

Improving Accuracy of Chapter-level Lecture Video Recommendation System using Keyword Cluster-based Graph Neural Networks

Purevsuren Chimeddorj*, Doohyun Kim*

*Student, Dept. of Computer Science and Engineering, Konkuk University, Seoul, Korea

*Professor, Dept. of Computer Science and Engineering, Konkuk University, Seoul, Korea

[Abstract]

In this paper, we propose a system for recommending lecture videos at the chapter level, addressing the balance between accuracy and processing speed in chapter-level video recommendations. Specifically, it has been observed that enhancing recommendation accuracy reduces processing speed, while increasing processing speed decreases accuracy. To mitigate this trade-off, a hybrid approach is proposed, utilizing techniques such as TF-IDF, k-means++ clustering, and Graph Neural Networks (GNN). The approach involves pre-constructing clusters based on chapter similarity to reduce computational load during recommendations, thereby improving processing speed, and applying GNN to the graph of clusters as nodes to enhance recommendation accuracy. Experimental results indicate that the use of GNN resulted in an approximate 19.7% increase in recommendation accuracy, as measured by the Mean Reciprocal Rank (MRR) metric, and an approximate 27.7% increase in precision defined by similarities. These findings are expected to contribute to the development of a learning system that recommends more suitable video chapters in response to learners' queries.

▶ **Key words:** Graph Neural Network, K-Mean++ Clustering, Video Lecture Chapter Recommendation

[요 약]

본 논문은 챕터 수준의 강의 동영상 추천 시스템에 있어서 추천의 정확도와 처리속도 간의 균형 문제, 즉, 추천 정확도를 향상시키려면 처리 속도가 저하되고, 반대로 처리 속도를 높일 경우 정확도가 감소하는 문제에 대하여 연구한다. 본 논문에서는 이의 해결을 위하여 TF-IDF, K-Means++ Clustering, Graph Neural Network(GNN) 등 다양한 기법을 복합적으로 활용하는 방법을 제안한다. 즉, 챕터들의 유사성을 바탕으로 클러스터를 사전에 구성함으로써 검색 시의 계산량을 줄여 속도를 향상시키면서도, 클러스터를 노드로 하는 그래프에 대하여 GNN을 적용함으로써 추천의 정확도를 향상시키는 방법을 제안한다. 실험 결과 GNN을 사용한 경우 추천의 정확도가 MRR 지표에서 약 19.7% 증가하였으며, 유사도 기반의 정밀도에 있어서 약 27.7% 증가하는 결과를 확인할 수 있었다. 이를 통해 학습자의 질의에 보다 적합한 동영상 챕터를 추천하는 학습시스템 구축에 기여할 것으로 기대한다.

▶ **주제어:** 그래프 신경망, K-Mean++ 클러스터링, 강의 동영상 챕터 추천

- First Author: Purevsuren Chimeddorj, Corresponding Author: Doohyun Kim
- *Purevsuren Chimeddorj (chimeddorj@naver.com), Dept. of Computer Science and Engineering, Konkuk University
- *Doohyun Kim (doohyun@konkuk.ac.kr), Dept. of Computer Science and Engineering, Konkuk University
- Received: 2024. 03. 18, Revised: 2024. 07. 10, Accepted: 2024. 07. 22.

I. Introduction

최근들어 그래프 신경망(Graph Neural Networks, GNN)이 소셜 네트워크, 분자 화학, 생물정보학, 교통시스템 등에서 다양한 분석과 예측에 널리 활용되고 있다. 특히 GNN은 데이터 내에 복잡한 상호 관계를 그래프를 통해 효과적으로 모델링 할 수 있다는 점[1]과 데이터 내에 내재된 상호 관계의 표현 학습 능력이 우수하다는 점[2], 그리고 그래프를 기반으로 하는 확장성 및 일반화 능력이 우수하다는 점[3] 등에서 상품, 구매자, 지역, 계절 등 다양한 측면이 고려되어야 하는 추천 시스템에서도 그 활용도가 높아지고 있다. 본 논문에서는 GNN을 사용하여 챗터 단위 간의 동영상 추천의 정확도를 향상시키기 위한 방법과 그 효과에 대하여 연구한다.

한편, 강의 동영상의 경우 일반적인 동영상과는 달리 동영상의 내용이 세부 주제인 챗터 단위로 구성되어 있는 경우가 대부분이다. 이러한 이유로 동영상 강의 검색이나 추천에 있어서도 챗터를 단위로 제공하는 것이 보다 효과적일 것이다. 이러한 배경 하에 본 연구에서는 추천 단위가 동영상 전체[5]가 아닌 챗터를 단위로 하는 강의동영상 추천 시스템을 설계하고자 한다.

한편, 챗터 단위의 추천 시스템에 있어서 강의 동영상의 증가할 경우 챗터의 개수 또한 크게 증가하게 되며 이에 따라 추천에 소요되는 처리 시간이 증가하는 문제가 발생하게 된다. 만약 이에 대응하고 간소화된 추천 방식을 도입하여 속도를 높일 경우 추천의 정확도가 감소하는 역효과가 발생하는 이른바 추천 속도와 정확도 사이의 trade-off 이슈가 발생하게 된다.

본 논문에서는 이상의 배경 하에 챗터 단위의 강의 동영상 추천 시스템에서 처리량을 감소시켜 속도의 저하를 방지하면서도 주어진 질의에 대한 부합도가 높은 챗터를 추천하는 방식으로써 클러스터링 기법과 GNN을 조합하는 방식을 제안하며 실험을 통해 이와 같은 조합이 얻는 이득을 제시하고자 한다.

