Kingomanager: 추천시스템을 활용한 대학생 맞춤형 정보 제공 어플리케이션 개발

강신규[1], 김준우[2], 박충현[3], 구형준[4]
성균관대학교 (시스템경영학과/소프트웨어학과[1], 수학과/컴퓨터공학과[2],
기계공학과/소프트웨어학과[3]) 학부생, 성균관대학교 교수[4]
rkdtlseb@naver.com[1], junoo1997@naver.com[2],
choonghyun.park10@gmail.com[3], kevin.koo@skku.edu[4]

Kingomanager: A Personalized Information-providing Application with a Recommendation System for University Students [1)]

Shingyu Kang[1], JunWoo Kim[2], ChoongHyeon Park[1], Hyungjoon Koo[1]
[1]Department of Computer Science and Engineering, Sungkyunkwan University
[2]Department of Computer Science, Sungkyunkwan University

요 약

대학 생활을 하면서 자신이 필요한 정보를 모두 챙기기는 쉽지 않다. 매번 학교 홈페이지나 관련 사이트에 접속하여 확인하는 것은 번거롭기도 하고 신입생의 경우에는 그런 정보의 존재조차 잘 모르는 경우가 많다. 때문에 이 논문에서는 웹 크롤링 방식을 통해 다양한 사이트에서 필요한 정보를 수집하고, 기계학습 모델 중 N-GCN을 기반으로 한 추천시스템을 이용하여 본인에게 맞는 추천과목, 동아리 모집공고, 학술대회, 채용공고 등의 정보를 제공해주는 Kingomanager를 소개한다. Kingomanager는 학생들의 학년, 관심분야를 고려해서 개개인별 맞춤 정보를 추천해준다. 추천 받은 정보들은 메신저 형태의 어플리케이션을 통해서 확인할 수 있고, 해당 정보들은 언제든지 다시 검색하여 다시 찾아볼 수 있다. 어플리케이션 구현에서 Front-end는 React-Native를 사용하였고, Back-end는 Flask와 AWS 서비스를 사용하였다. 본 논문에서는 성균관대학교 소프트웨어학과 학생을 대상으로 하는 프로토타입 어플리케이션을 개발했다.

1. 서론

대학생활은 오로지 학업에만 열중하였던 고등학교와 달리 자신의 진로활동에 필요한 정보를 직접 찾아봐야 한다. 하지만 학기 중에는 과제나 프로젝트 등의 학업과 병행하여 시기에 적절한 진로활동들을 찾아보기가 쉽지 않다. 또한 진로활동 정보는 여러 홈페이지에 널리 퍼져 있기 때문에, 본인에게 맞는 활동이 무엇인지 찾는데 많은 시간을 소비하게 되며, 적극적으로 홍보를 하지 않는 홈페이지의 공지사항들은 확인하기가 어렵다. 여러 가지 동아리, 대외활동에 대한 정보를 제공해주는 어플은 존재하지만 1) 학교 외부의 활동에 대한 정보만 제공하거나, 2) 개인 맞춤형 정보를 제공하지 않는 등의 한계점이 있다. 때문에 학생들이 이러한 어플들이 크게 도움을 주지 못한다고 생각하여 잘 사용하지 않게 된다. 때문에 본 논문에서는 성균관대학교 소프트웨어학과 학생들에게 개개인별 맞춤형 정보를 추천해주는 kingomanger를 제안한다. kingomanager는 개개인별 맞춤 정보를 제공하고, 학교 내, 외부에 국한되지 않고 정보를 제공한다.

