

기계학습을 이용한 동영상 서비스의 검색 편의성 향상

임연섭*

Machine Learning Assisted Information Search in Streaming Video

Yeon-sup Lim*

*Assistant Professor, Department of Convergence Security Engineering, Sungshin Women's University, Seoul, 02844 Korea

요약

유튜브와 같은 동영상 스트리밍 서비스에서의 정보 검색은 전통적인 정보 검색 서비스를 대체하고 있다. 이러한 동영상 안에서 원하는 세부적인 정보를 찾기 위해서는 사용자가 여러 부분을 반복해서 탐색하며 시간과 네트워크 대역폭을 낭비해야 하는 문제점이 있다. 본 논문에서는 클러스터링과 LSTM을 이용하여 이러한 사용자의 동영상 내 정보 검색을 보조하는 방법을 제안한다. 제안하는 방법은 사용자의 정보 검색을 위한 탐색 지점 순서와 DBSCAN이 범주화한 최종 목적 지점 범주를 이용하여 LSTM 모델을 학습하고, 이 모델을 이용하여 사용자가 검색을 시작할 때 선택한 탐색 지점 순서에 기반을 둔 사용자의 예상 목적 지점 범주를 제시한다. 실험 결과, 제안하는 방법이 사용자가 원하는 목적 지점을 평균적으로 98%의 정확도와 7초의 시간 오차로 찾아내는 것을 보였다.

ABSTRACT

Information search in video streaming services such as YouTube is replacing traditional information search services. To find desired detailed information in such a video, users should repeatedly navigate several points in the video, resulting in a waste of time and network traffic. In this paper, we propose a method to assist users in searching for information in a video by using DBSCAN clustering and LSTM. Our LSTM model is trained with a dataset that consists of user search sequences and their final target points categorized by DBSCAN clustering algorithm. Then, our proposed method utilizes the trained model to suggest an expected category for the user's desired target point based on a partial search sequence that can be collected at the beginning of the search. Our experiment results show that the proposed method successfully finds user destination points with 98% accuracy and 7s of the time difference by average.

키워드 : 동영상 스트리밍, 기계학습, 클러스터링, LSTM, 시퀀스 예측

Keywords : Video streaming, Machine learning, Clustering, LSTM, Sequence prediction

Received 15 December 2020, Revised 11 January 2021, Accepted 27 January 2021

* Corresponding Author Yeon-sup Lim(E-mail:yylim@sungshin.ac.kr, Tel:+82-2-920-7144)

Assistant Professor, Department of Convergence Security Engineering, Sungshin Women's University, Seoul, 02844 Korea

Open Access <http://doi.org/10.6109/jkiice.2021.25.3.361>

print ISSN: 2234-4772 online ISSN: 2288-4165

©This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.
Copyright © The Korea Institute of Information and Communication Engineering.

I. 서 론

정보 통신 기술의 발달은 문자 중심의 정보 전달이 아닌 동영상 등의 멀티미디어 매체를 통한 정보의 생산과 전달을 가능하게 하였다. 오늘날 인터넷에서 동영상을 전달하기 위한 유튜브 같은 서비스는 전체 트래픽의 65% 이상을 차지하고 있으며 [1], 정보의 검색에 있어서 사용자의 60% 이상이 기존의 검색 서비스가 아닌 이러한 동영상 스트리밍 서비스를 이용하고 있다 [2].

대부분의 동영상 스트리밍 서비스는 정보 검색을 위해 동영상 정보에 대한 키워드 검색을 제공하고 있다. 키워드 검색을 통해 사용자는 원하는 주제의 동영상에 빠르게 접근할 수 있지만, 해당 동영상에서 구체적으로 원하는 정보를 찾기 위해서는 전체 동영상을 시청하거나 전부를 시청하지 않더라도 탐색 도구를 통해 계속해서 원하는 부분을 찾아야 하는 한계점이 있다. 이와 관련하여 넷플릭스의 경우 영화나 드라마의 소개 부분을 넘기고자 하는 사용자 요구에 대해 Skip Intro 기능을 제공하지만, 이는 특정 지점의 동영상 정보에 대한 검색 편의성을 위한 기능이라고 볼 수 없다. 또한, 스트리밍 플레이어에서 검색 편의성을 위해 영상의 특정 지점에 대한 축소 사진을 제공하기도 하지만 이러한 축소 사진만으로는 원하는 정보의 검색은 쉽지 않다.

