클라이언트 기반 딥러닝을 이용한 검색 카테고리 추천 시스템

안철용*1), 박지수**, 손지곤*2)
*한국방송통신대학교 대학원 정보과학과
**동국대학교 융합교육원
e-mail:mydream0726@gmail.com

A Search Category Recommendation System Using Client-based Deep Learning

Cheol-Yong Ahn*, JiSu Park**, Jin Gon Shon*
*Dept. of Computer Science, Graduate School, Korea National Open University
**Convergence Institute, Dongguk University

요 약

웹 사이트 사용자들은 자신의 취향에 맞춘 웹 사이트 개인화 서비스를 원한다. 이에 따라 관련 기업들은 웹 사이트의 회원가입을 통해 사용자들의 개인 정보를 관리하여 개인화 서비스를 지원하고 있다. 하지만 기업들의 개인 정보 유출 사고와 잘못된 기업 간 공유로 개인 정보보호 관리에 어려움이 있다는 문제점이 있다. 본 논문에서는 클라이언트 기반 딥러닝(Client-based Deep Learning)과 웹 브라우저 표준 데이터베이스 IndexedDB를 사용하여 검색 카테고리 추천 시스템을 구현한다.

1. 서론

전 세계적으로 스마트 기기와 데스크톱의 보급률 증가로 웹 사이트 사용자들이 급격하게 증가하고 있다. 웹 사이트 사용자들은 개인 취향에 맞춘 검색, 뉴스, 상품 추천 등 다양한 개인화 서비스를 제공받기 원한다[8]. 웹 사이트 운영 기업들은 이러한 요구사항을 해결하기 위해 사용자들에게 회원가입을 통해 개인화 서비스를 제공하고 있다. 현재 이러한 회원가입 방식의 개인화 서비스는 개인정보를 기업 스토리지로 관리하기에 개인정보 유출에 대한 문제점을 가지고 있다. 개인정보 유출로 인한 피해는 해마다 증가가 되고 있으며 그에 따른 개인정보보호에 대한 필요성이 사회적으로 확산되고 있다. 또한 기업들 간의 개인정보 공유로 인해 개인의 입장에서는 스팸메일, 불필요한 통화로 이어지는 결과를 가져온다[1]. 이러한 문제를 해결하기 위해서는 웹 사이트 사용자의 개인정보를 로컬 스토리지에 저장하여 비로그인 사용자에게 개인화 서비스를 제공할 수 있는 방법이 필요하다[2].

비로그인 웹 사이트 사용자에게 무의식적 개인화 서비스를 제공하기 위해서는 사용자 측면에서 실행되는 개인화 서비스 시스템이 구현되어야 한다. 사용자 측면에서 개인화 서비스가 실행되면 회원가입의 필요성을 최소화할 수 있으며 기업에게 개인정보 제공을 최소화할 수 있는 이점이 발생된다. 사용자 측면의 개인화 서비스 구현을 위해서는 사용자가 웹 사이트를 접속 하였을 때 실행되는 클라이언트 스크립트 언어인 자바스크립트로 구현이 필요하다. 자바스크립트는 웹 브라우저 내에 다양한 웹 시스템 개발 기능을 제공하며 신기술 진화를 촉진시킬 수 있는 다양한 라이브러리를 지원한다[3]. 그리고 사용자의 웹 사이트 이용 패턴을 학습하기 위해서는 사용자 디바이스에서 수행되는 딥러닝 구현이 필요하다. 클라이언트 기반 딥러닝은 자바스크립트 베이스로 구현할 수 있는 딥러닝 모델이다.

본 논문에서는 검색 카테고리 추천시스템을 구현하여 클라이언트 기반 딥러닝 적용 방법과 개인정보보호 기반 웹 사이트 개인화 방안을 제시한다. W3C가 권고한 웹 브라우저 표준 데이터베이스 IndexedDB를 사용하여 사용자의 검색 카테고리 선택 정보와 검색한 시간정보를 저장한다. 저장된 정보를 클라이언트 기반 딥러닝 라이브러리 Tensorflow.js로 신경망 모델을 구현한다[4]. 구현된 신경망 모델을 통해 웹 사이트에 접속한 시간과 선택했던 검색 카테고리 정보를 학습하여 사용자에게 검색 카테고리 정보를 검색 창에 제공한다.

