



고려대학교

딥러닝 논문리뷰

선택 논문명

- Deep Learning-based Online Alternative Product Recommendation at Scale

학번 : 2020511038

이름 : 조송현

Deep Learning-based Online Alternative Product Recommendations at Scale

**Mingming Guo, Nian Yan, Xiquan Cui, San He Wu, Unaiza Ahsan,
Rebecca West and Khalifeh Al Jadda**

The Home Depot, Atlanta, GA, USA

{mingming_guo, nian_yan, xiquan_cui, san_h_wu, unaiza_ahsan,
rebecca_west, khalifeh_al_jadda}@homedepot.com

딥러닝 기반 온라인 대체 상품에 대한 (충분한 규모의) 추천 알고리즘

Published by 미국 홈디포社 (미국의 대형 소매유통체인)

귀밍밍 外

2021.04

개요 (Abstract 부분)

- **Alternative Recommender Systems** 는 E-commerce 업체들에게 중요 (소비자들이 massive product catalog 를 탐색할 때 가이드를 주고, 소비자들이 여러 선택지 중 적합한 상품을 찾는 데 도움을 줌)
- 하지만 소비자에게 맞는(fit) 상품을 추천하는 것은 Non-trivial 한 문제.
- 이 논문에서는 상품의 텍스트 정보(상품명, 상품상세 등) 와 소비자 행동(behavior) 데이터를 이용하여 **alternative products** 추천을 제안한다.
- 결과적으로, precision, recall 뿐 아니라 offline evaluation 또한 상당히 개선되었음 (final A/B test는 논문의 알고리즘이 통계적으로 유의한 방법으로 conversion rate을 무려 12%나 증가시킴)
- Product 정보의 Semantic meaning을 더 정확하게 잡아내기 위해서 **product embedding**을 학습을 위한 방법으로 **Siamese Network with Bidirectional LSTM**(양방향 LSTM를 결합한 삼 네트워크)을 설계하였다.
- 실제 고객이 선호취향을 더 매칭할 수 있는 Similarity space를 학습하기 위해, 논문에서는 실제 고객의 과거행동 비교 데이터를 학습을 위한 label로 사용하였다.
- 더불어, 컴퓨팅 측면에서 비용이 큰 kNN 계산을 빠르게 하기 위한 NMSLIB 을 사용하였다.

1. Introduction

Alternative Product list 를 얻는 방법은 크게 두가지로 볼 수 있는데

1. 상품 텍스트 요소에 기반한 방법 (content-based recommendation approach)

- 만약 2개 상품이 비슷한 속성 또는 요소를 가지고 있다면 1개 상품은 나머지 1개에 의해 replaced 될 수 있음. Item Embedding을 위해 전통적으로는 word2vec을 사용하여 Item간 유사성을 비교하였으나, 이는 Embedding Distance가 실제 고객의 쇼핑 선호도와 일치함을 보장해주지 못함.

2. 고객 행동(customer behavior)을 이용하는 방법

Alternative product 의 item-to-item collaborative filtering 하는 형태. 만약 고객들이 2개 상품을 함께 고려하는 빈도가 높다면 1개 상품은 다른 1개의 alternative가 될 수 있다는 것. 하지만 이러한 접근법은 cold start problem을 가짐.

※ 추천 시스템에는 기존 사용자의 정보를 기반으로 비슷한 성향의 사용자 평점을 예측하는 협업 필터링 방식이 있다. 사용자의 성향을 알기 위해 구매 이력과 같은 정보가 필요한데 이 정보가 없을 때 예측이 어려워지는데 이를 콜드 스타트 문제(cold-start problem)라 한다.