특히 COVID-19 감염 사태로 인한 온라인 수업의 확대는 이러한 동영상 내 세부 정보 검색 방법의 필요성을 더욱 증대시키고 있다. 예를 들어 동영상 강의를 시청하는 학생이 시청하고 있는 부분과 관련한 수업 내용 부분을 찾기 위해서는 기억에 의존하여 원하는 지점에 이를 때까지 반복해서 탐색을 계속해야 한다. 이는 불필요한 네트워크 트래픽 전송과 정보 검색 시간의 증가로 사용자 경험에 불편함을 초래할 수 있다.

본 논문에서는 기계학습을 이용하여 동영상 시청자들이 탐색한 패턴을 학습 후, 사용자가 정보 검색을 시작할 때 선택한 부분적인 탐색 순서에 기반하여 원하는 정보가 포함된 지점을 제시하는 방법을 제안한다.

II. 관련 연구

2.1. 기존 연구

최근 인터넷을 통한 동영상 스트리밍 서비스의 성장

은 사용자의 시청 패턴에 대한 분석과 서비스 경험 향상을 위한 연구에 많은 관심을 일으키고 있다.

Srinidhi [3]는 모바일 인터넷에서 데이터 사용량, 이동 패턴, 애플리케이션 사용과 같은 사용자의 행동을 조사하였다. Almeida [4]는 교육용 스트리밍 서비스에서의 사용자 세션과 상호작용에 대하여 분석하였다. Lobo [5]는 스트리밍 서비스에서 사용자 행동의 의존관계를 통계적 분석을 수행하였다. Li [6]는 모바일 실시간 비디오 스트리밍에서 사용자 행동의 특징을 분석하였다. Moldovan [7]는 모바일 비디오 스트리밍에서 수집한 시청자 행동을 분석하여 상호 간에 상관관계가 있음을 밝혔다.

Brampton [8]과 Wang [9]은 사용자 행동의 통계적인 분포를 이용하여 사용자들이 가장 관심을 가지고 시청한 부분으로의 연결을 제공하는 방법을 제안하였다. 영상에서의 가장 관심이 많은 부분으로의 직접적인 연결은 사용자가 찾고자 하는 정보 검색을 위한 도움이 될 수 있지만, [8, 9]의 연구에서는 사용자의 검색 과정에서 발생하는 여러 번의 지점 탐색을 고려하지 않고 단순히 특정 지점에서 다른 지점으로의 조건부 확률에 기반한 방법을 제안하는 한계가 있다. Laiche [10]은 동영상 서비스에서 사용자의 행동 패턴을 학습하여 사용자 경험 품질을 예측하는 기계학습 모델을 제안하였다. 하지만 [10]의 연구는 동영상 화질 등에 의한 평균 의견 점수에 집중하여 동영상 내 정보 검색 경험을 고려하지 않고 있다.

기계학습 알고리즘 중 RNN (Recurrent Neural Networks)은 센서가 감지한 데이터, 추가, 문자열 등의 순서를 가진 데이터를 처리하는 데 활용된다. 이러한 RNN에서는 필요한 데이터의 시간 격차가 클수록 성능이 저하될 수 있는 문제점이 있다. Hochreiter와 Schmidhuber가 제시한 LSTM (Long Short Term Memory) [11]은 이러한 RNN의 긴 의존 기간 문제를 해결하여 문자열 예측 등 여러 분야에서 성공적으로 RNN을 적용할 수 있게 하였다. 예를 들어, [12, 13]의 연구에서는 LSTM을 적용한 RNN을 이용하여 문장을 생성하거나, 문장 안의 단어에 내재한 속성을 추측하는 모델을 제안하였다. 본 논문에서는 동영상 내 사용자의 정보 검색 과정에서 순차적인 탐색 지점 데이터가 발생함에 주목하여 LSTM 네트워크를 사용자가 원하는 목적 정보를 가진 지점의 예측에 활용한다.

