캡스톤 18조 장우혁 멘토링 활용 보고서

프로젝트명	O24Sec	팀명	멜러리를 찾아서

팀 멘토링

* 간단히 팀의 공통사항에 대한 미팅 주제와 개인별로 느낀점을 적었습니다.

◉ [03/16] 팀 미팅 - 윤명근 교수님
- 암호화 판별과 내부 외부 분류의 threshold 선정에 대한 의문점
=> 느낀점: 임계치를 정한 기준을 설명할 수 있어야 합니다.

◉ **[03/19] Ceeya 미팅 - 문성익 멘토님**
(Q) 기사에 대해 자연어 처리를 해서 비슷한 부류끼리 유사도 기반을 통한 클러스터링을 진행하셨었는데 해당 프로젝트 진행시 어떤 과정을 통해서 진행하셨나요?

(멘토) 먼저 기사에 나오는 언어를 사전 기반으로 처리를 하였습니다. 자연어 처리 라이브러리를 통해 나온 단어들을 체크하고 tf-idf 유사도 분류 방법으로 진행을 했습니다. 이렇게 안하더라도 n-gram 방식으로 언어를 처리해서 하는 방법도 있었네요. 그 당시 n-gram에서 n의 크기를 계속 바꿔가면서 적절한 결과를 얻어낼때까지 주먹구구 방식으로 진행을 했었습니다.
현재 진행하시는 프로젝트에서 적용을 해볼때 바이트 단위를 n-gram으로 길이별로 쪼개서 나오는 바이트 열을 벡터로 만들어 해당 벡터에 대해 tf-idf로 검증해보는것도 해볼만한 시도일 것 같습니다.

(Q) 그렇다면 tf-idf 검증을 하실때 기준을 잡아야 하셨을텐데 어떤걸 기준으로 잡으셨나요? 그때 기준을 뽑는 방법은 어떤게 있으셨나요?
(멘토) 사실 이때 기준을 뽑을때는 확연한 기준이 있었습니다. 사람이 한 기사를 봐서 딱봐도 해당 분류에 대한 기사라는 것을 알 수 도 있었죠. 그리고 이미 약간 분류 되어있는것에 잘못된 분류가 없는지 실험을 할 때도 있어서 이때 잘 분류되어 있는 기사를 통해 분류를 진행하기도 하였습니다.

(멘토) 그리고 만약 단어의 순서에도 중요한 연관이 있을 수 있으니 그럴때는 n-gram 방식에서 n의 크기를 늘려 연속되는 문자열을 벡터에 넣기도 하였습니다. 이럴 경우에 전체적인 문장이 벡터화 되기 때문에 그 문장에 대해 검증을 할 수 있으니까요 만약 프로젝트 진행하시면서 순서나 긴 길이에 대한 시그니처가 있다면 n의 크기를 키워서 자르는 방식이 있을 수 도 있을것 같습니다. 근데 일단 n의 크기를 작게 만들어서 확인해보면서 n의 크기를 키우면 계속 비교해가면서 진행할 수 있으니 더 좋겠네요
(Q) 현재 프로젝트 진행중에 너무 많은 데이터로 인한 클러스터링 과정에서 소요되는 시간 문제와 데이터에 대한 피처를 조금만 크게 잡아도 나타나는 메모리 문제가 있습니다.

혹시 이럴때 어떻게 진행을 하면 효율적으로 진행 할 수 있을까요?

