언어 모델을 이용한 빈발 패턴 마이닝*

배현진O 김철연
숙명여자대학교
gloria9705@naver.com cykim@sookmyung.ac.kr

Frequent Pattern Mining using Language Model

Hyunjin BaeO Chulyun Kim
Sookmyung Women's University

요 약

빈발 패턴 마이닝에 있어 가장 중요한 일은 바로 빈발하게 발생하는 아이템 집합을 찾는 것이다. 빈발 집합을 찾는 유명한 기법으로는 Apriori나 FP Growth와 같은 기법들이 있지만, 계산 복잡도가 높고 지지도가 하이퍼 파라미터라는 점에서 문제가 발생한다. 따라서 일일이 계산을 해주지 않아도 빈발 집합을 찾을 수 있는 방법을 고안해보았고, 언어 모델을 사용해 빈발 집합을 생성해보았다. 본 논문에서는 아이템과 단어가 같은 맥락이라는 것을 고려해 언어 모델을 사용해 아이템 집합을 학습해 아이템셋을 만들어냈고, 기존의 기법과 비교해 성능을 비교해 언어 모델 기법을 빈발 패턴 마이닝에 적용하는 것이 유의미한 작업이라는 것을 밝혔다.

1. 서 론

빈발 패턴 마이닝은 데이터 마이닝 분야에서 가장 중요한 문제 중 하나이다. 이때 빈발 패턴을 찾기 위해서는 어떤 데이터셋에서 빈발하게 발생하는 아이템 집합, 즉 빈발 아이템 집합을 찾는 것이 가장 중요한 일이다. 가장 유명한 기법으로는 Apriori 알고리즘, FP Growth 알고리즘 등이 있다.

하지만 기존의 연관 규칙 기법들은 문제점들을 가지고 있다. 일단 Apriori 알고리즘은 모든 후보 아이템 집합에 대해 계산을 하기 때문에 계산복잡도가 매우 높으며, 더 많은 계산 공간을 필요로 한다.[1] 이를 보완하기 위해 FP Growth 알고리즘이 등장했지만, 매우 긴 패턴을 가진 데이터에서는 성능이 좋지 않다는 단점이 있다. 또한 기존의 기법들은 최소 지지도를 기준으로 해당 아이템 집합이 빈발한 지를 구별한다. 하지만 데이터셋마다 가장 이상적인 최소 지지도를 구하는 것은 현실적으로 불가능하며, 지지도를 기준으로 경계에 있는 아이템 집합 간의 경중을 나누는 것은 현실적이지가 않다.

최근 딥러닝 분야가 폭발적으로 성장하며 학습을 통한 패턴을 인식하는 방법들이 다양한 분야에서 고안되고 있다. 그 중 하나가 자연어 처리 분야이다. 언어를 이해하고, 분석하고, 처리하기 위해선 단어 시퀀스의 출현 패턴을 분석하는 과정이 필요하다. 이때 단어의 시퀀스와 아이템 집합을 동일한 맥락에서 이해할 수 있다. 단어를 아이템이라고 생각하면, 해당 시퀀스에서 시간과 공간의 정보를 제거한 것이 아이템 집합이 된다. 여기에서 착안해서 마이크로소프트사에서는 item2Vec이라는 아이템 기반의 협업 필터링 기법에 SGNS(Skip-gram with Negative Sampling)를 적용한 기법을 소개하기도 했다.[2] Word2Vec 이후, 자연어 처리 분야에서 다양한 task들을 해결하기 위한 다양한 모델들이 고안되고 있으며, ELMo[3], BERT[4] 등은 최고의 성능을 기록하기도 했다.

본 논문에서는 기존의 기법과 다양한 언어 모델을 적용해 빈발 아이템 집합을 생성해보고 비교해 빈발 패턴 마이닝에 학습을 적용하면 어느 정도의 성능을 보일 수 있는지를 확인해보고자 한다. 기존의 최소 지지도에 기반한 빈발한 아이템을 베이스라인으로 확률론적 언어 모델, LSTM을 사용한 언어 모델, GPT가 만들어 낸 빈발 집합을 비교했으며, 데이터셋으로는 영화에 대한 유저 별 평가를 제공하는 MovieLens 데이터셋을 사용했다.

2. 데이터셋[5]

미네소타 대학교의 GroupLens Research에서 만든 MovieLens는 유저 기반 영화 추천 시스템이다. 본 논문에서 사용한 데이터셋은 ml-latest-small 데이터셋으로 1996년 3월부터 2018년 9월까지 수집된 610명의 유저의 100,836개의 평점과 9742개의 영화로 구성되어 있다.

