https://doi.org/10.15813/kmr.2021.22.4.002

CTR 예측을 위한 비전 트랜스포머 활용에 관한 연구[1)]

A Study on Utilization of Vision Transformer for CTR Prediction

김태석 (Tae-Suk Kim) 배재대학교 경영학과[2)]
김석훈 (Seokhun Kim) 배재대학교 전자상거래학과[3)]
임광혁 (Kwang Hyuk Im) 배재대학교 전자상거래학과[4)]

〈 국문초록 〉

Click-Through Rate(CTR) 예측은 추천시스템에서 후보 항목의 순위를 결정하고 높은 순위의 항목들을 추천하여 고객의 정보 과부하를 줄임과 동시에 판매 촉진을 통한 수익 극대화를 달성할 수 있는 핵심 기능이다. 자연어 처리와 이미지 분류 분야는 심층신경망(deep neural network)의 활용을 통한 괄목한 성장을 하고 있다. 최근 이 분야의 주류를 이루던 모델과 차별화된 어텐션(attention) 메커니즘 기반의 트랜스포머(transformer) 모델이 제안되어 state-of-the-art를 달성하였다. 본 연구에서는 CTR 예측을 위한 트랜스포머 기반 모델의 성능 향상 방안을 제시한다. 자연어와 이미지 데이터와는 다른 이산적(discrete)이며 범주적(categorical)인 CTR 데이터 특성이 모델 성능에 미치는 영향력을 분석하기 위해 임베딩의 일반화(regularization)와 트랜스포머의 정규화(normalization)에 관한 실험을 수행한다. 실험 결과에 따르면, CTR 데이터 입력 처리를 위한 임베딩 과정에서 L2 일반화의 적용과 트랜스포머 모델의 기본 정규화 방법인 레이어 정규화 대신 배치 정규화를 적용할 때 예측 성능이 크게 향상됨을 확인하였다.

주제어: 클릭률, 심층신경망, 추천시스템, e-비즈니스

---

1) 이 논문은 2019년 대한민국 교육부와 한국연구재단의 인문사회분야 신진연구자지원사업의 지원을 받아 수행된 연구임(NRF-2019S1A5A8033018)
2) 제1저자, 교신저자, itmkim@pcu.ac.kr
3) 제2저자, kimshn@pcu.ac.kr
4) 제3저자, khim@pcu.ac.kr

논문접수일 2021년 10월 04일 | 1차 수정 2021년 10월 15일 | 게재확정일 2021년 10월 18일

1. 서론

추천시스템은 수많은 제품 가운데 고객이 선호하는 제품정보를 제공해 줌으로써 고객들의 구매를 촉진하여 판매자에게는 경쟁자 대비 수익성을, 구매자에게는 정보 과부화를 줄여 줄 수 있다. 추천시스템의 핵심은 사용자가 특정 상황에서 노출된 광고나 제품을 클릭할 가능성을 추정하는 Click-Through Rate(CTR) 예측이다. 추천시스템은 예측된 CTR을 기반으로 후보 항목의 순위를 결정하고 높은 순위의 항목들을 추천한다. 온라인 광고 시스템의 경우, 광고 순위 전략은 일반적으로 'CTR×입찰가'에 따라 달라지며, 여기서 입찰가는 광고가 클릭될 때 시스템이 받는 혜택이다. 따라서 정확한 CTR 추정은 기업의 비즈니스 이익에 직결된다.

CTR 예측 문제의 핵심은 예측에 영향을 주는 특징 간 상호작용을 효과적으로 모델링하는 것이다. 자연어 처리 및 이미지 분류 분야에서의 심층신경망(deep neural network, DNN)의 눈부신 성과는 CTR 예측에 DNN을 활용하여 특징 간 고차원 상호작용을 파악하고자 하는 시도를 촉발시켰다(김은미, 2021; 김태석, 2020; 문현실 등, 2020; 원종관, 2021). Deep & Cross Network(DCN)(Wang et al., 2017)은 다층 잔차(residual) 구조를 적용하여 특징의 고차 표현을 학습하였다. xDeepFM(Lian et al., 2018)은 새로운 CIN(Compressed Interaction Network)을 제안하여 특징 상호작용을 모델링하였다. FGCNN(Liu et al., 2019)은 Convolutional Neural Network(CNN)를 사용하여 2차원 공간에서 특징을 추출하여 특징 공간을 확장하였다. FiBiNet(Huang et al., 2019)은 bilinear 함수를 통해 특징 상호작용을 학습하였다. 이러한 연구들은 고차 특징 상호작용을 포착하기 위해 다수의 은닉 계층(hidden layer)을 적층하는 DNN을 활용하고 있다. 그러나 이러한 접근은 두 가지 한계가 있다. 첫째, DNN 기반 모델은 암시적(implictly) 방식으로 특징 상호작용을 학습하기 때문에 어떤 특징 조합이 의미 있는지에 대한 설명력이 부족하다. 둘째, CTR 데이터의 유용한 특징 상호작용은 극히 일부분에 국한되기 때문에 학습에 많은 매개변수를 필요로 하는 DNN은 많은 계산 자원이 필요하여 비효율적이다.

한편, 구글은 2017년 셀프-어텐션(self-attention) 메커니즘을 활용하는 트랜스포머(Transformer)라는 자연어 처리 모델을 발표하였다(Vaswani et al., 2017). 트랜스포머 모델의 등장 이전에는 대부분의 자연어 처리(자연어 생성, 질문-응답, 자연어 번역 등) 모델은 인코더-디코더(encoder-decoder) 구조를 가지는 Recurrent Neural Network(RNN) 기반이었다. RNN 구조의 가장 큰 한계는 구조상 순차적인 계산만이 가능하여 병렬연산이 불가능한 점이었는데 트랜스포머는 행렬곱을 통한 병렬연산을 수행하여 연산 시간과 계산 자원을 저감시켰다. 또한 셀프-어텐션 기능은 입력 문장에서 단어 간의 관계를 정량화하는 기준을 제시하여 단어간 상호작용에 대한 해석을 용이하게 하였는데 이러한 장점으로 인해 트랜스포머는 RNN 기반의 모델들 보다 앞선 성능을 달성하였다. 나아가, 2020년 구글은 이미지 분류를 위해 트랜스포머 구조를 활용한 비전 트랜스포머(Vision Transformer) 모델을 발표하였다(Dosovitskiy et al., 2020). 비전 트랜스포머는 이 분야의 연구 추세를 급격히 바꿀 정도의 큰 파급력을 가져왔다. 그전까지 AlexNet(Alex et al., 2012)이 처음으로 제안한 CNN 구조가 state-of-the-art를 달성한 이후 이 분야의 연구는 CNN 구조를 기반으로 진행되었다. 비전