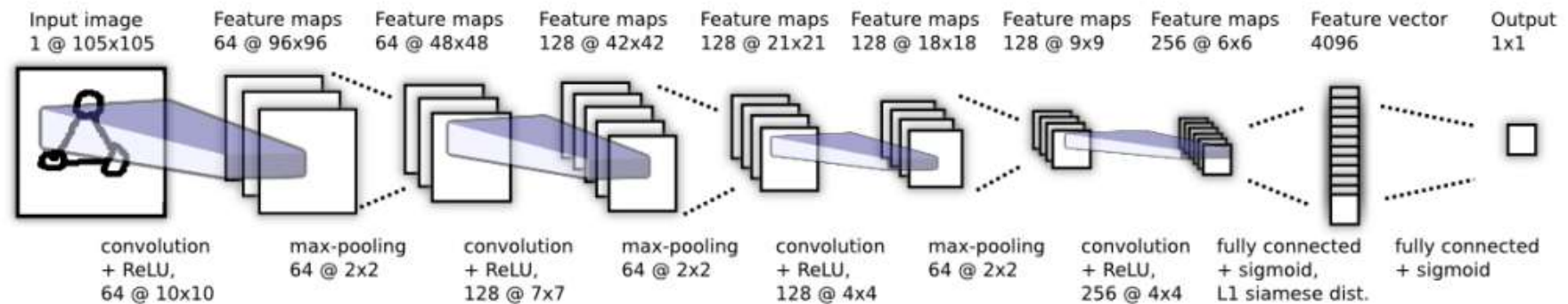
1. Introduction

논문에서 저자는 Supervised Product Embedding Learning process로 추천문제를 표현하였는데, 정확히 얘기하면 Siamese Network (쌍 네트워크)을 이용, 즉, 2개 상품의 요소(content)들과 고객 행동(behavior)을 이용하여 Top-N개 추천상품을 생성하는 방식의 Deep Learning 기반 embedding 접근법을 사용함.

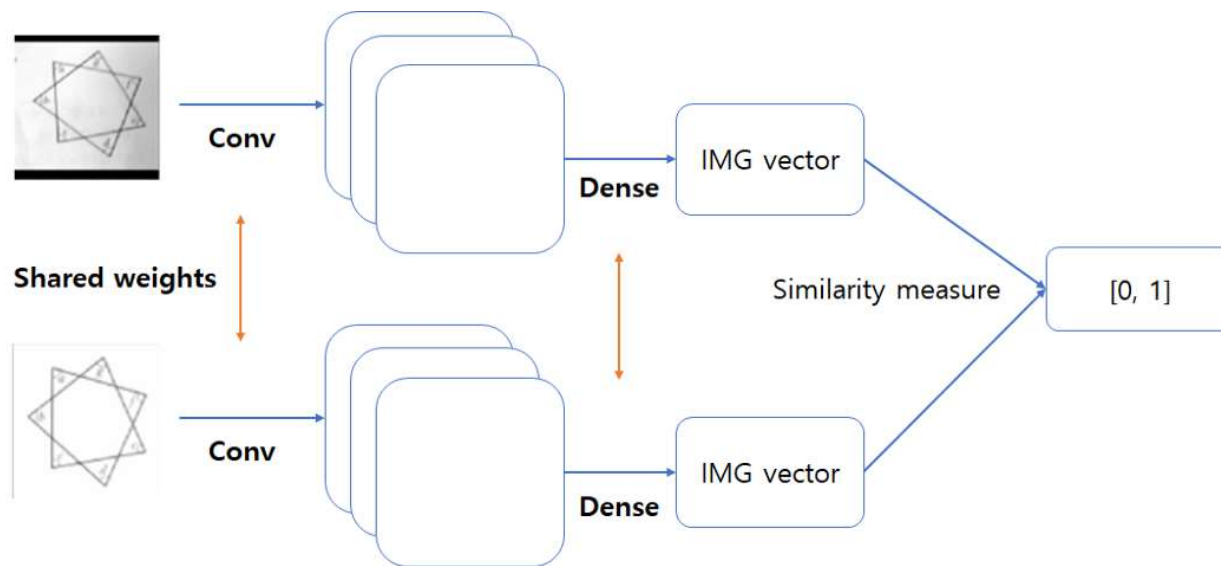
- Recommend alternative products using both product textual information and customer behavior data. This allows us to better handle both the cold start and relevancy problems. => **cold start 문제와 relevancy 문제를 동시에 해결**
- Use a Bidirectional LSTM structure to better capture the semantic meaning of product textual information. => **상품의 텍스트 정보 의미를 더 잘 잡아내기 위해 양방향 LSTM 방식을 이용.**
- Build a Siamese Network to incorporate co-compared customer behavior data to guide the supervised learning process and generate a product embedding space that better matches customer's preference. => **co-compared 고객행동 데이터를 지도학습 프로세스에 결합하기 위해 쌍 네트워크를 설계, 고객 선호를 더 잘 반영하는 상품 임베딩 스페이스를 생성함.**
- Our model outperforms baselines in both offline validations and an online A/B test. => **그 결과 offline 검증과 online A/B test 모두에서 Baseline을 상회하였음.**