2. 관련연구

2.1 딥러닝 추천 모델

현재 많은 기업들은 딥러닝 모델을 이용한 추천 시스템을 연구하고 있으며 고성능의 워크스테이션에서 추천 시스템을 서비스하고 있다. 딥러닝 모델 추천 시스템의 주요 기술은 여러 가지가 있다. 하나 이상의 입력 계층과 은닉 계층, 출력 계층으로 구성된 신경망인 Multilayer

1) 한국방송통신대학교 대학원 재학생
2) 교신저자

Perceptron과 단어를 밀집 벡터의 형태로 표현하는 Word Embedding이 있다[5]. 추천 시스템에서 많이 사용되는 머신러닝 학습 기법인 Factorization Machine은 성능이 우수하다는 효과를 입증하고 있다[6]. 이러한 기술을 바탕으로 구현된 딥러닝 추천 시스템은 고성능 워크스테이션에서 수행이 되어야 한다. 본 논문에서 제시한 검색 카테고리 추천 시스템은 웹 사이트 사용자의 스마트 폰, 태블릿 PC, 데스크톱에서 수행되기 때문에 추천 시스템의 주요 기술 적용에는 성능의 한계가 있다.

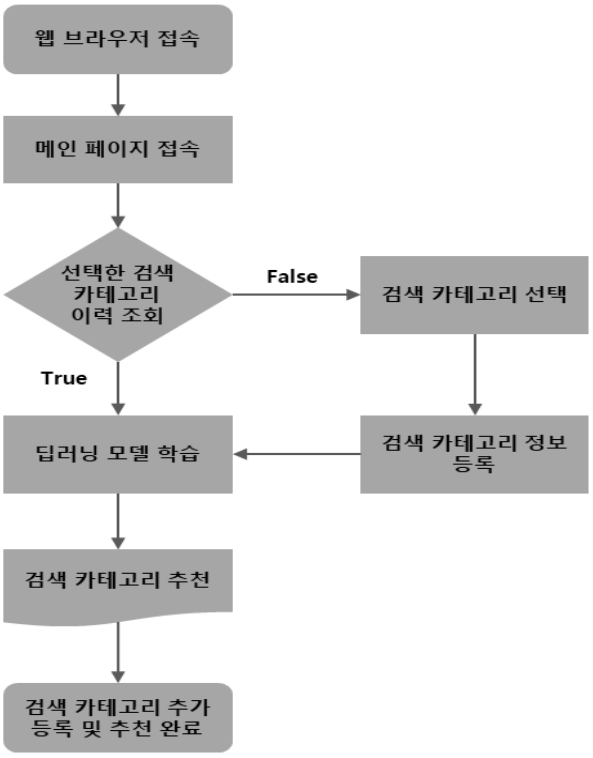
2.2 클라이언트 기반 딥러닝

웹 사이트 사용자의 스마트 폰 또는 데스크톱에서 수행되는 딥러닝은 사용자의 디바이스 CPU와 GPU를 사용하여 딥러닝 모델을 수행하기 때문에 사용자 중심의 딥러닝 모델을 구현할 수 있다는 특징이 있다. 이러한 특징으로 사용자 디바이스에서 수행되는 딥러닝은 신기술의 진화를 촉진시키는 결정적인 역할이 될 것으로 예측하고 있다[7]. 클라이언트 기반 딥러닝 라이브러리는 Caffe, TFLite, TensorFlow등 많은 종류의 라이브러리가 있지만 이 라이브러리들은 스마트폰 APP에 종속적으로 구현된다. 다른 스마트 폰 또는 다른 시스템에서 기존에 구현된 APP의 딥러닝 모델을 사용할 수가 없어 딥러닝 모델을 다시 구현해야 한다는 특징이 있다.

본 논문에서 제시한 검색 카테고리 추천 시스템의 딥러닝 모델은 다른 APP 또는 다른 시스템에서 딥러닝 모델을 다시 구현하지 않고 웹 사이트 접근만으로 실행할 수 있게 해야 한다. 그렇기 위해서 클라이언트 기반 딥러닝 라이브러리 Tensorflow.js를 이용하여 딥러닝 모델을 구현한다. Tensorflow.js는 웹 브라우저의 자바스크립트 기반에서 실행할 수 있도록 설계되어 있다. 웹 브라우저 내부에서 딥러닝 모델을 실행 시 WebGL을 통해 해당 디바이스의 GPU를 사용할 수 있으며, 자바스크립트 기반으로 딥러닝 모델을 실행 시에는 해당 디바이스 CPU만 사용한다[4].