본 논문의 제2장에서는 관련 연구 및 이전 연구와의 차이점에 대하여 요약차원에서 설명한다. 제3장에서는 본 논문의 접근 방법에 대하여 전체적인 구성과 아울러 세부 단계들을 설명한다. 제4장에서는 본 논문이 제안하는 GNN의 활용 방법을 보다 구체적으로 설명한다. 제5장에서는 실험 시스템의 구성과 다각적인 측면의 실험 결과를 제시하고 분석한다. 제6장에서는 이상의 연구를 바탕으로 구현된 챗터 기반의 동영상 추천 시스템을 소개한다. 마지막으로 제7장에서는 본 연구의 의의와 향후 과제에 대하여 설명한다.

II. Preliminaries

1. Related works

FM(Factorization Machine)[6]은 고객과 아이템 특성과 간의 상호 관계를 포착하는 전통적인 추천 방식으로 널리 알려져 있다. DeepFM[7]과 AutoInt[8]는 심층 신경망을 활용해 FM으로 하여금 보다 더 복잡한 상호 작용을 모델링할 수 있도록 한다. 한편, 최근들어 GNN이 활용되면서 심층 신경망에 의한 FM의 개선보다 더 나은 성과를 보이고 있으며, 추천 시스템에서도 탁월한 성과를 보이고 있다. 일례로 Wu 등[9]은 세션 데이터를 기반으로 아이템 그래프를 만들었고, Fan 등[10]은 고객과 항목의 정보를 활용하여 GNN을 소셜 추천에 적용하였다. 그러나, 본 논문의 주제인 챗터 단위의 강의 동영상 추천에 대한 연구와 관련해서는 GNN을 활용하는 연구는 본격적으로 이루어지지 않은 것으로 조사된다[11].

2. Our Previous works

본 논문의 뿌리가 되는 기존 GNN 기반의 강의 동영상 챗터 추천 연구[11]는 Word2Vec를 활용함으로써 챗터 간의 유사성을 내재시키는 데에 한계가 있었으며 GNN 구조와의 연계도 원활치 못하다는 단점이 있었다.

이와는 달리, 본 논문에서는 TF-IDF(Term Frequency-Inverse Document Frequency)[15]를 활용하여 동영상 챗터 내 스크립트 문장 간의 유사도를 내포하도록 하였으며, 단어의 순서와 이들의 연관된 정보를 보다 효과적으로 캡처하기 위해 N-gram을 적용하고, 출현 빈도를 반영하여 단어의 규모가 발산하지 않도록 하였다.

III. The Proposed Scheme

전술한 바와 같이 동영상 전체가 아닌 그보다 세분화된 챗터 단위의 추천 시 동영상 전체를 추천 단위로 할 때 보다 추천의 정밀도는 높아지지만, 탐색 대상의 규모가 비약적으로 증가하게 되어 성능의 저하가 초래될 수 있다. 본 논문에서는 이러한 문제점을 해결하기 위하여 챗터 별 키워드 세트를 바탕으로 클러스터링 기법과 GNN을 함께 사용함으로써 속도의 저하를 최소화하면서도 추천의 정확도를 향상시키는 방법을 제안한다.

본 논문에서는 챗터 단위 추천을 위하여 Fig. 1과 같이 유사한 특성을 지닌 챗터들을 클러스터로 구성하고 클러스터가 하나가 노드가 되는 GNN을 구성하여 클러스터 간의

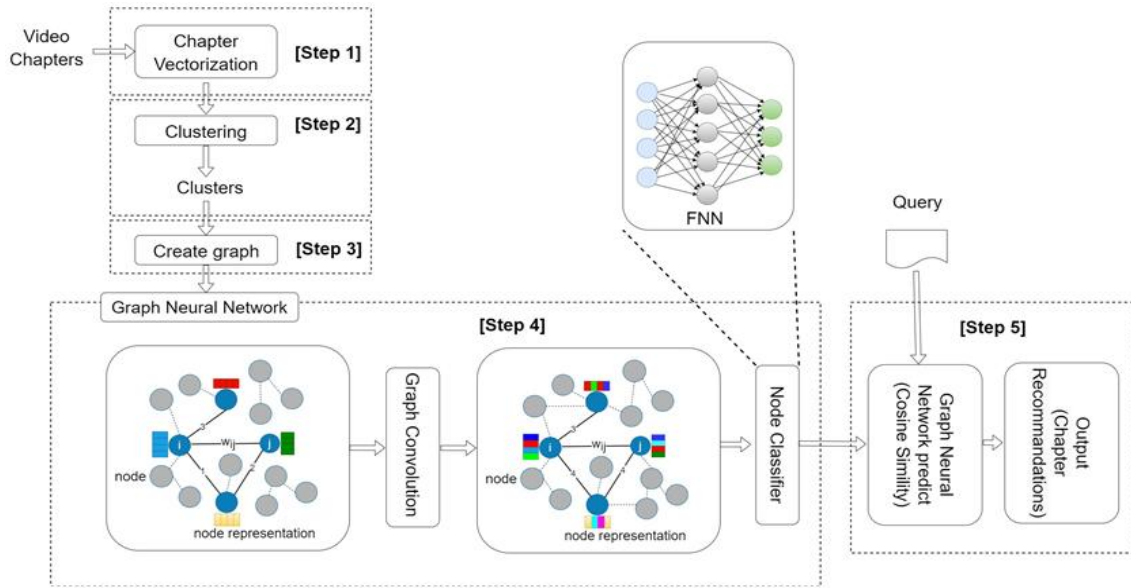


Fig. 1. System Architecture

관계를 추가적으로 세밀하게 학습하도록 하는 방식을 제안한다. 제안하는 방법의 주요 다섯 단계는 다음과 같다.