트랜스포머는 자연어 처리에서 트랜스포머가 단어를 처리하는 방안에서 착안하여 이미지를 패치로 나누고 패치들의 선형 임베딩(embedding) 배열을 트랜스포머의 입력으로 사용하였다. 비전 트랜스포머는 큰 데이터 세트(14M~300M 이미지)에서는 CNN 구조보다 좋은 성능을 달성함을 보였다. 충분한 크기의 데이터 세트와 사전 학습(pre-train)이 필요하다는 단점에도 불구하고, 초기 연구 성과로 큰 잠재력을 입증한 만큼 이를 개선할 많은 후속 연구들이 파생될 것으로 예상된다.

상기에서 언급한 것과 같이 DNN 기반 모델은 CTR 예측을 위한 계산 효율성과 예측 효과성에서 명확한 한계를 가진다. 트랜스포머 구조는 입력 요소 간 관계 정량화를 통한 해석의 용이성과 연산 병렬화로 인한 계산 효율성에서 강점을 갖는데 이는 특성상 입력 데이터의 수가 거대하고 특징 간 고차원 상호작용 파악이 핵심인 CTR 예측에 있어서 유리한 점이다. 본 연구에서는 CTR 예측을 위해 비전 트랜스포머를 활용함에 있어 고려해야 할 주요 설계 이슈를 도출한다. 구체적으로, 비전 트랜스포머의 입력처리를 위한 임베딩 과정에서의 일반화(regularization)와 비전 트랜스포머 내부의 정규화(normalization)의 효과를 예측 성능 관점에서 분석한다. 본 연구는 CTR 예측을 위해 비전 트랜스포머를 활용한 첫 연구라는 점에서 의의가 있다.

이 논문의 나머지 부분은 다음과 같이 구성된다. 2장에서는 어텐션을 기반으로 하는 모델인 트랜스포머와 비전 트랜스포머를 소개한다. 3장에서는 분석의 주요 대상인 임베딩과 일반화, 그리고 정규화에 대한 개념을 소개한다. 4장에서는 분석 실험을 수행하여 효과적인 비전 트랜스포머 설계에 대한 주요 시사점을 도출한다. 마지막으로, 결론과 추후 연구 방향을 5장에서 논의한다.

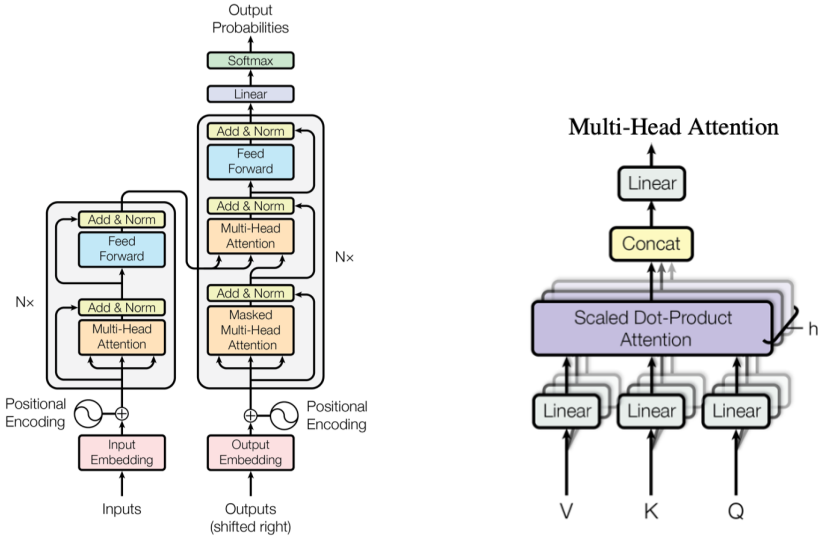
2. 어텐션 기반 모델

2.1. 트랜스포머 모델

자연어 처리에 주로 많이 활용된 모델은 sequence-to-sequence(seq2seq)이다(Sutskever et al., 2014). seq2seq 모델은 인코더와 디코더로 구성된다. 인코더는 입력된 문장으로부터 컨텍스트(context) 벡터를 만드는데, 컨텍스트 벡터는 입력 문장에 대한 정보를 압축하고 있는 벡터이다. 디코더는 전달받은 컨텍스트 벡터의 정보를 활용하여 다른 언어로 번역된 문장을 생성한다. Seq2seq 모델의 한계는 인코딩 과정에서 문장의 길이와 관계없이 고정된 길이의 벡터를 생성하기 때문에 입력 문장의 정보가 일부 손실될 수 있고, 트레이닝 세트보다 테스트 세트의 문장이 더 길 때 성능이 떨어지는 문제가 발생하는 점이다.

Seq2seq의 이러한 근원적인 문제는 2017년 구글이 발표한 트랜스포머 모델을 통해 극복되었다. 트랜스포머는 기존 seq2seq의 구조인 인코더-디코더를 따르면서도 어텐션만으로 구현한 모델이다. 어텐션은 디코더에서 해당 문장을 번역하기 위해 어떤 단어에 더 집중(attention)해야 하는지를 밝혀내는 개념으로 각 단어가 다른 모든 단어와 얼마나 연관되어 있는지를 attention 값으로 정량화한다. 이 모델은 발표 당시 자연어 처리에서 가장 좋은 결과를 달성하고 있던 RNN보다 우수한 성능을 달성하였다.