※ 삼 네트워크에 대해

- Siamese-Network
- 삼네트워크는 두 사진을 입력으로 받아서 두 이미지를 벡터화 시킨 이후, **두 벡터간의 유사도 (similarity in [0, 1]) 를 반환하는 네트워크**입니다. Network는 해당 이미지의 특징을 hand-crafted features가 아닌 data에서 직접 학습할 수 있으므로 **주어진 similarity 를 최적화 할 수 있는 양질의 feature**를 추출해줍니다.
- 삼 네트워크는 하나의 이미지를 하나의 벡터로 변환할 수 있는 weight를 가지고 있습니다. 아래 그림과 같이 이미지를 입력으로 받아 convolution 연산을 거쳐 하나의 벡터로 이미지를 인코딩합니다.



※ 삼 네트워크에 대해



이렇게 정의된 네트워크에 두 사진이 같을 경우 유사도(Similarity)를 1로 주고, 두 사진이 다를 경우 유사도(similarity)를 0으로 주어서 모델을 학습시킵니다. 이때 학습에 사용하는 loss는 다음과 같이 cross-entropy를 사용합니다.

이를 통해서 추출된 벡터간의 거리는 서로 유사한(=같은) 이미지끼리는 가까운 거리(높은 유사도)를 가지고, 서로 다른 이미지 간에는 먼 거리를 (낮은 유사도) 가지도록 학습이 진행됩니다. 이와 같은 방법을 **similarity learning** 혹은 **metric learning**이라고 부릅니다.

※ 양방향 LSTM에 대해

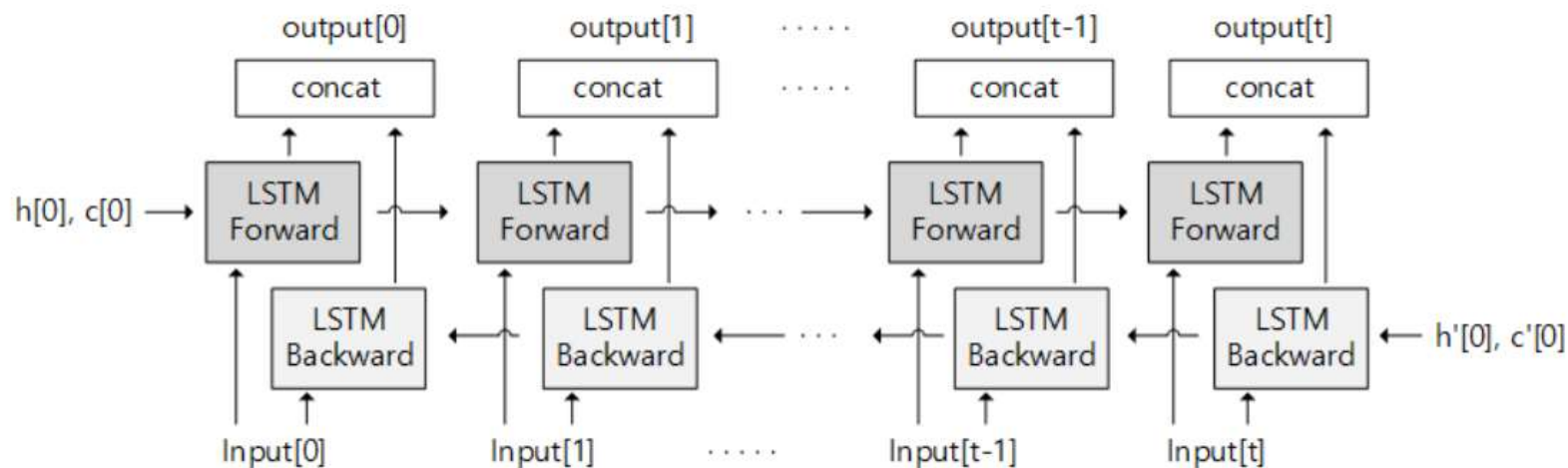
LSTM은 시퀀스 데이터에서 다음을 예측하는데 활용하는데, Bidirectional LSTM은 이전데이터 뿐 아니라 다음 데이터를 활용하여 이전에 뭐가 나올지를 예측.