[Step 1] Chapter Vectorization with TF-IDF

첫 번째 단계에서는 STT(Speech-To-Text)[17]를 통해 각 챕터의 스크립트를 얻은 후 이에 포함된 단어들을 토큰화시키면서[12], 불용어를 제거하고, 빈도수가 일정 수준 이하인 어휘는 제거한다. 또한 모든 챕터에 포함된 단어와 같이 빈도수가 높은 단어도 탐색에 큰 영향을 주지 않을 가능성이 있기 때문에 빈도수가 일정 수준 이상인 단어들은 제거한다. 그런 후, 단어의 순서와 연관된 정보를 효과적으로 캡처하기 위해 N-gram를 적용하고 모든 챕터를 아우르는 어휘 세트를 대상으로 TF-IDF를 적용하여 각 챕터 마다 하나의 고유한 TF-IDF 벡터를 생성토록 한다.

본 연구에서 Word2Vec과 같은 워드 임베딩을 사용하지 않고 TF-IDF를 사용한 이유는 챕터의 특성 때문이다. 즉, 임의 챕터를 놓고 볼 때 챕터의 스크립트는 단순히 몇 개의 단어가 아니라 문장들로 구성되어 있는 바, 워드 임베딩은 단어 간의 의미론적 관계를 학습하는 데 유용하지만, 문장 전체의 의미를 파악하기에는 한계가 있다. 반면, TF-IDF은 단어의 빈도와 문서 전체 속에서 각 단어의 중요도를 계산함으로써 문장 간의 유사도를 파악하는데 유용하기 때문이다.

[Step 2] k-means++ Clustering

두 번째 단계에서는 첫 번째 단계에서 얻은 각 챕터별 TF-IDF 벡터들을 대상으로 k-means++ 알고리즘을 활용

하여 클러스터를 도출한다. 도출된 클러스터의 갯수는 챕터의 개수보다 현저히 적은 숫자가 되어 추천 시스템의 계산량을 감소시켜 성능을 높이는 데에 기여하게 된다.

예를들어, 챕터 508개에 상응하는 TF-IDF 벡터에 대하여 k 값을 25로 하는 k-means++ 알고리즘을 적용할 경우 25개의 클러스터가 생성되어 이후 단계에서의 계산량이 감소하게 된다.

아울러, Centroid Diameter Distance 방법[16]을 이용하여 한 클러스터 내의 챕터들 간에 존재하는 유사도들을 계산한 후, 이들의 평균값을 산출하여 해당 클러스터의 밀집도를 구한다. 이를 통해 얻은 각 클러스터의 밀집도를 임계값과 비교하여 임계값 보다 높은 클러스터들에 대하여 다음 세 번째 단계를 적용한다.

[Step 3] Graph Construction with Clusters

세 번째 단계에서는 GNN을 사용하기 위해 클러스터를 하나의 노드로 하는 그래프를 생성하고 각 노드의 노드표현을 초기화 한다. 이를 위하여 먼저 앞 단계에서 생성된 클러스터 각각에 대하여 클러스터의 중심 좌표와 해당 챕터 내의 단어들 간의 거리를 바탕으로 클러스터를 대표하는 핵심단어 세트를 도출해 낸다. 그런 다음 모든 클러스터의 핵심단어 세트의 합집합에 상응하는 벡터를 노드표현(Node Representation) 벡터로 정의하고, 각 노드마다 해당 클러스터의 핵심단어가 존재하는지 여부를 노드표현 벡터에 0/1로 표시함으로써 각 노드의 노드표현 벡터, 또는 노드표현을 초기화한다. 한편, 클러스터 간의 모든 쌍에 대하여 핵심단어 세트의 교집합을 구하여 그 교집합의

디그리(degree)를 엣지 값으로 설정함으로써 GNN을 위한 그래프를 완성한다.

[Step 4] Applying GNN(Graph Neural Networks)

네 번째 단계에서는 GNN 기법을 이용하여 주어진 그래프 상에서는 아직 존재하지 않는 숨어있는 관계, 즉 새로운 엣지를 발견하거나 엣지 값을 더욱 강화하는 작업을 진행한다. 이에 대한 구체적인 설명은 제4장에 기술한다.

[Step 5] Searching with Cosine Similarity

다섯 번째 단계는 주어진 질문에 대하여 적합한 챗터를 추천하는 단계이다. 이를 위하여 주어진 질문의 TF-IDF 벡터와 각 노드의 노드표현 간의 코사인 유사도를 계산하여 유사도가 가장 높은 노드를 찾고, 해당 노드에 해당하는 클러스터 내의 챗터들 중에서 유사도가 높은 순위대로 추천하게 된다. 이에 대한 자세한 절차는 제4장에 기술한다.

IV. GCN and Recommendation

본 논문에서는 클러스터 간의 관계성을 보다 심도 있게 학습하기 위한 GNN 학습 기법인 GCN(Graph Convolutional Network) 기법을 활용할 것을 제안한다.

1. GCN and Node Classification

GNN은 복잡한 그래프 구조 속에서 노드 간의 관계 속에 숨어있는 의미 있는 정보를 추출하는 데에 매우 뛰어난 성능을 보이는 것으로 학계에 보고되고 있으며, 추천 시스템에서도 빈도 높게 사용되고 있다[1].

GNN 분야의 대표적인 학습 알고리즘인 GCN(Graph Convolutional Network)은 Prepare, Aggregate, Update라는 세 가지의 오퍼레이션이 적용된다. 먼저 Prepare에서는 노드표현과 노드간 가중치(weight)를 활용하여 이웃 노드에게 전달할 메시지를 준비한다. 여기서 노드표현이란 제3장에서 설명된 [Step 3]에 의해 초기화된 벡터로 이후 GCN 과정을 거치면서 결과적으로 노드 임베딩 벡터로 귀결되게 된다.

Aggregate 과정에서는 이웃의 메시지와 현재 노드표현을 바탕으로 해당 노드의 집계된 메시지를 계산한다. 마지막으로 Update 과정에서는 현재의 노드표현과 집계된 메시지를 결합하여 해당 노드의 노드표현을 업데이트 한다. 이렇게 업데이트된 노드표현은 노드의 새로운 상태이자

해당 노드의 새로운 임베딩 벡터가 된다[2]. 이러한 Prepare, Aggregate, Update 과정은 정해진 횟수 만큼 또는 수렴 조건이 만족될 때까지 반복된다.