<그림 1>의 트랜스포머 모델 구조에서 왼쪽 모듈은 인코더이고 오른쪽 모듈은 디코더이다. 인코더와 디코더는 모두 여러 번 쌓을 수 있게 구성되며, 이는 그림에서 'Nx'로 표현된다. 각 모듈은 주로 멀티-헤드 어텐션(multi-head attention) 및 피드 포워드(feed forward) 레이어로 구성되어 있다. 그림 하단의 입력(inputs)과 출력 문자열(outputs)은 임베딩 과정을 통해 n차원 벡



〈그림 1〉 트랜스포머 구조(좌) 및 멀티-헤드 어텐션 프로세스(우)(Vaswani et al., 2017)

터로 변환된다. RNN처럼 문자열이 입력되는 순서를 기억할 수 있는 순환 네트워크가 없으므로 전체 입력을 이루는 각 단어의 순서에 따라 상대적인 위치를 지정하는 위치 임코딩(positional encoding)을 통해 위치 정보가 임베딩 결과 벡터에 추가된다.

임베딩 결과 벡터는 셀프-어텐션을 수행하는 멀티-헤드 어텐션 모듈에 입력된다(<그림 1>의 우). 셀프-어텐션은 입력된 각 벡터에 대해 Query(Q), Key(K), Value(V) 벡터를 생성하는데 이 벡터들은 입력 벡터에 대해서 세 개의 학습 가능한 행렬들 (즉, 학습 파라미터)를 각각 곱함으로써 만들어진다. 그런 후, 생성된 벡터들로부터 다음의 식을 통해 어텐션 스코어를 계산한다.

$$a = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)$$

어텐션 스코어는 문장의 각 단어(Q)가 문장의 다른 모든 단어(K)에 의해 어떻게 영향을 받는지를 정량화한 것이다. Q와 K벡터 내적을 통해 단어간 유사도를 구하고 이를 K 벡터의 차원(d_k)으로 스케일링한다. 스코어는 최종적으로 소프트맥스(softmax) 함수를 통해 0과 1 사이의 확률값으로 계산된다. 마지막으로는 계산된 어텐션 스코어를 V에 곱하여 다른 단어와의 유사도를 원래 입력값에 반영한다. 최종적인 셀프-어텐션의 식은 다음과 같다.

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

나아가, 트랜스포머는 여러 번의 어텐션을 병렬로 사용하는 멀티-헤드 어텐션을 도입하여 성능을 더욱 향상시킨다. 도입할 헤드의 수를 h라고 하면 입력 벡터의 차원을 h로 나눈 차원을 가지는 Q, K, V에 대해 h개의 병렬 어텐션을 수행한다. 이러한 병렬 어텐션은 헤드마다 다른 시각으로 단어 간의 중요도를 파악하게 하여 정보의 표현 능력을 향상시키는 효과를 얻는다.

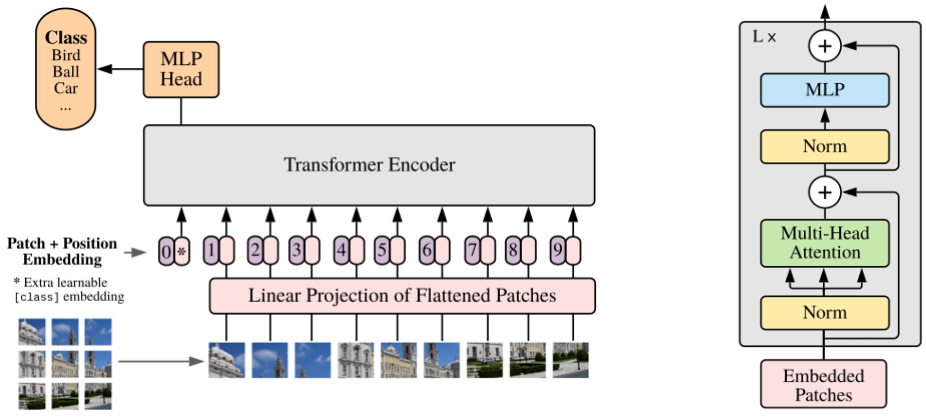
전체적인 트랜스포머의 동작은 다음과 같다. 인코더가 입력 시퀀스를 처리하면 어텐션 벡터인 K와 V가 출력물로 생성된다. 이 벡터들은 디코더에 전달되어 각 디코더의 어텐션 레이어 모듈이 디코더 입력 문장

의 적절한 위치에 집중할 수 있도록 도와준다. 디코더의 입력은 인코더의 입력 과정과 동일하게 임베딩을 한 후 위치 인코딩을 추가하여 디코더에게 각 단어의 위치정보를 더해준다. 디코더에서 각 스텝마다 출력된 단어는 다음 스텝의 디코더 입력에 들어가고 인코더와 마찬가지로 여러 개의 디코더를 거쳐 올라간다. 디코더를 거친 최종 결과는 디코더 모듈 위에 연결된 선형 계층(linear layer)과 소프트맥스 계층(softmax layer)을 통과하여 가장 높은 확률값을 가지는 단어가 해당 스텝의 최종 결과물로서 출력되게 된다.

2.2. 비전 트랜스포머 모델

어텐션을 이용한 트랜스포머는 자연어 처리에서 매우 좋은 성능을 달성하였지만 이미지 처리 분야에서는 그렇지 못했는데 이는 CNN이라는 기존 이미지 처리 분야의 de facto standard의 성능이 우수했으며 트랜스포머는 이미지 처리에 적합한 inductive bias가 부족하다는 인식 때문이었다. Inductive bias는 학습 모델이 지금까지 만나보지 못했던 상황에서 정확한 예측을 하기 위해 사용하는 추가적인 가정을 의미한다. CNN은 컨볼루션(convolution) 연산을 하는 계층을 적층하여 동작하는 신경망으로 이미지 분류를 위해서는 이미지 전체를 한 번에 처리하기보다 로컬 영역으로 범위를 좁혀 공간 정보를 추출하는 것이 성능을 위해 좋다는 가정에서 컨볼루션 연산이 설계된 것이다. 트랜스포머는 컨볼루션 연산과 같은 locality에 대한 가정이 없으므로 inductive bias가 부족하다.