예를 들어 "나는 _" 이라는 상태에서 다음에 뭐가 나올 지 예측하기란 어렵지만

"나는 _ 먹었다" 와 같은 문장 있다면 목적어에 맞는 음식이 나온다는 것을 예측

즉, **Bidirectional LSTM 모델은 앞에서 뒤로 한방향, 뒤에서 앞으로 한방향 양방향으로 학습하여 모델의 성능을 높인다.**

출력은 양방향의 값을 concatenate하여 word vector를 출력한다



2. Problem Formulation

- Anchor product(고객이 본 상품, $f_w(x_a)$)과 Alternative product(대체상품, $f_w(x_r)$)의 임베딩 간 **cosine similarity**를 **Energy function**으로 오른쪽 수식과 같이 사용.
- Positive class(Alternative 관계)에 대한 $L_+(y(i) = 1)$, Negative class(Alternative가 아닌 관계)에 대한 $L_-(y(i) = 0)$ 을 결합한 Loss Function
- Loss function은 오른쪽과 수식과 같게 됨.

$$E_w = \frac{\langle f_w(x_a), f_w(x_r) \rangle}{\|f_w(x_a)\| \|f_w(x_r)\|} \quad (1)$$

$$L_w^{(i)} = y^{(i)} L_+(x_a^{(i)}, x_r^{(i)}) + (1 - y^{(i)}) L_-(x_a^{(i)}, x_r^{(i)}) \quad (3)$$

The loss functions for the positive and negative cases are given by:

$$L_+(x_a^{(i)}, x_r^{(i)}) = |1 - E_w| \quad (4)$$

$$L_-(x_a^{(i)}, x_r^{(i)}) = \begin{cases} |E_w| & \text{if } E_w > 0 \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

3. Deep Learning Embedding Approach

Textual Data and Co-compared Data

고객은 상품 특징에 근거하여 상품 간 얼마나 유사하거나 또는 다른 지 비교 확인을 위해 검색결과페이지에서 여러가지 상품을 선택 할 수 있음. 이들은 서로에 alternative product인데,. 이러한 co-compared는 상품간 동일한 텍소노미(상품분류)내의 유사성에 대한 강력한 시그널이기도 함. 논문에서는 clickstream으로부터 co-compared 데이터를 추출하여 학습데이터로 사용함.

<i>Product ID</i>	<i>Product Title</i>	<i>Product Description</i>
'12345678'	60 Gal. Electric Air Compressor	This compressor offers a solid cast iron, twin cylinder compressor pump for extreme durability. It also offers 135 psi maximum pressure and air delivery 11.5/10.2 SCFM at 40/90 psi.

Table 1. Product Textual Data

<i>Product ID_1</i>	<i>Product ID_2</i>	<i>Co-compared</i>
'12345678'	'87654321'	1
'32187654'	'54321876'	1

Table 2. Co-compared Example

3. Deep Learning Embedding Approach

- 모든 상품들에 대한 임베딩을 생성하기 위해 양방향 LSTM과 삼 네트워크 components를 만들었음.
- 결과, 상품 임베딩 스페이스는 상품 텍스트 정보들의 의미와 고객 선호를 더욱 잘 반영하였음.
- 텍스트 데이터는 Sequential 형태로 이뤄져 있고, 텍스트의 순서는 네트워크에 중요한데, 논문에서는 input sequence의 양 방향으로 표현을 학습하기 위해 Bidirectional LSTM을 사용함
- 네트워크 설계를 위해 Keras와 Tensorflow를 사용함.
- Optimizer로는 RMSprop을 사용, Loss function은 Binary Cross Entropy 사용.