이상과 같은 과정을 통해 GCN의 결과가 얻어지면 각 노드 임베딩 벡터에 대해서 Node Classifier를 적용한다. 이는 노드 임베딩 벡터의 값들을 로짓(logits)으로 변환하는 것으로 이를 통해 각 노드는 최종적인 노드 임베딩 벡터를 얻게 된다. 이를 위해서는 FNN(Feedforward Neural Network)[14]이 사용된다.

2. Chapter-level Recommendations

GCN에 의해 GNN의 학습이 이루어지면 이를 기반으로 챗터 추천이 이루어진다. 먼저 질문이 입력되면 이의 TF-IDF 벡터와 각 노드의 임베딩 벡터 간의 코사인 유사도를 계산하여 질문과 유사도가 가장 높은 노드, 즉, 최고 유사도 노드를 선정한다. 만약 두 벡터의 원소 배열이 다를 경우 원소들 간에 상응하는 쌍을 찾아 재배치함으로써 코사인 유사도를 계산할 수 있다. 결과적으로 최고 유사도 노드에 상응하는 클러스터 내의 동영상 챗터들이 일차적인 추천 대상이 되므로 추천 대상 세트에 이들을 담는다.

그런 다음, 다시 최고 유사도 노드와 이웃 관계에 있는 노드들에 상응하는 클러스터 내 챗터들도 추천 대상 세트에 담는다. 이 과정에서 최고 유사도 노드로부터의 에지 깊이, 즉, 홉 수와 각 엣지 값을 고려할 수 있다. 본 논문의 실험에서는 홉 수는 1로 설정하였으며, 엣지 값은 고려하지 않았다. 즉, 최고 유사도 노드에 인접된 모든 노드에 상응하는 클러스터 내 챗터들을 모두 추천 대상 세트에 담는 것으로 하여 실험을 진행하였다. 이와 같이 추천 대상 세트가 도출되면 세트 내 동영상 챗터들의 TF-IDF 벡터와 질문의 TF-IDF 벡터 간의 코사인 유사도를 기준으로 우선순위를 매겨 추천을 하게 된다.

이상의 추천 과정은 결과적으로 두 단계를 거쳐서 추천이 이루어지는 것을 의미한다. 첫번째 단계는 최고 유사도에 해당하는 노드를 찾아 추천 대상 챗터 세트를 구축하는 단계이고, 두번째 단계는 추천 대상 챗터들 내에서 추천 순위를 결정하는 단계이다. 이와 같은 2 단계 접근은 추천시의 탐색 공간을 줄여 줌으로써 동영상 챗터의 개수가 증가하거나 클러스터의 개수가 증가하여도 탐색의 속도가 저하되는 것을 방지할 수 있는 역할을 하며, GCN을 통해 발견한 이웃 노드까지도 추천 대상에 고려함으로써 추천의 정확도도 높일 수 있는 효과를 얻을 수 있게 된다.

V. Experimental Comparisons

1. Experiment Facilitations and Dataset Preparations

Preparations

본 연구에서는 실험을 위하여 유튜브의 강의 동영상을 바탕으로 챕터 단위의 데이터셋을 구축하였다. 유튜브에서 강의 형태를 갖춘 동영상 중 무작위로 선택된 127개의 강의 동영상으로부터 총 508개의 챕터를 추출하였으며, 이 데이터셋을 학습과 실험에 활용하였다. 먼저 STT[17]를 활용하여 모든 챕터로부터 스크립트를 자동으로 추출하였으며 추출한 스크립트에서 불용어와 구두점을 제거하는 전처리 과정을 거쳤다. 실험 환경 구성은 Table 1과 같이 비교적 평범한 PC 환경에서 실험을 진행하였다.

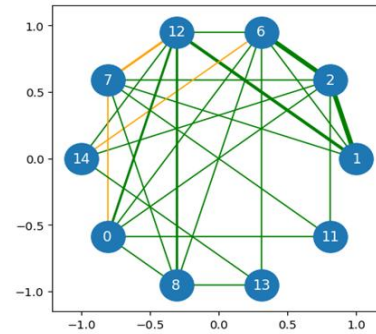
Table 1. Experimental Facilitations

Category	Specification
CPU	Intel(R) Core(TM) i5-10400 CPU @ 2.90GHz 2.90GHz
RAM	16GB
HHD/SSD	SSD
OS	Windows 10 Pro 64 bit
Development Tool	Google Colaboratory, Python3, 3.8.16
Lecture Video(Chapters)	127개(508개)

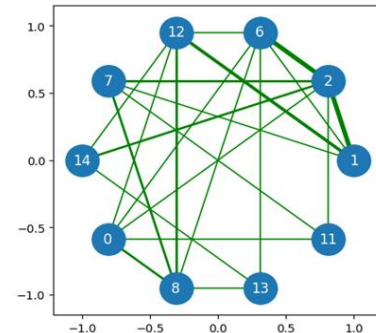
2. Experiments on Edge Enhancement

우선 본 논문에서 사용하는 GCN이 엣지 생성 및 강화는 GNN의 기본 기능을 제대로 수행하는지를 실험하였다. 이를 위하여 엣지들을 무작위적으로 임의로 제거한 후, GCN이 이를 얼마나 제대로 복원하는지를 확인하는 실험을 진행하였다.