다음 장에 계속

팀 멘토링	- (멘토) 일단 너무 많은 데이터로 인한 시간소요 문제에 대해 말해보자면 우리가 프로젝트를 진행하면서 처음부터 프로젝트 마지막에 쓸 결과를 보는것이 아니라 단계 단계 진행하면서 나오는 결과를 검증하는 것이 우선일 것 입니다. 따라서 전체 데이터에 대해서 실험을 하는 것이 아닌 샘플링을 통해 데이터 개수를 줄여놓고 먼저 만든 클러스터링 알고리즘이나 자연어 처리에 대한 효율성을 점검해 가면서 나갈 필요가 있습니다. 즉, 각각 해나아갈때 중간과정에서 결과를 볼 수 있을 만큼의 데이터만 샘플링 해서 시간을 줄여서 실험을 하는 것이 효과적일 수 있습니다. 메모리 문제에 대해서는 사실 로컬에서 실험을 하다보면 어쩔 수 없는 문제이기도 합니다. AWS에 sagemaker에서는 데이터를 맞춰 넣어주면 거기에서 클러스터링을 해주기 때문에 가능하다면 AWS를 써보는 것도 방법 중 하나일 것 같습니다. 아니면 만약에 Tensorflow를 사용가능하다면 Tensorflow를 통해 API식 접근으로 분산처리를 통해 좀 완화 될 수 도 있는데 이 방법은 조금 과한 것 같군요 일단 먼저 말했던 샘플링 처럼 데이터 개수를 줄여서 메모리 할당량을 낮춰보는 방법이 좋을 것 같습니다. (Q) 기사를 유사도 기반으로 분류 하셨을때 기준 기사와 분류 대상 기사의 유사도 임계치를 정하셨을텐데 이때 어떤 방법으로 임계치를 얻으 실 수 있으셨나요? (멘토) 임계치는 데이터를 하나씩 넣어보면서 제대로 분류가 되나 안되나 확인해가면서 기준을 잡았습니다. 결국 사람이 직접 판단을 한 것이죠 => 느낀점: 첫 번째 멘토링때, 멘토님의 전문 분야와 멘토님이 잘 알려주실 수 있는 부분을 서로 이야기 나눈 덕에 준비했던 질문들에서 좋은 답변을 받을 수 있었습니다. 또한 명확한 답이 없는 문제에 대해선 계속해서 토의를 나누며 최선의 선택지를 강구해보고, 중요도를 기준으로 다시 생각할 수 있는 기회가 됐습니다.
개인 멘토링	* 제 담당 분야인 1차 목표사항 달성을 위해 나누었던 팀, 개인 미팅의 멘토링 내용입니다. ◉ **[03/19] Ceeya 미팅 - 문성익 멘토님** (Q) 현재 데이터를 통계적으로 분석하여 순수한 내부만을 찾을 수 있는 특정 임계값으로 설정했는데, 내부/외부 임계값을 어떻게 잡을까요??? (멘토) 현재 클러스터링 실험을 하고 있으므로, 일단 순수한 내부 데이터만을 뽑을 수 있는 임계값으로 잘 설정했다고 생각합니다. 클러스터링 성능을 보면서 임계값을 낮추는 식으로 진행하면 좋을 것 같습니다. (Q) 분류 대상이 완벽하게 나눌 수 없는 데이터이다 보니 만든 알고리즘으로 100% 분별에 대한 확신성이 없습니다. 그래서 현재 임계값에 따라 FN 와 FP 사이의 Trade-off 관계에서 딜레마에 빠졌는데 이 경우에 어떤식으로 진행하는 것이 좋을까요?? (멘토) 음.. 사실 이 문제는 우리가 항상 고민하는 부분입니다. 그래도 현재 프로젝트에서는 이 과정이 결과나타나는 것이 아닌 결과를 만들기 위한 중간 처리 과정이기 때문에 마지막 처리 과정에서 좀 더 깨끗히 처리 할 수 있도록 클린한 데이터가 많이 남는 쪽으로 방향을 잡으면 좋을 것 같습니다. 예를 들어 내부 외부 둘다 조금씩 섞이게 될 바에는 내부는 완전히 깔끔하게 남기고 외부에는 조금 섞이더라도 한쪽이라도 클린하게 만들어 좋은 결과를 얻는것이 좋을 것 같습니다. => 느낀점: 현재 고민하고 있는 부분인 내부 외부 임계값에 대한 추가적인 아이디어는 얻지 못했지만, 한계나 장애물에 봉착했을 경우 내릴 수 있는 판단 기준과 최선의 선택을 내려야 한다는 사실을 깨닫게 해주셨습니다.

개인 멘토링	◉ [03/16] 팀 미팅 - 윤명근 교수님 - IP B-class대역을 사용하여 내부를 분류하는 기준의 설명이 부족해, 모든 데이터의 B-class 통신 그래프를 pyplot으로 표현(node: B-class IP, edge: 통신) => 느낀점: 데이터 시각화의 중요성을 깨달을 수 있었습니다. 또한, 현재 데이터 분류 기준에 문제가 있다는 것을 인지하지 못한 것을 다시 생각해볼 수 있었습니다. - 암호화 판별과 내부 외부 분류의 threshold 선정에 대한 의문점 => 느낀점: 임계치를 정한 기준을 설명할 수 있어야 합니다. ◉ [03/18] 개인 미팅 - 윤명근 교수님 - 3/16 팀 미팅에서 피드백 받은 내부 분류기준 임계치에 대한 실험 결과를 교수님과 공유하여 피드백을 받았습니다. => 느낀점: 접근 방식의 잘못을 이해하게 되었고 새 분류 기준을 강구해야 할 것 같습니다. ◉ [03/23] 팀 미팅 - 윤명근 교수님 - 내부 외부 분류 임계치 설정1: 모든 기간(1월 ~ 12월 : 1개월 ~ 12개월(연속)의 경우의 수 85가지의 실험)을 통해 확실한 임계치 설정 => 느낀점: 임계치 선정 방식을 가지고 있는 데이터 셋에서 통계적으로 선정하였는데, 성능과 상관 없이 그 기준이 합리적이고 명확하고 절대적이며 모든 데이터 셋에서 유효한 상대성을 가져야 함을 알 수 있었습니다. 임계치를 대변할 수 있는 선정 과정을 다시 생각해야 할 것 같습니다. ◉ [03/26] 팀 미팅 - 윤명근 교수님 - 내부 외부 분류 임계치 설정2: 각 기관에서 네트워크 개념으로 접근하여 합리적이고 타당한 공식을 도출함 => 느낀점: 가끔 문제라는 것을 인식하지 못하는 경우가 있는데, 한 문제에 대해 여러번 피드백을 받으면서 뿌듯한 결과를 낼 수 있어 기뻤습니다.