그런데 대규모 데이터 세트에서 트랜스포머를 학습시키면 이러한 제약을 극복할 수 있음을 발견하였고 그 결과로 제안된 모델이 비전 트랜스포머이다. 비전 트랜스포머는 기존 트랜스포머 모델의 구조를 거의 그대로 사용하지만, 인코더-디코더 구조인 트랜스포머와는 달리 인코더만으로 구성된다. 비전 트랜스포머에서 단어가 아닌 이미지를 입력하기 위해 비전 트랜스포머 구조(<그림 2>의 좌)에서 볼 수 있듯이 이미지를 패치로 나눠서 각 패치를 단어처럼 취급한다. 각 패치는 2차원의 형태이기 때문에 단어처럼 다루기 위해 패치를 1차원 벡터로 Flatten하여 인코더에 입력한다. 이때 클래스 정보를 담을 클래스 토큰(class token)과 패치 간의 위치 정보를 학습하는 위치 임베딩이 추가로 인코더에 입력된다. 클래스 토큰은 인코더 모듈을 통해 학습 후 출력 결과가 식별기(MLP Head)에 입력되어 이미지 분류에 대한 최종적인 예측이 나온다. 비전 트랜스포머의 인코더는 거의 트랜스포머의 인코더 부분과 동일하나(<그림 2>의 우) 멀티-헤드 어텐션



〈그림 2〉 비전 트랜스포머 구조(좌) 및 인코더 구조(우)(Dosovitskiy et al., 2020)

과 Multi-layer Perception(MLP) 전에 레이어(layer) 정규화를 수행하는 점이 차이점이다. 또한, MLP는 2단으로 구성된 피드 포워드 네트워크이며 활성화 함수로 트랜스포머에서 채택된 ReLU가 아닌 GELU를 사용하는 점도 차이점이다.

3. CTR 예측을 위한 비전 트랜스포머 설계 이슈

3.1. 임베딩과 일반화

먼저, CTR 예측을 위한 비전 트랜스포머 활용의 첫 단계는 데이터 전처리이다. 원시 CTR 데이터에는 사용자 정보(성별, 직업 등), 컨텍스트 정보(상품 정보, 구매 이력 등) 등의 특징을 포함하는데, 이러한 정보는 대부분 이산적(discrete)이며 범주적(categorical)인 특성을 가진다. 범주적 변수는 일차적으로 원-핫 인코딩(one-hot encoding)을 통해 이진(binary) 특징 집합(벡터)으로 변환되는데, 이로 인해 범주형 변수의 특징 벡터 차원은 매우 높으며 동시에 벡터 대부분의 값이 0으로 희소(sparse)한 특성을 가지게 된다. 이러한 고차원의 희소한 벡터는 계산의 효율성이 떨어지기 때문에 임베딩 과정이 수반된다. 임베딩은 고차원의 희소 특징을 저차원의 잠재(latent) 공간에 삽입함으로써 데이터를 조밀한(dense) 입력 공간에 매핑하여 학습이 효과적으로 수행되도록 전처리하는 과정이다. 이 과정을 통해 하나의 특징은 '1 × 임베딩 크기'의 벡터로 확장되는데 만약 10개의 특징에 대해 크기 20의 임베딩을 적용하면 10 × 20의 행렬이 생성된다. 행렬의 각 요소는 신경망의 학습과 함께 업데이트되며 특징 공학(feature engineering)의 필요를 제거하여 더 높은 예측 정확성을 달성할 수 있게 한다.

원시 데이터에서 특징 i가 가지는 모든 값의 집합을 S_i라 할 경우, 총 n개의 특징 정보로 생성되는 임베딩 벡터의 개수는 $|S_1| \times \ldots \times |S_n|$이며, 임베딩의 크기를 k라고 할 경우 $k \times |S_1| \times \ldots \times |S_n|$의 학습 파라미터가 생성된다. CTR 데이터의 경우 특징이 가질 수 있는 값의 범위가 큰 경우가 많으며 (예를 들어, 제품 ID) 이는 학습 파라미터의 수를 매우 증가시킨다. 일반적으로 신경망의 학습에 있어 학습 파라미터가 증가할 경우, 학습이 오버피팅될 가능성이 높아지는데 특별히 전체 학습 파라미터 중 임베딩과 관련된 파라미터의 비중이 증가할수록 임베딩 처리과정에서 오버피팅을 억제하여 예측 성능 저하를 방지할 방안의 필요성이 대두된다.

일반화는 학습 파라미터가 너무 큰 값들을 갖지 않도록 조정하여 오버피팅을 억제시키는 방법으로 L1과 L2 일반화가 있다(Ng, 2004). L1 일반화는 학습 파라미터를 업데이트할 때 참조하는 손실 함수에 파라미터의 절대값에 비례하는 페널티를 더하는 방식으로, 특징의 임베딩 값이 대부분 0인 모델에서 불필요한 특징에 대응하는 파라미터를 0으로 만들어 해당 특징을 제거하는 효과가 있다. L2 일반화는 손실 함수 파라미터의 제곱값에 비례하는 값을 더하는 방식으로 아주 큰 값이나 작은 값을 가지는 이상치(outlier) 파라미터에 대해 0에 매우 가까운 값으로 만들어 일반화 능력을 개선시키는 효과가 있다. 일반적으로 패널티에 대한 효과는 L2가 L1보다 효과적이라고 알려져 있다(Cortes et al., 2012). 본 연구에서는 임베딩 레이어에서 L2 일반화를 채택하는 것이 비전 트랜스포머의 예측 성능 향상에 효과적인지에 관해 연구를 수행한다.

3.2. 입력 데이터 정규화

앞 절에서 원시 데이터의 특징값이 가지는 값의 범위가 특징에 따라 다를 수 있음을 언급하였다. 특징

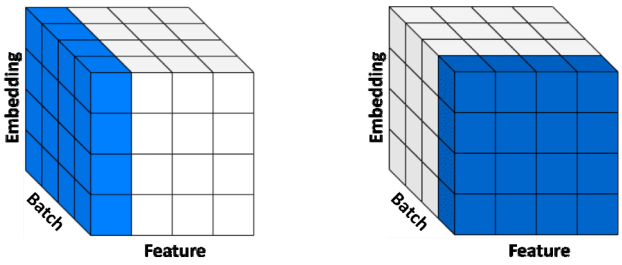
간 값의 범위가 차이가 클 때 손실 함수의 모양은 값의 범위가 큰 특징 축으로는 넓게, 값의 범위가 작은 축으로는 좁게 형성되어 전체적으로는 불규칙한 모형(특징이 2개 존재한다면 한쪽으로는 길고 한쪽으로는 짧은 럭비공 모양)이 형성된다. 신경망의 학습은 손실 함수 상에서 손실을 최소화하는 경사 하강법(Gradient Descent)을 통해 수행되는데 특징값 범위에 차이가 크게 날 때 특징별로 업데이트되는 gradient의 차이가 존재하게 된다. 다시 말해, 어떤 특징 축으로는 작게, 다른 축에 대해서는 크게 업데이트가 되는 비대칭 궤적을 가지게 되어 손실 함수의 최소값을 찾아가는 데 많은 시간이 소요될 수 있다. 이 문제는 손실 함수의 모양을 비교적 규칙적인 형태로 만들어 최소값을 찾는 경사 하강법을 빠르게 진행하게 함으로 해결할 수 있는데 이를 위해 입력 데이터의 분포를 0의 평균과 단위 분산으로 조정하는 과정이 정규화이다. 정규화는 모델에 주입되는 데이터들을 균일하게 만드는 모든 방법이라고 할 수 있다.

