파이널 프로젝트(악성댓글 필터링 봇)

- 주제 선정 배경
    - 최근 유명인 극단적 선택 등 악성댓글, 사회문제 대두
- 악플 기준 정의
    - 악플 처벌 판례 기준 ex. 특정인 비방, 공연성 등
    - 우리가 만든 욕설/비속어 사전
        - 욕설: 일반적인 욕설
        - 저속한 표현: 타인에게 불쾌감을 주는 속되고, 격이 낮은 표현
        - 선정적인 표현: 성적으로 자극적인 표현
        - 폭력적인 표현: 신체적 위협에 대한 표현
        - 차별적인 표현: 지역/인종/국가/종교 등에 기반한 차별 표현
        - 비하적인 표현: 상대방에게 모멸감과 수치심을 주는 비하 표현
    - 정태륭 <한국의 욕설백과>
- 데이터셋 구축
    - 댓글 데이터 크롤링
    - 탐색적 데이터 분석(EDA) 및 전처리
        - 댓글 데이터 개수 -> unique로 중복 제거
        - 댓글 데이터 길이(각 댓글의 길이 분포) -> X00자 이상은 제거
        - (많이 사용되는 단어 wordcloud)
        - 악성댓글 데이터 분포 -> 앞서 정의한 유형이 포함되면 2로 일괄 라벨링한 결과, 악성댓글 X천 개임을 확인
        - 특수문자 및 영어 비율 -> 특수문자 제거, 영어로만 이뤄진 댓글 제거
    - 라벨링
        - 앞서 정의된 유형의 표현을 포함한 댓글 -> 2로 라벨링
        - 판단하기 애매한 댓글 -> 1로 라벨링
        - 앞서 정의된 유형의 표현을 포함하지 않은 댓글 -> 0으로 라벨링
    - 데이터셋 구축
        - 라벨링 결과 -> 총 X만 X천 개의 댓글 중 0,1,2의 비율
        - 데이터셋에서 1점을 부여받은 댓글은 실제로 악플 여부를 판단하기 어려움 -> 학습을 위한 데이터셋에서 제외
        - 2점을 부여받은 댓글을 악플, 0점을 부여받은 댓글을 악플이 아닌 것으로 간주 -> 약 X만여 건의 라벨링된 댓글 데이터셋 구축
- 모델링
    - 입출력 정의
        - 입력: 댓글 내용
        - 출력: 0~1 사이의 실수(real number) -> 댓글 내용이 악플일 확률 cf. threshold

- 텍스트 전처리
  - 댓글에는 오타나 비표준적인 표현, 한글 자모음을 활용한 표현이 자주 나타나므로 단어나 형태소 단위는 적절하지 않다고 판단 -> 음절 단위로 댓글 토큰화
  - 만들어진 토큰을 모델에 적용하기 위해 고유 인덱스 번호를 부여 -> 토큰 라벨링
  - 토큰화와 라벨링이 이루어진 토큰을 의미없는 숫자(0)을 이용해 데이터의 길이를 동일하게 만듦 -> 토큰 패딩
- 모델 구조 설계
  - 딥러닝 모델간 비교
    - LSTM
      - (뉴럴 넷 구조) + 성능
    - CNN
      - (뉴럴 넷 구조) + 성능
    - 베타 테스트 및 오버 샘플링
      - 데이터가 불균형할 경우 학습 시 편향 발생할 가능성 존재. 구축한 학습 데이터셋의 악플과 악플이 아닌 댓글 간의 비율이 ?:? 정도였기 때문에, 악플에 해당하는 데이터를 ?배 정도 복제(혹은 추가)해 데이터의 균형을 맞춰줌
      - "쓰레기는 쓰레기통에" 등 성능 향상 예시
  - 딥러닝 vs 머신러닝 모델 비교
    - (모델 구조) + 성능 -> 딥러닝에 비해 모델 성능 떨어짐
  - 최종 모델 선정 + 최종 성능
- 웹 시연
  - 시연 동영상
    - 사진 + 글 업로드
    - 댓글 블라인드 기능 OFF -> ON
    - 실제 연예인 게시물에 달린 악성댓글 작성 -> 블라인드 처리
    - (신고하기/이의제기)
  - 댓글을 달아주세요!
    - http://commentfilter.pythonanywhere.com/
    - Lesson Learned(웹 About 페이지)
      - 댓글 데이터의 객관성 문제 -> 신고/이의제기 기능 구현, 크라우드 소싱 + 평균으로 데이터 라벨링 필요
      - 성능 문제 -> 주기적인 베타 테스트/신고,이의제기 기능을 통해 악성댓글 학습 데이터셋 업데이트 필요
- Q&A
  - 웹 Q&A 포스트에 질문이나 의견 댓글로 달아주세요!

+