2. Kingomanager 디자인

kingomanager는 사용자 정보를 입력받으면, 먼저 클릭한 횟수에 따라 기존 데이터베이스에 존재하는 해당 feature의 갯수를 늘리는 형식으로 가중치를 부여한다. 그 후, 모델을 학습시키고, 추천받아야 하는 사용자의 feature를 입력받아 입력에 알맞은 컨텐츠 5개를 추천 데이터베이스에 전송해준다. Learning Rate = 0.01, epoch = 200, Training data = 750개, Validation data = 250 개를 적용하여 모델을 학습시켰는데, overfitting이나 underfitting을 방지하기 위해 Early stopping을 사용했다. input이 입력되면 추천 카테고리 top3를 출력해준다. Evaluation Metric으로는 Accuracy와 F1 score를 사용했다. Accuracy = 0.875, F1 score = 0.868정도로 나왔다. Accuracy는 우리의 목적인 예측값과 결과값이 같아야하기 때문에, 그리고 F1 score은 아무리 Accuracy 값이 높게 나와도 F1 score가 낮게 나오면 Accuracy 결과값은 의미없기 때문에 사용하였다.

2.1 그래프 합성곱 신경망 (N-GCN) 모델

Graph는 vertices(node)와 edge로 구성된 구조를 말한다.

$$G = (V, E)$$

과 같은 식으로 나타낼 수 있는데, 각 node는 feature vector를 가지고 있다. 앞에서 말한 것처럼, 기존의 모델과 다르게 그래프를 사용해서 얻을 수 있는 차별점으로는 노드들 사이의 패턴을 찾는 데 집중하는 것이 아니라 노드들 사이의 관계에 집중한다는 것이다. Adjacency matrix(A)와 각 feature vector로 구성된 feature matrix(H)를 통해 GNN layer를 통과할 때마다 각 matrix(H)의 정보를 업데이트 해준다.

$$H(l+1) = \sigma(AH(l)W(l) + b(l))$$

1) 이 논문은 2023년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을받아 수행된 연구임 (No. 2022-0-01199, 융합보안대학원(성균관대학교))

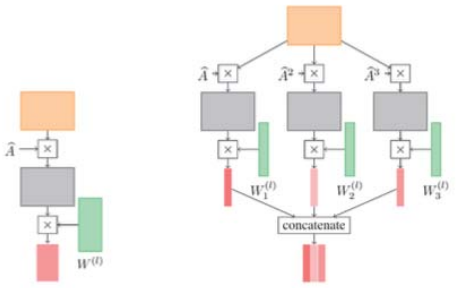

그림 1. **전통적인 GCN layer와 mixhop layer**

추가로, Adjacency matrix에서 각 노드에 연결된 가중치는 서로 다를 것이므로, Node embedding을 통해 구한 값인

$$z_u^T z_v$$

를 통해 각 노드들마다 Pagerank를 사용해 각 노드 사이에 어느 정도 연관성이 있는지를 가중치를 통해 나타낸다. 이 Pagerank를 사용해서 GNN layer를 지날 때마다 가중치를 최신화시켜 각 노드의 가중치가 수렴하게 된다. GCN을 사용했기 때문에, 여기에 추가적으로 Convolution을 적용해 H를 구해 나간다. 최종적으로, N-GCN은 그림1.과 같이 위의 GCN의 과정을 사용하는데, GCN 결과를 층(layer)을 나눠서 결과를 계산하고 그 결과들을 Concatenate해서 최종 결과를 도출한다.

2.2 프런트엔드 디자인

Kingo Manager 어플리케이션의 프론트엔드는 Auth navigation과 Main navigation으로 이루어져 있다. Auth navigation은 Login, Signup pages가 포함되어 있고, Main navigation은 Main, Settings page로 이루어져 있다. useCotext를 이용하여 만든 전역변수 user의 email값이 null이면 Auth navigation을 띄우고, 값이 존재한다면 Main navigation을 띄워준다.

2.3 백엔드 디자인

Kingo Manager의 전체적인 동작은 사용자가 자신의 관심사에 맞는 추천정보를 얻는 것과, 사용자의 관심사를 추천 알고리즘에 반영하여 모델을 학습시키는 과정을 통해서 이루어진다. 즉, AI Model과 사용자 UI가 양방향으로 필요한 정보를 통신할 수 있도록 백엔드 아키텍쳐를 설계하였고, 추천정보는 Local PC로부터 데이터베이스에 post 요청을 통해서 저장된 후, 다시 한번 서버로부터 어플리케이션에 전송되는 방식으로 구현되었다. 또한 학습 파라미터를 전송하는 구조는 반대 방향으로 이루어지는데, 어플리케이션에서 사용자의 컨텐츠 열람 횟수를 누적하여 카운트한 후에 100회 이상 횟수가 누적되면, 어플리케이션에서 서버로의 post 요청이 발생한다. 사용자 관심도 정보는 데이터베이스에 저장되며, Local PC가 필요한 때에 서버로부터 PC로 전송된다.