신경망에서 주로 사용되는 정규화 방법은 배치(batch) 정규화와 레이어(layer) 정규화이다(Huang et al., 2020). 일반적으로, 배치 정규화는 이미지 분류 분야에서 널리 채택되고 있으며 레이어 정규화는 자연어 처리에서 사용되는 정규화 방식이다. <그림 3>은 CTR 데이터에 대한 배치 및 레이어 정규화 적용을 도식화한 것이다. 3차원으로 표현된 비전 트랜스포머의 입력 데이터는 (특징, 임베딩, 배치)의 3차원 배열을 갖는다. 배치는 학습 데이터를 업데이트하는데 사용되는 데이터 묶음의 크기이다. 그림에서 색칠된 부분은 정규화가 진행되는 단위이다. 배치 정규화의 단위는 특징의 개수만큼 존재하며 개별 특징에 대해 (배치, 임베딩)의 2차원 형태로 표현될 수 있다(<그림 3>의 좌). 임베딩 k에 대해서 배치 크기에 해당하는 수의 데이터가 존재하고 이로부터 μ_k와 σ_k^2를 계산하면 j번째 배치, k번째 임베딩에 해당하는 데이터 x_{jk}에 대한 정규화는 다음의 식으로 표현된다.

$$\hat{x}_{jk} = \gamma_k \frac{x_{jk} - \mu_k}{\sqrt{\sigma_k^2 + \epsilon}} + \beta_k$$

식에서 μ_k, σ_k^2는 데이터로부터 얻은 평균과 분산이며, γ_k와 β_k는 각각 scaling, shift 파라미터이고 ϵ은 수치적 불안정성(0으로 나누는 문제)을 피하기 위해 0에 근사한 작은 값을 사용한다. 레이어 정규화는 <그림 3>의 오른쪽 그림에서와 같이 정규화 단위가 배치 별로 수행되는 점이 배치 정규화와의 차이다. 레이어 정규화의 대상은 (특징, 임베딩)을 축으로 하는 2차원의 형태인데 정규화 적용 과정은 배치 정규화와 동일하며 차이점은 배치 정규화에서는 배치에 대한 평균과 표준 편차를 구하였다면 레이어 정규화는 특징에 대해서 평균과 표준 편차를 구한다는 점이다.

CTR 예측 성능 향상을 위한 정규화 방법을 찾기 위해 본 연구에서는 비전 트랜스포머 모듈의 정규화 방



〈그림 3〉 정규화: 배치 정규화(좌) 및 레이어 정규화(우)

법으로 배치와 레이어 정규화를 적용하여 예측 성능을 비교 분석한다.

4. 실험 결과

4.1. 실험 설정

실험에 사용할 데이터는 Avazu 데이터 세트로 Avazu社에서 모바일 광고의 CTR 예측을 위한 알고리즘 경연대회를 위해 2014년에 공개하였다. 데이터 세트는 총 10일간의 클릭 로그에 대한 기록으로 site 관련 정보(id, domain, category), 앱 관련 정보(id, domain, category), 디바이스 정보(id, ip, model, type 등) 등 총 23개의 특징 필드를 담고 있으며 이 중 8개 특징 필드에 대한 정보는 공개되지 않았다. 데이터 샘플 수가 4천만 개 이상으로 거대하고 양의 클릭과 음의 클릭(클릭 하지 않음)의 비율이 1:5로 클래스의 불균형이 심하여 본 실험에서는 비율을 1:1로 조정한 총 250만 개의 데이터를 실험에 사용한다. 실험에 사용될 모델은 구글에서 발표된 비전 트랜스포머 모델(Dosovitskiy et al., 2020)을 사용하며 임베딩 레이어에서 L2 정규화의 적용과 트랜스포머 모델 내부의 정규화 모듈에 기존 정규화 방식인 레이어 정규화 외에 배치 정규화와 정규화 효과 비교를 위한 정규화 미적용 모델을 추가로 구성한다.

실험은 일반화와 정규화 효과를 검증하는 두 가지 실험으로 구성된다. 일반화 효과 검증을 위해서 정규화 미적용, 레이어, 배치 정규화 적용 모델에 대해 트랜스포머의 층을 1, 2, 3, 4, 임베딩 크기를 4, 8, 16, 32로 설정하고 임베딩 레이어의 L2 일반화를 적용한 경우와 적용하지 않은 경우의 성능을 측정한다. 정규화 효과 검증을 위한 실험에서는 트랜스포머의 층을 1, 3, 5, 7, 임베딩 크기를 4, 8, 16, 32, 64, 128로 설정하여 각 층과 임베딩 크기의 조합에서 정규화 미적용, 레이어, 배치 정규화 적용 모델의 성능을 측정한다. 본 실험에서는 정규화를 적용하지 않은 모델을 Base 모델로 명명한다. 두 실험에서 공통적으로 트랜스포머 내부의 MLP와 마지막 MLP Head의 뉴런의 수는 (128, 64), 멀티-헤드 어텐션 모듈 내의 헤드 수는 2, drop out 비율은 0.5, 배치 크기는 2048로 설정한다. 모델 간의 성능 비교를 위한 지표로 거의 모든 CTR 연구 논문에서 활용하는 AUC(Area Under ROC)를 활용한다. AUC는 무작위로 선택한 양의 샘플(클릭 샘플)이 무작위로 선택한 음의 샘플보다 순위가 더 높을 확률을 측정하는 지표이다(song et al., 2019). AUC의 상한은 1이고 AUC가 높을수록 CTR 예측 성능이 향상된다.