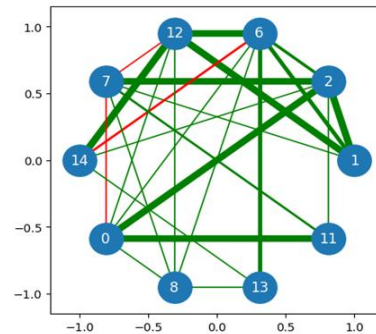
Fig. 2를 볼 때 그래프 내에서 (0, 7), (6, 14), (7, 12)에 해당하는 엣지를 삭제한 후 GCN을 수행하였다. 그 결과 Fig. 2(c)에서 보는 바와 같이 삭제된 3개의 엣지가 정상적으로 복원되는 것을 관찰할 수 있었다. 한편, GNN은 노드와 엣지 정보를 활용하여 미처 발견되지 않은 새로운 관계를 모델링하는 데 탁월한 능력을 보인다. 이러한 능력은 복잡한 관계와 구조를 갖는 데이터에서 숨은 연결성을 파악하는데 큰 장점을 지닌다. 예를 들어 Fig. 2(c)를 다시 볼 때 (0, 11), (12, 14)와 같이 기존에 존재하던 엣지의 값(굵기)도 강화된 것을 볼 수 있다. 이를 볼 때 GNN 고유의 장점인 노드 간의 숨은 관계를 생성하고 강화하는 기능이 실질적으로 발휘될 것을 확인할 수 있었다.



(a) Before Edge Deletions



(b) After Edge Deletions



(c) After Edge Recoveries

Fig. 2. Edge Recovery Experiments

3. Comparisons: GNN vs. Cluster-based approaches

본 실험에서는 클러스터링 결과만을 사용했을 경우와 GNN을 적용했을 경우에 대한 비교 실험에 주안점을 두었다. 이를 위해 MRR(Mean Reciprocal Rank)[13] 지표를 활용하였다. MRR은 추천 시스템의 성능을 정량적으로 평가하는 데 자주 사용되는 지표로, 추천된 챕터 중 정답에 해당하는 챕터를 어떤 순위에 배치하였는지를 측정하는 지표이다. 본 논문에서는 비교 대상인 두 방법, 즉, 단순 클러스터링 방법과 GNN을 적용한 방법 간의 성능을 비교하기 위해 도입되었으며 비교 결과는 Fig. 3과 같다. 질문의 다양성을 높이고 더 정확한 실험 결과를 얻기 위해 Table 2와 같이 서로 다른 총 11개의 질문을 활용하였다.

Table 2. Question Lists for Experiments

No	Questions
1	How does Hadoop address the challenges posed by big data, particularly in terms of storage and processing?
2	How can understanding Python help individuals interested in image processing?
3	What inspired Rachel to pursue a career in AWS, and what challenges did she face along the way?
4	How many weights and biases are involved in the network discussed, and why are they important?
5	In what ways does Google Cloud's combination of infrastructure and software as a service offerings provide a competitive advantage in the market?
6	Where is RAM located in relation to the motherboard?
7	What is natural language processing (NLP)?
8	What are some of the key themes and concepts that the Professor of International Art Education at Alto University will explore in his talk, particularly in relation to art education, visual culture, critical pedagogy, media studies, and the concept of culture jamming?
9	Why is it suggested to implement a few algorithms from scratch in Python using only pure Python and NumPy?
10	How does the TF upgrade V Two tool assist in transitioning existing TensorFlow 1.12 Python scripts to TensorFlow 2.0, and where can users report any issues they encounter during the conversion process?
11	How does TensorFlow Extended (TFX) assist in putting machine learning models into production, and what does Google's decision to open source TFX mean for the broader machine learning community?

Fig. 3에서 볼 때 GNN 기법은 노드 수가 25일 때 MRR 최고값인 0.91을 나타내었으며, 반면 클러스터링 기법은 약 0.76을 나타내었다. 이는 GNN이 정답에 해당하는 챗터를 평균적으로 더 높은 순위에 위치시킨다는 것을 나타내는 것으로, GNN을 사용하는 것이 추천의 정확도를 높이는 데에 효과적임을 시사하고 있다. 한편, Fig. 3에서 볼 수 있듯이 클러스터 개수가 예를들어 10개로 줄어들 경우 MRR 값이 급격히 낮아지는 것으로 나타났다. 이는 클러스터의 수가 감소함에 따라 각 클러스터가 대표하는 특성이 너무 일반화되거나 세분화되지 않아 챗터 간의 유사성을 올바르게 파악하기 어려워졌기 때문인 것으로 보인다. 이는 추천의 정확도에 있어서 클러스터 개수가 중요한 변수로 작용하며, 적절한 클러스터 개수를 선택함으로써 추천 시스템의 성능을 향상시킬 수 있음을 시사하고 있다.

4. Impact of Node Representation Vector Size on MRR

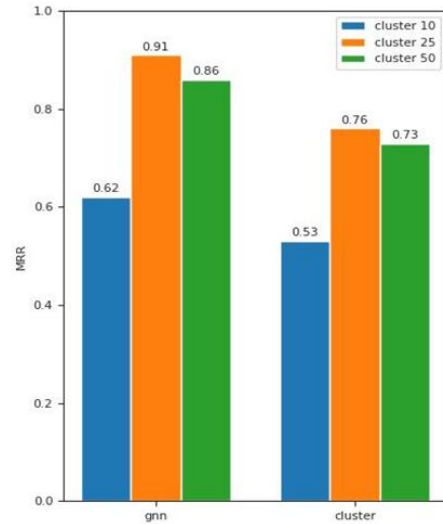


Fig. 3. MRR Comparisons

GNN을 사용하는 추천 시스템에 있어서는 제4장에서 언급된 노드표현 벡터의 크기도 추천의 정확도에 영향을 미치는 중요 변수에 해당된다. 본 논문에서 노드표현은 키워드의 세트에 해당하는 특징벡터로 정의되며 각 노드는 특징벡터 상에 각 키워드의 해당 여부를 0/1로 표현한 벡터를 각 노드표현 벡터의 초기값으로 설정된다.