Siamese Network with Bidirectional LSTM

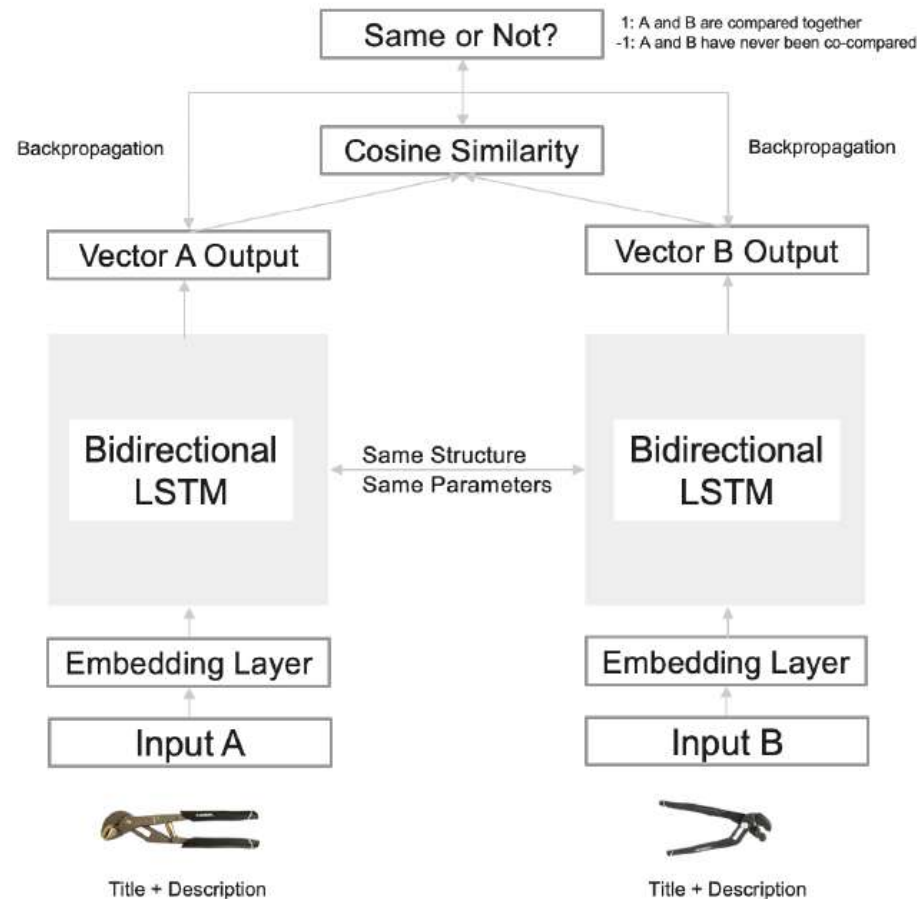


Fig. 1 Siamese Network with Bidirectional LSTM

3. Deep Learning Embedding Approach

Creating Training Data by Sampling

- Positive and Negative Sampling

- A와 B가 co-compared, B와 C가 co-compared이면 (A,B,C)가 connected graph를 형성. D와 E가 co-compared, E와 F가 co-compared이면 (D,E,F)가 connected graph를 형성.
- 논문에서는 각각의 상품에 대해 **same set내** 랜덤 샘플링을 통해 **positive sample**을 생성, 예를들면 [A,C,1]
- 각각의 상품에 대해 서로 다른 **connected graph**를 먼저 샘플링하고 그 그래프에서 1개 상품을 샘플링하여 **negative sample**을 생성, 예를들면 [A,D,-1]
- -1은 negative label인데 최종적으로 0으로 replace 해줌
- negative sampling 공간이 훨씬 큰데, 이는 매우 적은 수의 상품들이 co-compared되기 때문. (positive sample 1개당 negative sample 3개 꼴)
- 이렇게 하면 모델 학습에서의 **validation loss**가 최소화됨.