2.2. Long Short Term Memory Networks

전통적인 인공 신경망은 시계열로 입력된 데이터를 적극적으로 활용하지 못하는 문제점이 있었다. 이전에 수집한 데이터를 나중에 발생할 데이터 예측에 활용하지 못하는 것이다. RNN은 자신을 반복 학습하여 은닉 계층에 저장된 이전 단계에서 얻은 정보를 현재의 입력 값과 함께 예측에 사용하여 이러한 문제점을 해결하였다 [14]. 그림 1은 입력 x_t 를 받아 h_t 를 출력하는 RNN의 기본적인 반복 구조를 나타낸다.

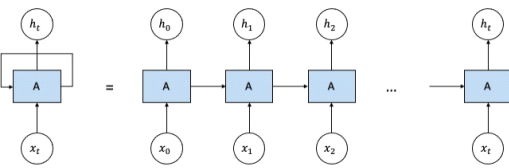


Fig. 1 Recurrent Neural Networks [14]

LSTM은 RNN의 긴 의존 기간으로 인한 문제점을 각각의 블록에 시간 순서에 따른 cell state (C_t)를 적용하여 해결하였다. 그림 2에 보이는 것과 같이 LSTM의 셀은 잊기 게이트 (forget gate), 입력 게이트 (input gate), 출력 게이트 (output gate)에 의해 제어된다. RNN과 마찬가지로 스스로를 반복 학습하여 최종 출력값을 계산하지만, 위의 세 가지 게이트를 통해 학습하는 정보를 조절한다.

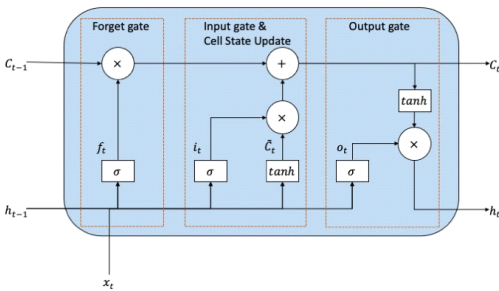


Fig. 2 LSTM Cell Structure

첫 번째로 잊기 게이트를 통해 과거의 cell state (C_{t-1})로부터 어떤 정보를 잊을 것인지를 결정한다. 식 (1)과 같이 잊기 게이트는 h_{t-1} 과 x_t 의 입력에 가중치 (W_f)와 sigmoid 함수를 적용하여 0과 1 사이의 값으로 이루어진 벡터 f_t 를 계산한다. f_t 는 C_{t-1} 에 적용 (element-wise product, $C_{t-1} \times f_t$)되며 1은 정보를 기억

함을 0은 정보를 잊는다는 것을 뜻한다. 따라서 $C_{t-1} \times f_t$ 는 시간 t 에서 계산에 사용할 과거 (시간 $t-1$)의 정보를 의미한다.

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (1)$$

다음으로는 새로운 정보 중 어떤 것을 cell state에 저장할 것인지를 입력 게이트를 통해 결정한다. 잊기 게이트와 마찬가지로 식(2)와 같이 h_{t-1} 과 x_t 의 입력에 가중치 (W_i)와 sigmoid를 통해 어떠한 값을 갱신에 사용할 것인지 결정된 후, 식(3)에서 tanh 활성화 함수를 이용해 생성한 cell state 후보 벡터 \tilde{C}_t 에 적용하여 계산에 사용할 새로운 정보 $\tilde{C}_t \times i_t$ 를 결정한다. 따라서 과거와 새로운 정보를 고려한 현재의 cell state C_t 는 식(4)와 같이 두 벡터 $C_{t-1} \times f_t$ 와 $\tilde{C}_t \times i_t$ 의 합으로 이루어진다.

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (2)$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \quad (3)$$

$$C_t = C_{t-1} \times f_t + \tilde{C}_t \times i_t \quad (4)$$

마지막으로 출력 게이트는 식(5)를 이용하여 출력에 사용할 정보를 선택하기 위한 벡터 o_t 를 생성한다. 현재의 cell state에서의 후보 출력은 tanh 활성화 함수로 계산하며 최종 출력 h_t 은 식(6)에 의해 결정한다.

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (5)$$

$$h_t = o_t \times \tanh(C_t) \quad (6)$$

III. 제안 연구

3.1. 연구 가설

본 연구는 온라인 스트리밍 서비스를 통한 강의 동영상과 같이 잦은 정보 검색이 요구되는 동영상을 대상으로 한다. 이러한 동영상의 시청 및 탐색 패턴에 대해 다음과 같은 가설을 설정하고 클러스터링과 기계학습 알고리즘을 적용한다.