3. Kingomanager 구현

프론트엔드에서는 React-native를 사용하여 어플리케이션을 개발하였다. 구조가 복잡한 편이 아니고, 어플 개발에 사용되는 인력이 적은 만큼 ios와 android를 동시에 개발할 수 있는 크로스 플랫폼 앱이 적절하다고 판단하여 react-native를 선택하게 되었다. 백엔드와의 통신을 위해서 Promise기반 HTTP클라이언트인 axios를 사용하였다. 백엔드 서버는 전반적으로 AWS(Amazon Web Services)의 cloud service를 사용하였고, 소스 코드는 python 기반의 Flask API와 SQL Alchemy 라이브러리를 적극적으로 활용하였다.

4. 한계점 및 향후계획

먼저, train dataset을 생성하는 과정에서 GNN 모델의 각 노드가 가지는 feature의 갯수가 다양하지 않아, 각 user의 상태 정보를 자세하게 표현하지는 못했다. 또한 feature에서 사용자의 관심도를 확인하는 인자로서, 추천해줄 컨텐츠가 속해있는 카테고리를 사용하였기 때문에, 같은 카테고리 내에 속한 정보라면 사용자가 관심있게 여긴다는 가정을 내포하고 있다. 또한 카테고리 모델을 통해서 카테고리 범위까지의 추천을 완료한 후에는 해당 카테고리 내에서 랜덤한 컨텐츠를 추천해준다는 한계점도 존재한다. 위 한계점들을 보완하기 위해서 앞으로 충분한 사용자 데이터가 축적된다면, feature를 더욱 다양화하고, GNN 모델의 노드를 콘텐츠 단위까지 확장시키는 방법으로 보다 정교한 추천을 할 수 있을 것이다. 또한, 서버의 AI Model로부터 얻은 추천정보를 정시마다 사용자에게 전달하는 매커니즘은 설계되어 있지만, 사람의 개입없이 완전히 자동화된 방법으로 모델을 학습 및 작동시키려면 AI Model의 알고리즘을 서버상에 업로드하여야 한다. 초기에 이벤트가 발생할 때 동작을 수행하는 aws Lambda 서비스를 사용하여 딥러닝 모델을 서버상에 업로드하려고 했지만, 정시마다 추천정보를 업로드하기 위해서는 또 다시 제 3자의 post 요청이 필요하였다. 따라서 백엔드에서의 또 하나의 Future work로서, aws EC2 서비스를 사용하여 서버상에 존재하는 컴퓨터에 추천 모델을 업로드하고, 이를 통해 완전히 자동화된 시스템을 구축할 수 있을 것이다.

5. 결론

장기적으로, Kingo Manger는 성균관대학교의 대학생을 위한 진로활동 추천 서비스로서 사용자가 점점 증가할수록 사용자 정보를 학습하는 과정을 통해 보다 정밀한 추천을 할 수 있다. 또한 서비스 지원범위를 점점 확대해서 보다 일반적인 대학생들이 사용할 수 있는 진로활동 추천 서비스로 자리잡을 것이다.

참고문헌

[1] Wu, S., Tang, Y., Zhu, Y., Wang, L., Xie, X., & Tan, T.; Session-Based Recommendation with Graph Neural Networks.; AAAI; 2019; 9p

[2] Sami, A., Nazanin, A.,Hrayr, H., Amol, K., Bryan P.; A Higher-Order Graph Con- volutional Layer.; NIPS; University of Soutern California; 2018; 9p;

[3] N-GCN Homepage, https://github.com/benedekrozemberczki/MixHop-and-N-GCN.