본 논문에서는 노드표현 벡터 크기의 변화, 즉, 키워드의 개수의 변화에 따른 MRR 값의 변화를 실험하였다. 실험결과 Fig.4에서 볼 수 있듯이 키워드의 개수를 50개로 하였을 때 MRR의 성능이 최고치를 보임을 알 수 있었으며, 그보다 적거나 많아질 때 MRR 성능은 감소하는 것으로 나타났다. 이러한 실험결과는 키워드 개수가 추천시스템의 정확도에 큰 영향을 미친다는 점을 시사하고 있다.

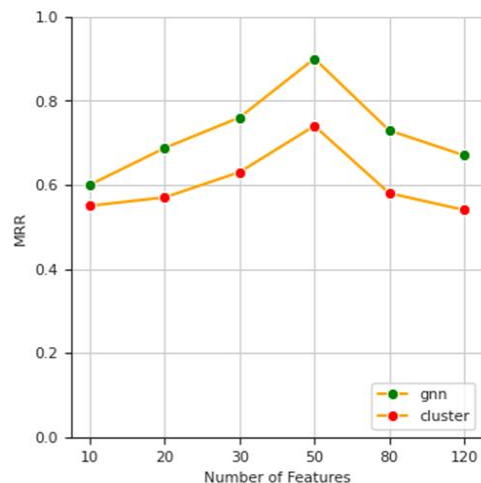


Fig. 4. MRR variations in number of features

Fig.3의 실험결과와 Fig. 4의 실험결과를 종합해서 볼 때 실험에 사용된 강의 동영상 챕터 세트에 대해서는 노드 수가 25개이면서 키워드 수는 50개 일 때 최고의 MRR 값을 보이는 것으로 나타났다.

5. Experiments for Precision

제3장에서 전술한 바와 같이 본 논문에서 구축하는 강의 동영상 챕터의 추천은 추천값을 수반한다. 즉, 추천되어지는 챕터들은 코사인 유사도 값을 갖고 있으며 이는 [0,1] 구간의 실수 값이다. 이러한 유사도 값을 바탕으로 하는 성능 지표는 추천 순위를 바탕으로 하는 MRR와는 달리 추천 시스템의 성능을 보다 총체적으로 나타낼 수 있다.

본 논문에서는 유사도 값을 기반으로 하는 성능지표를 정의하기 위하여 아래와 같은 두 개의 임계값을 설정하였다.

A_{th} : Recommended Threshold

B_{th} : Relevant Threshold

A_{th} 는 Recommended Threshold로 [0,1] 구간 내의 값이며, 추천된 모든 챕터들 중에 질의와 무관하다고 가정되는 챕터들을 제외하기 위한 임계값이다. 본 실험에서는 $A_{th}=0.2$ 로 설정했으며, 이는 유사도 값이 0.2 이하인 챕터는 질의 내용과 무관한 챕터로 가정했다는 의미이다.

B_{th} 는 Relevant Threshold로 [0,1] 구간 내의 값이며, 이는 추천되어 나온 모든 챕터 중에서 질의의 내용과 유사성이 높다고 판단되는 챕터들을 한정짓기 위한 임계값이다. 따라서 $[B_{th}, 1]$ 구간에 해당하는 챕터는 주요 추천 챕터에 해당하며, $[A_{th}, B_{th}]$ 구간의 챕터는 단순 추천 챕터에 해당한다.

이러한 단순추천 구간을 고려하는 이유는 $[B_{th}, 1]$ 에 해당하는 주요추천 챕터의 유사도와 $[A_{th}, 1]$ 에 해당하는 챕터의 유사도 간의 비율을 보기 위함이다. 해당 비율이 높을수록 주요챕터의 추천이 의미 있는 것으로 볼 수 있기 때문이다. 이상의 개념은 아래의 수식으로 표현되며, 이는 정보검색의 여러 가지 평가지표 중 하나인 정밀도 (precision) 지표에 실수를 반영한 지표로 볼 수 있다. 본 논문에서는 이를 편의상 $Precision_{Real}$ 라 부른다.

$$Precision_{Real} = \frac{\sum_{\{i | similarity(i) \geq B_{th}\}} similarity(i)}{\sum_{\{j | similarity(j) \geq A_{th}\}} similarity(j)}$$

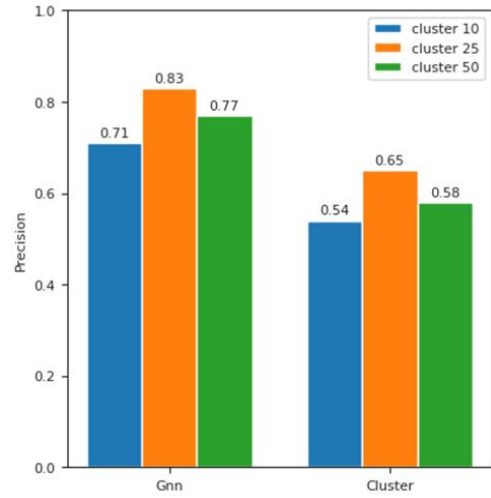


Fig. 5. Experiments on Precision in Real

Fig. 5에서 볼 수 있듯이 GNN을 적용했을 경우 $Precision_{Real}$ 또한 상대적으로 높은 성능을 보였으며, 노드의 개수가 25개이면서 키워드가 50개 일 때 가장 높은 값을 보여 MRR 실험결과와 일관성을 보이는 것으로 나타났다.

또한 Fig. 6에서 볼 수 있듯이 B_{th} 의 값을 0.4에서부터 0.9까지 변화를 주며 $Precision_{Real}$ 의 변화를 관찰했을 때 일정 구간, 즉, 예를들면 $B_{th}=0.7$ 전후 부근과 같이 GNN의 $Precision_{Real}$ 값이 GNN을 사용하지 않았을 때에 비하여 월등히 높은 구간이 존재하는 것으로 관찰되었다. 이는 GNN 기반의 강의동영상 챕터 추천 시스템을 구축할 때 적절한 B_{th} 를 설정함으로써 정밀도 차원에서 추천 결과를 더 효과적으로 활용할 수 있음을 시사하는 것으로 볼 수 있겠다.

