + Value overlap + Embedding) 문제점을 찾고 문제점을 타개할 해결방안 찾기, 큰 부분은 모방해도 되지만 독창성 있는 아이디어로 나의 연구를 할 것
    - Efficient Joinable Table Discovery in Data Lakes: A High-Dimensional Similarity-Based Approach 읽고 정리
    - DeepJoin: Joinable Table Discovery with Pre-trained Language Models 읽고 정리
    - 테이블 융합 관련 논문 추가 검색
* **기타사항**
  + **리포트 작성 시, 논문의 내용정리 같이 남이 한 부분에 대한 설명은 자세하게 하지 말고 그 중에서 내가 얻은 아이디어나 내 연구에 관한 부분을 자세하게 설명할 것**
  + **데이터 융합 프로세스**



* **융합 가능도 척도**
  + 임베딩 거리 기준 값을 정하고 기준값보다 임베딩 거리가 가까우면 융합 가능한 데이터셋이라 판단함
  + 융합 가능한 데이터셋을 사람이 하나씩 확인하여 유의미한 융합이 되는 데이터셋을 찾음
  + 을 융합 가능도 척도라 정의함



혼합형 데이터셋 예시

등급의 semantic

융합 가능도 딥러닝 아키텍처 : 임베딩 벡터 + 메타데이터(융합 가치를 정량화할 수 있는 무언가)   
조인할 수 있는 컬럼이 필요함(A : 주민번호, B : 주민등록 번호로 조인이 가능한 테이블이면?) 융합을 하는 건 가치가 있기 때문, 가치가 있다? -> 임베딩이 가깝게 되어있다.

조인할 수 있다, 없다 -> 사람이 판단

이걸 조인해서 가치있는 정보가 나오나? -> 모델이 판단

Id를 가지는 컬럼이 의미는 같지만 겉모양이 다를때

그래프 임베딩

정형테이블을 그래프로 바꾸는 논문읽기

그래프 정보를 임베딩한 정보와 기존 윤종찬 연구원과의 차이점 발견

나만의 새로운 패러다임 찾기

데이터셋은 영문 데이터만 사용

융합 가능성 평가는 임베딩 성능을 확보한 후에

테이블 융합의 연산은 Union / Join이 있음 -> Union / Join 을 구분해줄 수 있는 모델이 있으면 좋겠음.

Ex. Output 값이 1이면 Union 0이면 Join

조인키만 있어도 조인이 되는게 아닐 수 있음. 나중에 코드를 짤 때 조인(유니온)이 가능한 예제를 sql로 조인해보고 조인이 가능한 예제를 모델에 넣어서 조인이 가능하다고 판독이 되는 것을 확인해보기

6월 학술대회 준비 (완성된 논문이 아니더라도 결과가 의미가 있는 것이라면 논문 작성 가능)

유니온 / 조인 중 하나만 따로라도 혹은 기존의 실험 내용 중 소논문으로 만들거리를 찾아보기 (정보과학회 투고 예정)