Creating Training Data by Sampling

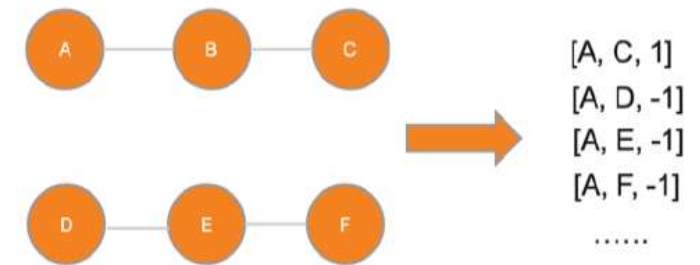


Fig. 2 Connected Graphs

<i>Number of training data records</i>	<i>Number of products involved</i>
331900	65684

Table 3. Training Data Statistics

3. Deep Learning Embedding Approach

Training the Model and Generating (모델 트레이닝 및 생성 과정)

- 삼 네트워크 트레이닝 과정은 10시간 정도 소요되었음
- 임베딩 레이어와 양방향 LSTM 레이어만을 사용하였음.
- 결과적으로 상품 제목의 Hidden State와 상품 상세의 Hidden State를 Concatenation함.
- 논문에서는 상품 제목과 상세 데이터에 기반한 수백만개 임베딩을 생성하였음.
- 각 상품에 대해 similarity metric(cosine similarity)을 이용하여 다른 수백만개 상품들과의 distance를 계산하고, Top N 추천 상품들을 얻기 위해 similarity scores를 상위부터 하위까지 랭크.
- heavy한 계산을 위해 성능이 우수한 NMSLIB* 라이브러리를 선택하였음

* NMSLIB(Non-Metric Space Library). 유사아이템 찾기를 위한 AkNN(Approximate kNN) 패키지 중 하나. 속도 및 정확도에서 Annoy를 압도하는 현재로서 가장 좋은 알고리즘으로 평가.



4 Performance Evaluation

- offline evaluation 과 online A/B test를 이용.

1) Baseline 1 . Attribute Based

- 상품 속성을 이용하여 추천 생성. numerical 그리고 categorical 속성들을 포함. 카테고리 속성들은 원핫인코딩을 통해 numerical format으로 변환. 두 상품간 거리는 cosine similarity를 이용하여 계산됨. 즉, content-based방법.

2) Baseline 2 . Frequently Compared

- 이 기준 알고리즘은 실제 고객이 co-compared 한 데이터를 사용. co-comparison 횟수로 rank된 것을 추천. 다만, cold start 문제때문에 다수의 상품들이 추천을 받지 못함. (고객의 상품 브라우징 행동에 의해 만들어지는 item-to-item 관계에 기반한 것이므로) 즉, collaborative filtering 방법.

3) Proposed . 딥러닝 기반

- cosine similarity 의 cutting threshold 를 0.8로 설정. 이러한 한계값 설정은 인간 전문가 (검증자)가 수천개 랜덤 샘플들을 시험한 결과 판단을 한 것에 기반.
- 논문에서는 최소 0.8 similarity 를 갖는 추천만을 선정

4 Performance Evaluation

1) Precision and Recall

비교 1 :

평가에는 2주간의 실제 고객 구매 데이터를 사용. (추천상품과 구매 상품을 비교)

우측 테이블4를 보면 논문 알고리즘이 baseline 1 알고리즘보다 훨씬 나은 성능 보였음을 보여준다.

이유: Frequently compared는 매우 적은 수의 co-compared 상품 sets만을 커버하므로.

Attributes based 는 커버리지는 넓지만 relevancy가 낮음.