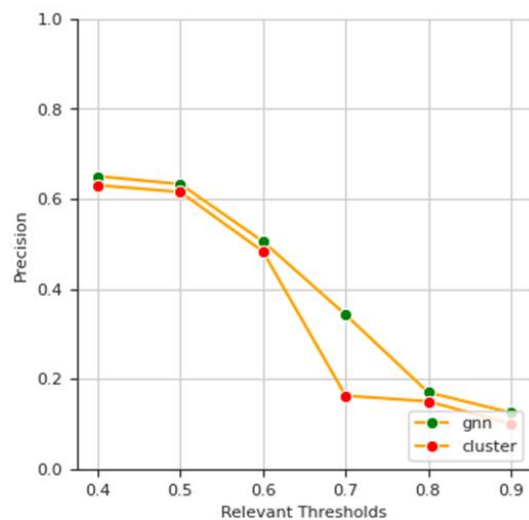


Fig. 6. Variations of Precision in Real over Relevant thresholds

VI. Demonstration System

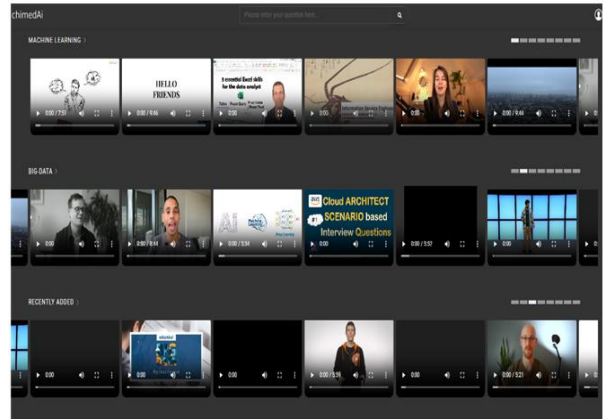
상기의 과정을 통하여 얻어진 실험결과를 바탕으로 웹 환경에서 강의 동영상 챗터를 추천하는 시연시스템을 개발하였다. 홈 화면(Fig. 7(a))에서 본 연구의 실험에서 사용한 508개 강의 동영상의 일부를 확인할 수 있다. Fig. 7(b)는 “what is machine learning?”이라는 학습자의 질문에 가장 유사도가 높은 노드와 챗터들을 탐색하여 유사도가 높은 챗터를 순위대로 추천한 챗터 목록을 예시로 보여준다. 목록 상의 챗터들 중에서 한 챗터를 클릭하면 해당 챗터가 속한 강의 동영상으로 이동한 후 해당 챗터부터 플레이 되게 된다. 한편, 재생 목록은 일련의 질의 과정을 통해 얻어진 여러 동영상들을 한데 모은 목록으로, 특정 주제나 관심 있는 내용의 동영상 챗터들을 모아 연속적으로 재생할 수 있다(Fig. 7(c)). 본 시스템은 현재 오픈 소스 소프트웨어로 공개되어 있다¹⁾.

VII. Conclusions

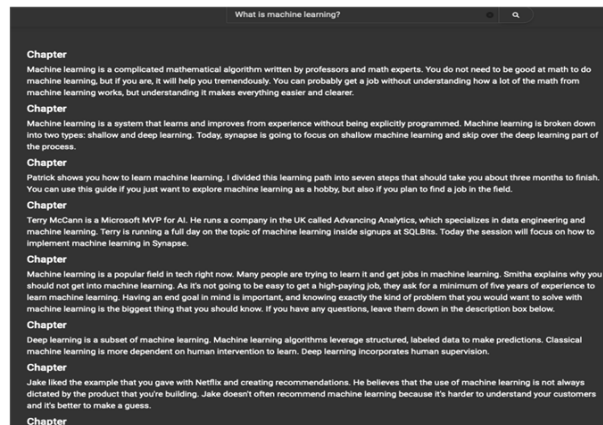
본 연구는 챗터 단위의 강의 동영상 추천 시스템의 성능 향상을 목적으로 다양한 알고리즘과 GNN 기술을 조합하여 추천의 정확도와 속도를 동시에 추구하는 방안을 제시하였다.

첫 번째로, TF-IDF 기법과 클러스터링 기법을 이용하여 유사한 챗터들을 그룹화하여 추천 시의 탐색공간을 줄임으로써 속도의 저하를 방지하였다. 두 번째로, 클러스터를 노드로 하는 그래프를 구성하고 이에 GNN 기법을 활용하여 클러스터 간의 연결성을 학습하도록 하였다. 마지막으로, 그래프 노드 차원의 유사도 측정과 챗터 차원 유사도 측정, 즉, 2단계의 코사인 유사도 적용을 통해 추천의 속도를 높이면서도 학습자의 질의에 보다 일치하는 챗터를 추천하도록 하였다.

실험 결과 GNN과 클러스터링 기법을 조합한 추천 방식이 클러스터링 만을 사용하는 방식에 비하여 MRR 지표에서 약 19.7% 증가하였으며, 유사도 기반의 정밀도에 있어서 약 27.7% 증가하는 결과를 확인할 수 있었다. 본 논문의 접근법과 연구 결과가 강의 동영상 중심의 학습시스템 구축에 있어서 학습자의 질의에 보다 적합한 동영상 챗터를 추천하는 데에 기여할 수 있을 것으로 기대한다.