4.2. 일반화 효과

임베딩 크기의 증가는 모델 복잡도를 증가시켜 고차원 특징 상호관계 파악에 이바지함과 동시에 오버피팅의 위험도 증가시킨다. 임베딩 크기별 임베딩의 일반화 적용에 대한 AUC 결과를 <표 1>에 정리하였다. 실험 결과에 의하면 임베딩에 대한 L2 일반화는 일반화 미적용 대비 Base 모델의 경우 3.09~3.69%, 레이어 정규화 모델은 2.46~4.39%, 배치 정규화 모델은 2.7~3.4%의 AUC 성능 향상을 달성하였다. L2 일반화 적용에 따른 AUC 성능 향상은 일반화가 오버피팅을 효과적으로 억제하여 테스트 세트에 대한 범용성을 확보한 결과로 해석된다. 특별히 Base 모델의 결과는 정규화 효과가 배제된 일반화의 효과를 나타내는데 임베딩 크기가 4, 8, 16, 32로 증가될 경우 AUC 증분은 3.091%, 3.459%, 3.686%, 3.235%로 증가함을 보였다. 이는 임베딩 크기 증가에 따른 일반화 처리가 효과적임을 나타낸다. 임베딩 크기 32부터는 학습 파라미터 증가로 인한 오버피팅의 영향으로 일반화 효과가 감

〈표 1〉 임베딩 크기에 따른 일반화 성능 비교

정규화			Base		레이어		배치	
일반화			No	L2	No	L2	No	L2
임베딩 크기	4	Mean	70.406%	73.497%	68.801%	73.195%	70.519%	73.613%
		t-stat	-35.2432		-16.5952		-41.5788	
		p-value	0**					
	8	Mean	70.185%	73.644%	69.560%	73.576%	70.533%	73.933%
		t-stat	-48.3824		-7.8918		-25.3369	
		p-value	0**					
	16	Mean	69.955%	73.641%	71.038%	73.798%	70.668%	73.919%
		t-stat	-44.3607		-16.0776		-15.5808	
		p-value	0**					
	32	Mean	70.406%	73.641%	71.288%	73.748%	71.221%	73.926%
		t-stat	-13.2390		-18.9181		-50.8488	
		p-value	0**					

(**: $p < 10^{-4}$(two tailed t-test))

〈표 2〉 정규화 방법과 임베딩 크기에 따른 임베딩 및 트랜스포머 계층 훈련 파라미터 수와 임베딩 계층 파라미터 비율

정규화			Base	레이어	배치
임베딩 크기	4	임베딩	6,040,148	6,040,148	6,040,148
		트랜스포머	325~1009	341~1073	357~1137
		비율(%)	99.983~99.995	99.982~99.994	99.981~99.994
	8	임베딩	12,080,296	12,080,296	12,080,296
		트랜스포머	729~2433	761~2561	793~2689
		비율(%)	99.98%~99.994	99.979~99.994	99.978~99.993
	16	임베딩	24,160,592	24,160,592	24,160,592
		트랜스포머	1921~6817	1985~7073	2049~7329
		비율(%)	99.972~99.992	99.971~99.992	99.97~99.992
	32	임베딩	48,321,184	48,321,184	48,321,184
		트랜스포머	5841~21729	5969~22241	6097~22753
		비율(%)	99.955~99.988	99.954~99.988	99.953~99.987

소하기 시작한 지점으로 해석된다.

측정된 일반화 효과는 0.1%의 AUC 상승도 의미가 있는 CTR 예측 분야임을 참작할 때 성능 향상 정도가 매우 높은 것인데 이는 임베딩과 관련된 학습 파라미터 수의 비중과 관련이 깊은 것으로 해석된다. <표 2>에는 임베딩과 트랜스포머 정규화 방법별 학습 파라미터 개수가 정리되었다. 각 임베딩 크기에서 생성되는 임베딩 모듈의 학습 파라미터 수는 특징들이 가지는 값들의 조합 개수인 1,510,037개에 해당 임베딩 크기를 곱한 수가 된다. 트랜스포머 모듈은 트랜스포머 층의 수(1~4)에 따라 학습 파라미터 개수에 차이가 발생한다. 표의 결과에 따르면, 모든 임베딩 크기와 정규화 방법에 대해 임베딩 모듈의 학습 파라미터는 전체 파라미터에서의 비중이 99% 이상을 차지한다. 이렇게 임베딩 파라미터가 차지하는 비중이 절대적으로 많으면 임베딩 파라미터로 인해 오버피팅에 빠질 가

능성이 매우 높아질 수 있고 실제로, 일반화를 적용하지 않은 모델들은 학습 과정에서 임베딩 크기와 정규화 방법에 상관없이 매우 심한 오버피팅 추세를 보였다.

4.3. 정규화 효과

신경망에서는 층이 깊어지면 기울기 소실(gradient vanishing) 문제가 심화하여 학습이 잘 이루어지지 않는 경향이 있으며, 손실 함수의 표면 모양이 매끄럽지 못한 형태를 가져 local optimum에 빠지기 쉬워진다. 또한 비선형 함수의 연속적인 적용으로 모형의 복잡도가 증가하여 오버피팅의 가능성이 커진다. 정규화는 이러한 문제를 완화하기 위해 적용되는 기법이다. <표 3>은 트랜스포머 층의 깊이에 따른 세 가지 정규화 모델의 AUC 성능을 측정한 결과이다. 표의 결과에서 확인할 수 있듯이 배치와 레이어 정규화는 초기에는 층이 깊어짐에 따라 AUC 성능이 향상하는 경향을 보였다. 층의 깊이에 따른 AUC 증가는 비선형성의 증가로 모델 복잡도의 AUC 성능에 대한 기여도가 증가함을 의미한다. 하지만, 어느 수준 이후부터는 성능이 감소하기 시작하는데 이는 층이 깊어짐에 따라 서두에 전술한 문제들의 심화로 인해 정규화의 효과가 점차 감소하는 것으로 해석되며 배치 정규화와 레이어 정규화 모두 3층 이후 성능 하락이 나타났다. 반면 Base 모델은 층이 깊어짐에 따라 성능이 처음부터 지속적으로 감소하는 것을 확인할 수 있다. 실험에서 Base 모델의 학습 추이는 층이 깊어질수록 검증 손실(validation loss)이 지속적으로 증가하는 오버피팅 현상이 심화함을 보였는데 이것이 예측에 대한 범용성을 떨어뜨려 성능 저하를 가져온 주요 원인으로 해석된다.