Offline Evaluation:

1) Precision and Recall:

	Precision			Recall		
	Top 1	Top 5	Top 10	Top 1	Top 5	Top 10
Attribute Based	0.23%	0.13%	0.10%	0.15%	0.34%	0.33%
Frequently Compared	0.75%	0.51%	0.47%	0.51%	0.93%	1.02%
Deep Learning Based	1.47%	0.81%	0.61%	0.91%	2.08%	2.59%

Table 4. Precision and Recall with Raw Sessions

4 Performance Evaluation

1) Precision and Recall

비교 2 :

딥러닝 기반이 attribute based보다는 훨씬 높지만 baseline 2(**frequently compared**) 보다는 그렇지 않은데, 이유는 논문에서 모델 학습시 사용한 label이 **co-compared data**로부터 나온 것이기 때문.

즉, 모델은 frequently compared의 성능에 의해 upbound 가짐.

Comparison 2:

	Precision			Recall		
	Top 1	Top 5	Top 10	Top 1	Top 5	Top 10
Attribute Based	0.21%	0.13%	0.12%	0.16%	0.31%	0.34%
Frequently Compared	2.48%	1.81%	1.75%	1.65%	2.65%	2.76%
Deep Learning Based	1.72%	0.90%	0.67%	1.09%	2.33%	2.85%

Table 5. Precision and Recall with Filtered Sessions

4 Performance Evaluation

2) Coverage

- 속성 기반에 비해 딥러닝 기반 어프로치는 anchor coverage를 49.7% 증가시킴.
- frequently compared와 비교하여 36.3% 증가시킴.
- 대부분 상품들은 제목과 상세설명이 있으므로 논문의 딥러닝 기반 접근법은 good recommendation이 가능하도록 anchor products들의 커버리지를 상당히 개선시킴.

4 Performance Evaluation

Online A/B Testing

- **conversion rate**
- A/B Test는 3주간 진행되었고 성공여부로 전환율 측정
- Conversion rate(전환율)이란 구매횟수를 방문수로 나눈것.
- **논문의 딥러닝모델 attribute와 frequently를 결합한 하이브리드 알고리즘보다 나은 결과를 보였음 (전환율 12% 더 높았음). 즉, 매우 성공적.**

5. Related Work

- 추천시스템을 위한 전통적인 방법 은 content-based 인데 이방법은 cold-start문제를 잘 해결할 수 있음.
- Collaborative Filtering은 사용자행동에 기반한 다른 방법인데 예를 들어 Matrix Factorization 같은 것이 널리 사용됨
- 논문의 두 개 baseline 알고리즘 보면 하나는 content based이고 다른 하나는 고객행동기반의 collaborative filtering임.
- 딥러닝 방식은 현재 학문적 영역 말고도 산업 영역에서 추천시스템 위해 널리 사용됨. 에어비앤비의 추천시스템과 핀터레스트의 추천엔진이 그 예.
- 최근대부분 딥러닝 연구는 개인화 추천에 초점 맞춰져 있는데, 양방향 LSTM을 더한 삼 아키텍처를 활용 딥 네트워크 논문이 기존에도 존재하지만, **이 논문에서는 human annotated label이 아닌 고객의 click stream으로부터 추출한 labels를 사용한 것이 차별점.**
- 텍스트데이터와 고객 시그널로부터 상품 임베딩을 학습함으로써 **Alternative Recommendation** 제공에 좀 더 초점을 맞춘 작업

결론 및 소감

- 추천 시스템은 온라인 소매업자들에게 매출신장을 위한 핵심 기능이므로 그 중요성은 점차 커질 것임.
- 고객들에게 대체 상품을 자동화된 방법으로 수월하게 제공하기 위해서 논문에서는 deep learning approach to generate product embeddings based on a Siamese Network with Bidirectional LSTM 을 사용하였음.
- 논문에서 고객의 클릭 활동과 상품 텍스트정보로부터 함께 Co-compared data를 추출하여 임베딩 스페이스를 생성하는 과정이 인상적이었음.
- 양방향 LSTM 및 양방향 RNN를 활용하여 Sequential Data를 모델링하고, 삼 네트워크를 활용하여 Alternative Recommendation System을 구성하는 방안에 대해 더 많은 논문을 찾아보며 깊이 공부해야 겠다는 생각을 하게 되었음.

Thank you



고려대학교