(a) Home page of Demonstration System



(b) List of Recommended Video Chapters



(c) Replaying a Selected Chapter

Fig. 7. Demonstration Displays

한편, 본 논문은 강의동영상 세트가 고정적으로 주어진 경우에 대하여 연구가 진행되었고 이를 바탕으로 시연 시스템도 구현되었다. 그러나, 강의동영상 추천 시스템이 보다 유용하게 사용되기 위해서는 새로운 강의동영상이 연

1) <https://github.com/chimeddor/recommendation-system-videos-chapter>

제든지 추가 될 수 있어야 할 것이다. 이를 위해서는 챗터 클러스터의 동적 변화와 이에 따른 GNN의 점진적 학습 (Incremental Learning) 기법[18]에 대한 연구가 필요할 것이다. 이와 같은 연구는 향후 연구 주제로 남겨놓도록 하겠다.

ACKNOWLEDGEMENT

This paper was written as part of Konkuk University's research support program for its faculty on sabbatical leave in 2023.

REFERENCES

- [1] Wang, X., et al., "Neural Graph Collaborative Filtering." Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, pp.165-174, July 2019. DOI: 10.1145/3331184.3331267
- [2] Ying, R., et al., "Graph Convolutional Neural Networks for Web-Scale Recommender Systems." Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2018. DOI: 10.1145/3219819.3219890
- [3] Wang, X., et al., "Deep Learning for Recommender Systems: A Survey and New Perspectives." ACM Computing Survey, Vol. 52, Issue 1, pp.1-38, 2019. DOI: 10.1145/3285029
- [4] Choi, Yerin; Yeen, Yeen-heui; Kim, Dong-Guem, "Dialogue System for User Customized Lecture Recommendation." KIICE, pp.84-86, Oct. 2022.
- [5] Davidson, James; Liebold, Benjamin; Liu, Junning; Nandy, Palash; VanVleet, Taylor; Gargi, Ullas; Gupta, Sujoy; He, Yu; Lambert, Mike; Livingston, Blake; et al., "The YouTube video recommendation system." RecSys '10, Proceedings of the fourth ACM conference on Recommender systems, 2010. DOI: 10.1145/1864708.1864770
- [6] Rendle, Steffen, "Factorization machines." Proceedings of ICDM, 2010. DOI: 10.1109/ICDM.2010.127
- [7] Guo, Hui Feng; Tang, Ruiming; Ye, Yunming; Li, Zhenguang; He, Xiuqiang, "DeepFM: a factorization-machine based neural network for CTR prediction." Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, pp.1725-1731, 2017. DOI: 10.24963/ijcai.2017/239
- [8] Song, Weiping; Shi, Chence; Xiao, Zhiping; Duan, Zhijian; Xu, Yewen; Zhang, Ming; Tang, Jian, "AutoInt: Automatic feature interaction learning via self-attentive neural networks." Proceedings of CIKM, 2019. DOI: 10.1145/3357384.3357925
- [9] Wu, Shu; Tang, Yuyuan; Zhu, Yanqiao; Wang, Liang; Xie, Xing; Tan, Tieniu, "Session-based Recommendation with Graph Neural Networks." Proceedings of AAAI, 2019. DOI: 10.1609/aaai.v33i01.3301346
- [10] Fan, Wenqi; Ma, Yao; Li, Qing; He, Yuan; Zhao, Eric; Tang, Jiliang; Yin, Dawei, "Graph Neural Networks for Social Recommendation." Proceedings of WWW, pp. 417-426, 2019. DOI: 10.1145/3308558.3313488
- [11] Chimeddorj, P., "Chapter-Level Lecture Video Recommendation System Using Graph Neural Network", MS Thesis, Konkuk University, Seoul, 2023. <http://konkuk.dcollection.net/common/orgView/200000684126>
- [12] Natural Language Toolkit, <https://www.nltk.org>
- [13] Shi, Y., Karatzoglou, A., Baltrunas, L., Larson, M., Oliver, N., & Hanjalic, A., "Climf: learning to maximize reciprocal rank with collaborative less-is-more filtering." Proceedings of the sixth ACM conference on Recommender systems, pp. 139-146, 2012. DOI: 10.1145/2365952.23659
- [14] Schmidhuber, J., "Deep learning in neural networks: An overview," Neural Networks, 61, pp.85-117, 2015. DOI: 10.1016/j.neunet.2014.09.003
- [15] Deldjoo, Y., Schedl, M., Cremonesi, P. and Pasi, G., "Recommender systems leveraging multimedia content," ACM Computer Survey, vol. 53, no. 5, pp. 1-38, Sep. 2020. DOI: 10.1145/3407190.
- [16] Bolshakova, N.; Azuaje, F.; "Cluster validation techniques for genome expression data." Signal Processing, Vol. 83, Issue 4, 2003. DOI: 10.1016/S0165-1684(02)00475-9
- [17] <https://www.assemblyai.com>
- [18] Ahrabian, K., et al., "Structure aware experience replay for incremental learning in graph-based recommender systems," CIKM, pp. 2832-2836, 2021.

Authors



Purevsuren Chimeddorj received the B.S. degree in Computer Science from Huree University of ICT(Information and Communication Technology), Mongolia in 2020, and M.S. degrees in Computer Science

and Engineering from Konkuk University in Seoul, Korea, in 2023. He has participated in various projects, including the Collaboration E-Learning Platform with Video Telephony, Multi-chatbot Service on AI Cloud, and the Development of a Chapter-level Lecture Video Recommendation System. His research interests include web-based intelligent recommendation systems and cloud computing.



Doohyun Kim received the B.S. degree in Computer Engineering from Seoul National University, Korea, in 1985, M.S. and Ph.D. degrees in Computer Science from KAIST(Korea Advanced Institute of Science

and Technology), Korea, in 1987 and 2003 respectively. He is currently a professor in the Department of Computer Science and Engineering at Konkuk University in Seoul, Korea. Before joining Konkuk University in 2004, he was a Principal Researcher at the Electronics and Telecommunications Research Institute (ETRI). His research interests include embedded intelligence and cloud computing.