하지만 기존의 언어 모델을 그대로 사용하면 문제가 발생한다. 언어 모델은 시간과 공간적 정보까지 포함해 패턴을 분석하기 때문이다. 예를 들어 '나는 학교에 간다'라는 문장이 있을 때, 언어 모델이 '나는 간다'라는 문장을 만들어낼 확률이 매우 작다. 하지만 빈발 패턴 마이닝에서는 '나는 간다'라는 아이템 집합에 대해서도 지지도가 커지게 된다. 따라서 본 논문에서는 한 아이템 집합에 대해 부분 집합을 만들어 그 부분 집합도 학습 데이터로 사용했다.

지지도가 0.27 이상인 영화 10개를 뽑아 그 영화에 대한 평가만 사용했으며, 해당 영화에 대해 평점이 4점 이상인 경우에 해당 유저의 아이템 집합에 해당 영화를 추가했다. 총 437명의 평가를 바탕으로 아이템 집합을 만들었다. 그 아이템 집합의 각 집합에 대해 부분 집합을 생성해 주었으며, 그렇게 생성된 아이템 집합은 총 38,491개이다.

* 본 연구는 문화체육관광부 및 한국저작권위원회의 2020년도 저작권연구개발사업의 연구결과로 수행되었음(2019-CONTEXT-9500).

3. 모델

언어 모델(Language Model)이란 어떤 단어 나열에 확률을 부여해 특정한 단어의 시퀀스에 대해 그 시퀀스가 일어날 가능성이 어느 정도인지, 얼마나 자연스러운 단어 순서인지를 확률로 평가하는 모델을 말한다. 본 논문의 실험에는 3개의 종류의 모델이 사용되었다.

3.1 확률론적 언어 모델

첫번째는 마르코프 체인 모델(Markov Chain Model)이라고도 불리는 확률론적 언어 모델이다. 확률론적 언어 모델은 이전에 출현한 일련의 단어들에 대해 조건부 확률을 이용해 다음에 출현할 확률이 가장 높은 단어를 구하는 언어 모델이다.[6] 오직 직전 단어에만 의존해 확률로 다음 단어를 예측하기 때문에 이전의 맥락이 고려되지 않는다.

본 논문에서는 직전 한개의 단어에 의존해 다음 단어가 정해지는 bigram 모델을 사용했으며, nltk 패키지의 ngrams 모듈을 사용했다.

3.2 LSTMLM[7]

두번째 모델은 LSTM을 이용한 언어 모델이다. LSTM은 순환 신경망(RNN)에 게이트(gate)라는 구조를 더해 RNN이 가지고 있는 기울기 소실/폭발 문제를 방지할 수 있는 계층이다. 단순 RNN 계층에서는 시간에 따라 기울기가 소실되어 중요한 정보를 잃어버려 장기 의존 관계를 학습할 수 없는 경우가 발생한다. 또 구조 상 행렬 곱 연산이 반복되기 때문에 오버 플로우를 일으키는 경우 또한 발생한다. 이를 해결하기 위해 LSTM에서는 게이트를 도입했으며, 게이트는 데이터의 흐름을 제어할 수 있어 장기 의존 관계를 학습할 수 있다.

본 논문에서는 1개의 LSTM 계층으로 이뤄진 간단한 모델을 사용했으며, 10,000 에폭 동안 학습을 진행했다.

3.3 GPT-2

마지막 모델은 GPT-2이다. LSTM이나 GRU를 도입한다고 해도, 여전히 단어 시퀀스가 길어질수록 중요한 정보가 사라지는 경우가 발생한다. 또 이 계층들로 만든 Seq2Seq은 인코더와 디코더 단에서 항상 같은 길이의 벡터를 사용해 입력 시퀀스를 처리하게 되는데, 만약 시퀀스에 길이가 길어진다면, 이 고정된 벡터에 그 정보들을 다 담기가 힘들다. 그래서 만들어진 것이 바로 '어텐션(Attention)'이다. 어텐션은 모든 정보가 아닌, 꼭 필요하고 중요한 정보에만 주목해 시퀀스를 변환하는 구조이다. 이 어텐션 구조에서, 불필요한 순환 구조를 제거한 것이 셀프 어텐션(self-attention) 구조이며, 이 구조를 가지고 있는 인코더-디코더 모델을 트랜스포머(transformer)라고 한다.[8] 트랜스포머가 처음 고안되고 난 후 대부분의 state-of-the-art 모델들은 이 구조를 채택하고 있다. GPT-2 또한 OpenAI에서 만든 트랜스포머(transformer) 기반의 언어 모델로, 기존의 GPT와는 비슷한 구조를 가졌지만 사이즈를 키워서 학습한 모델이다.