배치 정규화는 모든 정규화 테스트 설정에서 가장 좋은 AUC 성능을 얻었으며 전체적으로 레이어 정규화 대비 AUC 성능이 0.238%, Base 모델 대비 0.269% 우수하였다. 일반적으로 배치 정규화는 손실 함수의 표면을 부드럽게 해주는 smoothing 효과를 발생시켜 학습을 빠르게 하며 local optimum에 빠지는 것을 방지해준다고 알려져 있다(Wu et al., 2018). 또한, 배치 단위 정규화의 평균과 분산은 전체 샘플값과 차이를 발생시켜 모델에 노이즈를 추가하는 일종의 일반화 기능을 발생시켜 범용성에 도움을 준다. 아울러, 본 실험에서는 CTR 예측에 있어 배치 정규화를 적용하는 것이 레이어 정규화보다 성능상의 이점이 존재할 수 있는 추가적인 가능성을 확인하였다. 3장에서 기술하였듯이 동일 임베딩 사이즈에 대해 배치와 레이어 정규화의 대상 데이터는 각각 배치 크기와 특징의 개수에 비례한다. 일반적으로 CTR 특징의 개수는 100개 미만인 경우가 대부분(본 실험에서는 22개)이지만 배치 크기는 연산 자원(메모리 등)의 가용 범위 내에서 100 이상의 경우가 흔하게 사용될 수 있다(본 실험에서는 2048을 사용). 이러한 정규화 범위의 차이는 CTR 예측 분야에서 레이어 정규화 대비 배치 정규화가 효과적일 수 있으며 궁극적으로 AUC 성능 차이를 발생시킬 수 있음을 시사한다.

〈표 3〉 트랜스포머 계층 깊이에 따른 정규화 성능 비교

계층 수		1	3	5	7	평균
정규화	배치	73.995%	74.027%	73.908%	73.898%	73.957%*
	레이어	73.700%	73.808%	73.720%	73.648%	73.719%
	Base	73.733%	73.705%	73.665%	73.650%	73.688%

(*: $p < 10^{-2}$(two tailed t-test))

5. 결론

본 논문에서는 CTR 예측에 비전 트랜스포머 모델을 활용함에 있어 데이터 입력처리를 위한 임베딩 과정의 일반화, 트랜스포머의 정규화 과정이 예측 성능에 미치는 영향을 분석하여 CTR 예측을 위한 트랜스포머 활용 시 참고할 수 있는 설계 가이드를 제공한다.

CTR 원시 데이터는 대부분 이산적이며 범주적인 특성을 가지기 때문에 이를 이진 특징 벡터로 변환하면 특징 벡터의 차원이 매우 높으며 동시에 대부분의 값이 0으로 희소한 특성을 가지게 된다. CTR 데이터 특성으로 인해 특징이 갖는 값의 범위는 특징에 따라 편차가 매우 심하고 나아가 이산값에 매핑된 특징값들의 조합 수가 매우 많아지는데 이는 임베딩 과정에서 대규모의 학습 파라미터를 생성시키게 된다. 증가한 파라미터는 오버피팅을 유발하고 예측에 대한 범용성을 저하시켜 궁극적으로 예측 성능의 저하를 가져올 수 있다. 본 논문에서는 CTR 데이터 특성이 트랜스포머의 성능에 미치는 영향을 일반화와 정규화 방법 측면에서 실험을 통해 분석을 수행하였다.

실험 결과에 따르면 임베딩 과정에서 적용한 L2 일반화는 정규화 효과를 배제한 Base 모델의 경우 3% 이상의 AUC 성능 향상 효과가 있었다. 이에 대한 주요 원인은 전체 학습 파라미터 중 임베딩과 관련된 파라미터의 비중이 99% 이상으로 높아 일반화로 인한 오버피팅 저감의 효과가 큰 것으로 분석되었다. 또한, 정규화 실험에서는 배치 정규화가 비전 트랜스포머의 기본 탑재 방식인 레이어 정규화보다 AUC 성능이 우수하였다. 이는 정규화 범위가 배치 정규화의 경우는 배치 크기로, 레이어 정규화의 경우는 특징의 개수에 의해 결정되는데 일반적으로 배치 크기를 특징의 개수보다 크게 설정할 수 있어 정규화의 효과가 더 크기 때문이다.

본 연구는 이미지 분류와 자연어 처리 분야에서 최근 제안되어 가장 좋은 성능을 달성하고 있는 트랜스포머 기반 모델을 CTR 예측을 위해 활용하고자 하는 배경에서 수행되었다. 본 연구는 이미지와 자연어 데이터와는 다른 특성을 가지는 CTR 데이터로 인해 CTR 예측을 위한 비전 트랜스포머 활용 시 차별적으로 고려되어야 할 설계 인사이트를 실험을 통해 처음으로 도출한 연구 결과라는 의의를 지닌다. 연구의 결과물은 사용자의 잠재된 관심사를 정확히 파악하여 구매 전환을 통해 매출을 향상시키고 특정 행동(예: 광고 클릭을 통한 웹사이트 방문 등)을 유도하는 것을 포함한 온라인 광고 및 전자상거래 시스템에서의 성과 창출을 위해 활용될 수 있다.

후속 연구 방향은 다음과 같다. (1) 본 연구는 범주 데이터만을 갖는 데이터 세트에 대해 수행하여 얻은 결과라는 한계를 지닌다. 추후 연구 방향으로 연속적(continuous)인 수치데이터의 특성을 고려한 임베딩 학습 방안을 수립할 예정이다. (2) 본 연구는 특징 간 상호작용 파악을 위해 비전 트랜스포머 인코더에 탑재된 멀티-헤드 어텐션 모듈을 활용하였는데 CTR 데이터 특성을 고려한 어텐션과 MLP 모듈의 최적 구조에 관한 연구가 추가적인 성능 향상을 위해 필요하다. (3) 본 연구의 결과를 포함하여 전술한 후속 연구를 통합한 모델은 클래스의 비율이 불균형한 대규모 데이터 세트에서 기존에 제안된 CTR 예측 모델들과의 성능 비교 실험을 통해 평가를 진행할 예정이다.