3. 검색 카테고리 추천 시스템

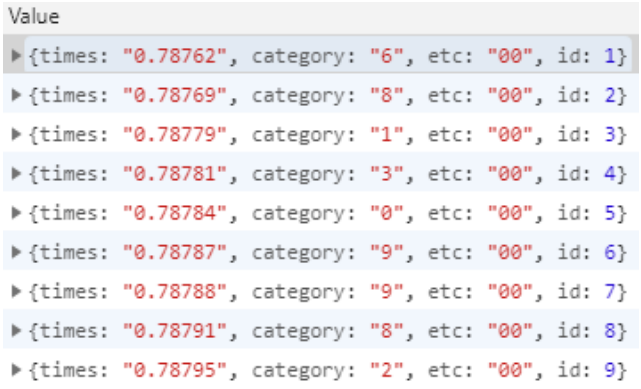
이번 장에서는 본 논문에서 제안된 검색 카테고리 추천 시스템을 설명한다. 제안된 시스템은 (그림 1)과 같다. 사용자는 추천 시스템이 구현되어 있는 웹 사이트의 메인 페이지에 접속하면 시스템은 IndexedDB에 검색 카테고리 데이터가 있는지 조회한다. 등록된 검색 카테고리 데이터가 있으면 Tensorflow.js 로 구현된 신경망 모델이 과거의 검색 카테고리 선택 정보를 학습하여 현재 시간에 선택될 높은 확률의 카테고리를 추천한다. 검색 카테고리 선택 박스, 입력 박스, search 버튼은 HTML 태그로 만들었다. 검색 카테고리 데이터가 없거나 추가로 등록을 하기 위해서는 카테고리를 선택하여 search버튼을 클릭하면 IndexedDB에 현재의 시간 정보와 카테고리 정보를 등록한다.



(그림 1) 검색 카테고리 추천 흐름도

3.1 검색 카테고리 등록 알고리즘

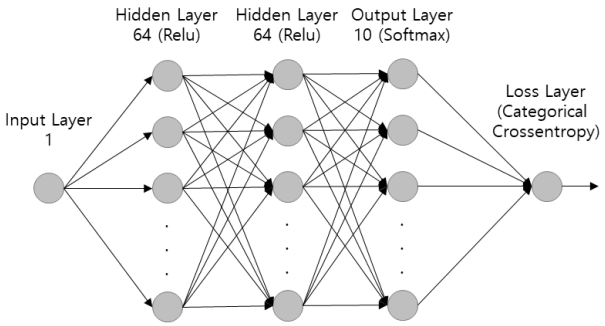
카테고리는 news, study, job 등 총 10가지의 항목으로 구분되어 있다. (그림 2)의 IndexedDB 정보를 보게 되면 times, category, etc, id 총 4개의 칼럼을 가지고 있다. times 칼럼은 시간 정보를 하루 24시간을 초 단위 기준으로 Normalization를 적용한 데이터이다. (현재 시간(초) - 1초) / (86400초 - 1초)의 공식을 적용하여 데이터의 범위를 0과 1사이로 변환하여 등록된다. category 칼럼은 총 10개의 카테고리 값을 0~9 사이의 인덱스 값으로 저장하였다. etc 칼럼은 예비 칼럼으로 만들었으며 id 칼럼은 IndexedDB에서 자동으로 등록되는 고유키 값이다 칼럼은 예비 칼럼으로 만들었으며 id 칼럼은 고유키 값이다.



(그림 2) IndexedDB에 등록된 카테고리 데이터

3.2 딥러닝 모델 학습

Tensorflow.js로 구현된 신경망은 IndexdDB에 저장된 시간 정보와 카테고리 정보를 학습 후 카테고리를 추천한다. IndexdDB에서 제공하는 Cursor 객체를 통해 테이블 기준으로 저장된 데이터를 조회할 수 있다. time 칼럼의 데이터는 0~1 사이의 float 형태의 데이터 타입으로 신경망 학습의 X 값으로 사용되고, category 칼럼의 데이터는 0~9의 int형 데이터 타입으로 One-hot Encoding을 적용하여 신경망 학습의 레이블 값 Y로 적용하였다. 신경망의 네트워크 구성은 (그림 3)과 같이 4개의 계층으로 구성하였다. 입력 계층은 time 데이터 1개를 입력할 수 있는 노드로 구성하였고 은닉 계층은 Relu 활성화 함수를 적용한 64개의 노드, 출력 계층은 10가지의 항목을 가진 카테고리 값을 출력하기 위한 Softmax 노드로 구성하였다. 옵티마이저는 RMSProp으로 사용하였으며 손실 함수는 Categorical Crossentropy로 사용하였다. Epoch의 값은 10이고 BatchSize는 10이다.