본 논문에서는 가장 작은 124m 모델로 1,000 에폭 동안 학습을 진행했다.

4. 실험 및 평가

비교 베이스라인으로는 연관 규칙 알고리즘을 이용했다. 최소 지지도가 0.1 / 0.05 / 0.01 / 0.005인 아이템 집합들을 빈발 집합으로 두고, 각 모델들이 생성한 itemset들과 비교해 얼마나 유사 한지를 비교해보았다. 최소 지지도 별 총 아이템 집합 개수는 45 / 165 / 626 / 821개이다.

각 언어 모델 별로 학습에 사용한 9가지 영화 아이템('110', '260', '296', '318', '356', '527', '593', '1196', '2571')을 시작 아이템으로 주고 아이템 집합 생성을 진행했다. 이때 학습이 아이템 아이디 크기 순으로 진행되었기 때문에 가장 큰(마지막) 아이템인 '2959'는 시작 아이템에서 제외했다. 그 다음엔 생성된 집합들을 각 지지도의 빈발 집합 개수 만큼 랜덤으로 뽑아 사용했다. 그렇게 각 언어 모델 별로 생성한 아이템 집합(문장)을 언어 모델이 만들어낸 빈발 집합이라고 보았다. 아이템 집합은 정렬되어 학습 및 생성되었다. 중복되는 아이템 집합은 그대로 사용했으며, 아이템 두개 이상으로 구성된 아이템 집합만 사용했다.

Recall과 precision는 다음과 같은 식으로 계산했다.

$$\text{recall} = \frac{\text{ii}}{\text{i}}, \qquad \text{precision} = \frac{\text{iii}}{\text{ii}}$$

i. 연관 규칙 알고리즘이 만들어낸 집합 갯수
ii. 각 언어 모델이 만들어낸 집합 중 연관 규칙 알고리즘이 만들어낸 집합과 일치하는 갯수
iii. (ii)에서 중복 집합 제외

결과는 다음과 같다.

평가	최소지지도	확률론적	LSTMLM	GPT
recall	0.1	1.000000	1.000000	1.000000
	0.05	1.000000	1.000000	1.000000
	0.01	1.000000	1.000000	0.993610
	0.005	0.995128	0.970767	0.995128
precision	0.1	0.488888	0.311111	0.200000
	0.05	0.406060	0.309090	0.230303
	0.01	0.364217	0.274760	0.241157
	0.005	0.341493	0.265997	0.252141
만들어 낸 빈발 집합 갯수	0.1	45	45	45
	0.05	165	165	165
	0.01	626	626	622
	0.005	817	797	817
중복되지 않은 집합 갯수	0.1	22	14	9
	0.05	67	51	38
	0.01	228	172	150
	0.005	279	212	206

그림 1 전체 실험 결과

Recall은 세 모델 모두에서 높았다. 즉 해당 언어 모델이 만들어내는 아이템 집합이 연관 규칙이 만들어내는 아이템 집합과 일치하는 비율이 높다는 의미이다. 최소 지지도가

낮아질 수록 값이 감소하긴 하지만 그렇게 큰 폭으로 감소하지는 않았다.

하지만 precision값은 recall 값에 비해 상대적으로 작았다. 첫번째로 해당 언어 모델이 중복된 아이템 집합을 많이 만들어 냈기 때문이다. 두번째로는 언어 모델이 만들어 낸 아이템 집합 중 최소지지도에 기반한 빈발한 아이템 집합에 포함 되지 않는 경우, 언어 모델이 만들어 낸 아이템 집합의 부분집합이 연관규칙에서 등장하는 경우가 많았기 때문이다.

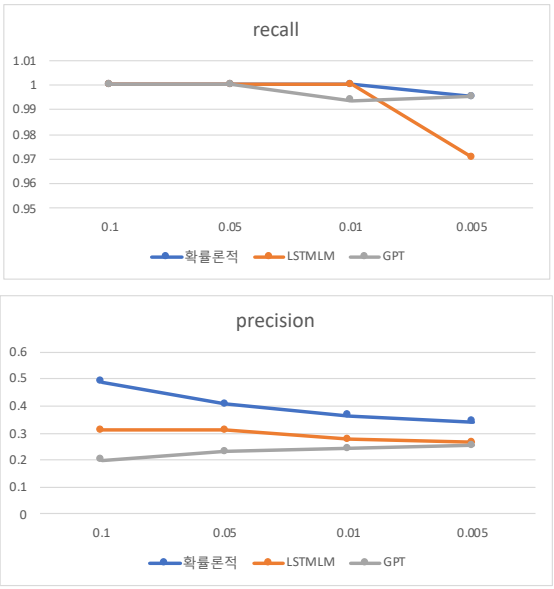

그림 2 Recall과 Precision 그래프

LSTM이 생성해낸 문장	빈발 집합에 포함되는 집합
['110', '2571', '2959']	[110, 2571], [2571, 2959]
['110', '260', '296', '2571']	[110, 260], [260, 296, 2571]
[110, 296, 2571, 2959]	[110, 296], [110, 296, 2571], [296, 2571, 2959], [2571, 2959]