〈참고문헌〉

[국내 문헌]

1. 김은미 (2021). 감성분석을 이용한 뉴스정보와 딥러닝 기반의 암호화폐 수익률 변동 예측을 위한 통합모형. **지식경영연구, 22**(2), 19-32.
2. 김태석 (2020). CNN을 활용한 CTR 예측 시각화 분석. **자료분석학회논문지, 22**(6), 2603-2614.
3. 문현실, 임진혁, 김도연, 조윤호 (2020). 시각 정보를 활용한 딥러닝 기반 추천 시스템. **지식경영연구, 21**(3), 27-44.
4. 원종관, 홍태호 (2021). 텍스트 마이닝과 딥러닝을 활용한 암호화폐 가격 예측: 한국과 미국시장 비교. **지식경영연구, 22**(2), 1-17.

[국외 문헌]

5. Alex, K., Sutskever, I., & Hinton, G. (2012). Imagenet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems, 25*, 1097-1105.
6. Cortes, C., Mohri, M., & Rostamizadeh, A. (2012). *L2 regularization for learning kernels*. arXiv preprint arXiv:1205.2653.
7. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., & Houlsby, N. (2020). *An image is worth 16x16 words: Transformers for image recognition at scale*. arXiv preprint arXiv:2010.11929.
8. Huang, L., Qin, J., Zhou, Y., Zhu, F., Liu, L., & Shao, L. (2020). *Normalization techniques in training DNNs: Methodology, analysis and application*. arXiv preprint arXiv:2009.12836.
9. Huang, T., Zhang, Z., & Zhang, J. (2019). FiBiNET: Combining feature importance and bilinear feature interaction for click-through rate prediction. In *Proceedings of the 13th ACM Conference on Recommender Systems*, 169-177.
10. Lian, J., Zhou, X., Zhang, F., Chen, Z., Xie, X., & Sun, G. (2018). xDeepFM: Combining explicit and implicit feature interactions for recommender systems. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1754-1763.
11. Liu, B., Tang, R., Chen, Y., Yu, J., Guo, H., & Zhang, Y. (2019). Feature generation by convolutional neural network for click-through rate prediction. In *Proceedings of the World Wide Web Conference*, 1119-1129.
12. Ng, A. Y. (2004). Feature selection, L1 vs. L2 regularization, and rotational invariance. In *Proceedings of the 21st International Conference on Machine Learning*, 78.
13. Song, W., Shi, C., Xiao, Z., Duan, Z., Xu, Y., Zhang, M., & Tang, J. (2019). AutoInt: Automatic feature interaction learning via self-attentive neural networks. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, 1161-1170.
14. Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *Proceedings of the 28th International Conference on Neural Information Processing Systems*, 3104-3112.
15. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 6000-6010.
16. Wang, R., Fu, B., Fu, G., & Wang, M. (2017). Deep & cross network for ad click predictions. In *Proceedings of The 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1-7.
17. Wu, Y., & He, K. (2018). Group normalization. In *Proceedings of the European Conference on Computer Vision*, 3-19.

저 자 소 개

김 태 석 (Tae-Suk Kim)

현재 배재대학교 경영대학 경영학과 조교수로 재직 중이다. 한국과학기술원에서 산업경영학 학사, 산업공학 석사, 박사 학위를 취득하였고, University of Illinois Urbana-Champaign (UIUC)와 University of California, Riverside(UCR)에서 Post-Doc, 삼성종합기술원에서 전문연구원을 역임하였다. 주요 관심분야는 IT 경영, Data mining, 지능형 정보기술 등이다. 지금까지 IEEE Transactions on Networking, IEEE Transactions on Consumer Electronics, IEEE Transactions on Mobile Computing, IEEE transactions on Vehicular Technology 등 주요 학술지에 논문을 발표하였다.

김 석 훈 (Seokhun Kim)

현재 배재대학교 경영대학 전자상거래학과 교수로 재직 중이다. 한남대학교에서 컴퓨터공학 박사 학위를 취득하였고, 수원여자대학교 교수를 역임하였다. 주요 관심분야는 E-Commerce 시스템, 웹 데이터베이스, 모바일 웹 프로그래밍 기술 등이다. 지금까지 Multimedia Tools and Applications, Applied Sciences, Intelligent Automation and soft computing 등 주요 학술지에 논문을 발표하였다.

임 광 혁 (Kwang Hyuk Im)

현재 배재대학교 경영대학 전자상거래학과 교수로 재직 중이다. 한국과학기술원에서 전산학 학사, 산업공학 석사, 박사 학위를 취득하였고, 삼성전자 반도체연구소 책임연구원을 역임하였다. 주요 관심분야는 데이터마이닝, 빅데이터, 경영정보시스템, 지능형 정보시스템 및 보안기술 등이다. 지금까지 Applied Intelligence, Expert Systems with Applications, Applied Mathematics and Information Sciences 등 주요 학술지에 논문을 발표하였다.

〈 Abstract 〉

A Study on Utilization of Vision Transformer for CTR Prediction

Tae-Suk Kim*, Seokhun Kim**, Kwang Hyuk Im***

Click-Through Rate (CTR) prediction is a key function that determines the ranking of candidate items in the recommendation system and recommends high-ranking items to reduce customer information overload and achieve profit maximization through sales promotion. The fields of natural language processing and image classification are achieving remarkable growth through the use of deep neural networks. Recently, a transformer model based on an attention mechanism, differentiated from the mainstream models in the fields of natural language processing and image classification, has been proposed to achieve state-of-the-art in this field. In this study, we present a method for improving the performance of a transformer model for CTR prediction. In order to analyze the effect of discrete and categorical CTR data characteristics different from natural language and image data on performance, experiments on embedding regularization and transformer normalization are performed. According to the experimental results, it was confirmed that the prediction performance of the transformer was significantly improved when the L2 generalization was applied in the embedding process for CTR data input processing and when batch normalization was applied instead of layer normalization, which is the default regularization method, to the transformer model.

Key Words: Click-Through Rate, Deep Neural Network, Recommender systems, e-business

* Department of Business Administration, Pai-Chai University
** Department of Electronic Commerce, Pai-Chai University
*** Department of Electronic Commerce, Pai-Chai University