(그림 3) 검색 카테고리 추천시스템의 딥러닝 신경망 네트워크 구성도

4. 실험

본 논문에서 제안한 시스템의 성능 평가를 위해 데스크톱(PC)과 스마트 폰으로 진행하였다. 데스크톱의 환경은 OS:윈도우 10 프로페셔널, CPU:AMD Ryzen2600x, GPU:Nvida Gtx1060이고 스마트 폰은 GALAXY S8, OS:Android 9 버전이다. 데스크톱 환경에서의 웹 브라우저는 Chrome 76.0 버전과 FireFox 3.0 버전을 사용하였다. 스마트 폰 환경에서 브라우저는 Samsung Browser 9.4 버전을 사용하였다. 2개의 디바이스와 3개의 브라우저 환경에서 진행하는 이유는 Tensorflow.js, IndexedDB가 정식으로 크로스 플랫폼을 지원되는지에 대한 실험이 필요하다. 학습 데이터는 100개의 검색 카테고리 정보를 각각의 브라우저 IndexdDB에 등록하였다. 등록 방법은 데스크톱과 스마트 폰 브라우저에서 구현된 검색 카테고리 시스템의 카테고리 선택 화면에서 카테고리를 임의로 선택하여 search 버튼을 클릭하여 등록하였다.

실험 결과는 표 1과 같다. 100개의 카테고리 데이터를 학습한 정확도는 저조하며 로딩 시간 또한 많은 것으로 나타난다. 데스크 탑의 Chrome 브라우저와 Firefox의 정확도는 비슷하지만 로딩 시간에서 많은 차이를 보였으며 스마트 폰의 로딩 시간은 데스크톱과 비교해서 길었지만 정확도는 데스크톱 보다 좋은 결과가 나왔다.

<표 1> 데스크톱과 스마트 폰 환경에서 테스트한 평균 정확도와 평균 로딩 시간

구분	데스크톱 (Chrome)	데스크톱 (Firefox)	스마트 폰 (Samsung Browser)
정확도(%)	19.1	23.9	24.4
로딩 시간 (초)	2.89	4.87	10.22

5. 결론

본 논문에서는 웹 사이트 사용자의 검색 카테고리 정보와 시간 정보를 IndexedDB에 저장하고 저장된 검색 카테고리 정보를 조회하여 딥러닝 학습 모델을 통해 검색 카테고리를 추천하는 시스템을 제안하였다.

Tensorflow.js로 웹 브라우저에서 딥러닝 모델을 수행하여 클라이언트 기반 딥러닝 구현 방법을 제시하였고 웹 사이트 사용자 브라우저에 주요 정보를 저장함으로써 웹 사이트 관리 기업의 개인 정보 유출을 최소화할 수 있는 방안을 제시하였다. 사용자들의 스마트 폰, 데스크톱의 딥러닝 학습 성능은 기업이 운영하는 고성능의 워크스테이션보다 낮은 퍼포먼스를 낸다. 이러한 한계점 해결을 위해 클라이언트 기반 딥러닝 모델의 추가적인 연구가 필요하다.

참고문헌

[1] Sun-Young Lee, “Analysis of User’s Recognition for Personal Information Agreement and New Policy”, The Journal of Korean Institute of Information Technology 12(8), 2014.08, 85-92

[2] Hee Won Myeong, Jung Ha Paik, Dong Hoon Lee “Study on implementation of Secure HTML5 Local Storage”, Journal of Korean Socieity for Internet Information 2012. Aug: 13(4): 83-93

[3] Xiuquan Qiao, Pei Ren, Schahram Dustdar, Ling Liu , Huadong Ma, Junliang Chen, “Web AR: A Promising Future for Mobile Augmented Reality－State of the Art, Challenges, and Insights”, Proceedings of the IEEE, Vol. 107, No. 4, April 2019

[4] Daniel Smilkov, “TENSORFLOW.JS: MACHINE LEARNING FOR THE WEB AND BEYOND”, arXiv:1901.05350v2 [cs.LG] 28 Feb 2019.

[5] Maxim Naumov, “Deep Learning Recommend ation Model for Personalization and Recommenda

tion Systems", arXiv:1906.00091v1 [cs.IR] 31 May 2019
[6] Steffen Rendle, "Factorization machines" In Proc. 2010 IEEE International Conference on Data Mining, 2010, 995 - 1000.
[7] Mengwei Xu, Jiawei Liu, Yuanqiang Liu, Felix Xiaozhu Lin, Yunxin Liu, & Xuanzhe Liu, "A First Look at Deep Learning Apps on Smartphones", arXiv:1812.05448v2 [cs.LG] 23 Jan 2019
[8] Hong, Hui-Yun, "모바일 인터넷 이용 성향 및 모바일 검색 요구 기반의 검색 기술 연구 방향", Volume 26, Issue 4, Pages.26-30, 2009, 1226-4725(pISSN)