그림 3 LSTMLM이 만들어 낸 집합 중 그 부분집합이 빈발 집합에 포함되는 경우 예시

또 보통 recall이 감소하면 precision이 증가하는 반비례 관계를 가지고 있지만 그런 관계를 보이지 않았다. 학습 데이터가 작고, 네트워크도 작기 때문이다. 하지만 현재 데이터가 작음에도 recall 값이 높고, 연관 규칙이 만들어 낸 집합이 아니더라도 부분 집합인 경우가 많음을 고려한다면, 학습 데이터가 많아지고, 모델을 좀 더 키워서 더 많은 시간을 학습하는 경우, 각 모델 별로 더 많은 집합을 생성해 낼 수 있을 것이며, precision 또한 올라가 더 다양한 빈발 집합을 생성해 낼 수 있을 것이다.

5. 결론

'빅데이터'의 개념이 들어오면서 빈발 패턴 마이닝에 있어서 큰 변화를 맞았다. 큰 데이터 집합을 다룰 수 있도록 기존의 알고리즘을 응용해 사용하고 있으며, 멀티 쓰레드 구조를 사용한 병렬 알고리즘인 BigFIM[9] 등 다양한 알고리즘들이 등장하고 있다. 하지만 데이터는 점점 더 많아지고, 새로운 아이템들이 매 순간 등장하며, 계속 빈발 집합을 갱신해야 하는 상황이 발생하고 있다.

반면에 최근 들어 폭발적으로 성장한 언어 모델들이 GLUE, SQuAD등의 벤치마크들을 계속해서 갱신하고 있으며, 특히 Google에서 발표한 T5는 SuperGLUE에서 인간과 비슷한 성능을 갱신하기도 했다.[10] 위에서 말했듯, 아이템과 단어는 비슷한 맥락을 공유한다. 아이템은 단어에 비해 시간과 공간의 정보가 필요 없을 뿐이다. 아이템 집합을 만들어 내고, 그 문맥을 이해하고, 모르는 아이템 집합에 대해 대처할 수 있다는 점에서 언어 모델을 빈발 집합 마이닝에 적용해보는 것은 의미가 있는 작업이다.

참고문헌

[1] Ritu Garg and Preeti Gulia, "Comparative Study of Frequent Itemset Mining Algorithms Apriori and FP Growth", *International Journal of Computer Applications, 126*, 8-12, 2015.

[2] O. Barkan and N. Koenigstein, "ITEM2VEC: Neural item embedding for collaborative filtering," *2016 IEEE 26th International Workshop on Machine Learning for Signal Processing (MLSP),* 1-6, 2016.

[3] Peters, Matthew E., Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee and Luke Zettlemoyer. "Deep contextualized word representations." *NAACL-HLT*, 2018.

[4] Devlin, Jacob, Ming-Wei Chang, Kenton Lee and Kristina Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding", *ArXiv, abs/1810.04805*, 2019.

[5] F. Maxwell Harper and Joseph A. Konstan, "The MovieLens Datasets: History and Context", *ACM Transactions on Interactive Intelligent Systems (TiiS), 5, 4: 19:1-19:19. https://doi.org/10.1145/2827872*, 2015.

[6] Yoshua Bengio, Holger Schwenk, Jean-Sébastien Senécal, Fréderic Morin and Jean-Luc Gauvain, "Neural Probabilistic Language Models", *Journal of machine learning research 3, no. Feb*, 1137-1155, 2003.

[7] Tensorflow, "Text generation with an RNN", https://www.tensorflow.org/tutorials/text/text_generation

[8] Ashish Vaswani and Noam Shazeer and Niki Parmar and Jakob Uszkoreit and Llion Jones and Aidan N. Gomez and Lukasz Kaiser and Illia Polosukhin, "Attention is All You Need", *Advances in neural information processing systems 2017*, 5998-6008, 2017.

[9] Moens, Sandy, Emin Aksehirli and Bart Goethals. "Frequent Itemset Mining for Big Data." *2013 IEEE International Conference on Big Data,* 111-118, 2013.

[10] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li and Peter J. Liu. "Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer", 2019.