- 특정 시청 지점에서 탐색을 시도하는 사용자들은 대부분 유사한 지점에서의 탐색을 수행한다. 예를 들어 특정 이론 강의 부분에서 의문점이 생긴 학생들은 연

관 정보를 찾기 위해 대부분 비슷한 지점으로 탐색을 시도할 것이다.

- 탐색을 시도할 때 사용자는 임의의 지점을 선택하는 것이 아니라 최대한 원하는 정보가 위치할 것으로 생각하는 지점을 선택한다.
- 선택한 지점이 원하는 정보를 포함하는지 확인하기 위해 사용자들은 짧은 시간 동안 내용을 확인하고, 원하는 지점이 아니라면 선택한 지점 근처를 다시 탐색할 것이다.
- 해당 지점이 원하는 정보를 포함할 경우, 사용자는 계속해서 동영상을 시청할 것이다.

이러한 가설 아래에서 기계학습 알고리즘에 적용하기 위한 정보를 정의하고, 사용자 동영상 탐색 패턴 데이터를 인공적으로 생성하여 제안한 알고리즘의 성능을 확인한다.

3.2. 클러스터링과 LSTM에 기반한 동영상 내 정보 검색 지점 예측

검색 지점 예측을 위하여 다음과 같은 알고리즘을 사용하여 LSTM모델을 학습시킨다. 그림 3은 제안한 방법의 개요를 나타낸 그림이다.

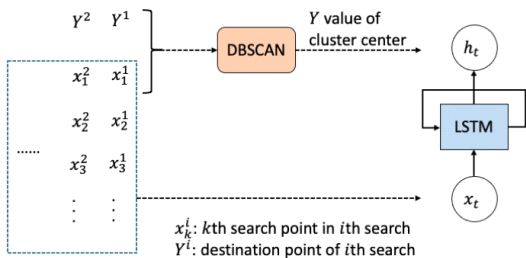


Fig. 3 Overview of the Proposed Method

학습 데이터에서 사용자의 최종 목적 지점은 (시작 지점, 최종지점) 데이터에 DBSCAN 클러스터링 알고리즘 [15]을 적용하여 범주화하고, 각 클러스터 중심의 최종지점 값을 예상 최종지점으로 정한다. LSTM 모델은 최종지점을 제외한 연속된 N개의 사용자의 탐색 지점 순서를 입력으로 하고 예상되는 목적 지점 범주의 확률을 출력으로 한다. 예를 들어 10개의 (시작 지점, 최종지점) 데이터 $(x^1, Y^1), (x^2, Y^2), \dots, (x^{10}, Y^{10})$ 가 하나의 클러스터를 형성하고 이 클러스터의 중심을 (x, Y) 라 할 때, 해당 10개의 검색 순서의 예상 최종지점은 Y 로

한다. 이후 LSTM은 최종지점을 제외한 검색 순서 $(x_1^1, x_2^1, x_3^1, \dots), (x_1^2, x_2^2, x_3^2, \dots), \dots, (x_1^{10}, x_2^{10}, x_3^{10}, \dots)$ 을 입력으로 하여 Y 에 대한 확률을 최대화하도록 학습시킨다.

이를 위해서 3개의 은닉 레이어를 사용하며, 첫 번째 은닉 레이어는 256개의 LSTM 셀에서 예측 지점 순서를 출력하게 하여 마지막 스텝에서 cost를 계산 후 오류가 전파되도록 설정한다. 두 번째 은닉 레이어는 첫 번째와 같은 LSTM을 사용하며 첫 번째 은닉 레이어와 완전 연결 (fully-connected)한다. 마지막 레이어는 softmax 활성화 함수 [16]를 사용하여 원하는 확률값으로 출력이 이루어지도록 한다. 각 은닉 레이어 사이에는 Dropout 정규화를 적용한다. 이는 레이어의 유닛을 고정된 개수만큼 무작위로 선택, 삭제해서 오버피팅을 방지하기 위함이다 [16]. 손실 함수는 범주형 교차 엔트로피 (categorical_crossentropy)를 사용하고 옵티마이저는 Adam [17]을 적용한다. 표 1은 Keras [18]로 표현한 모델 생성 코드이다.

Table. 1 Keras code for generating the LSTM model

```

model = Sequential()
model.add(LSTM(256, input_shape = (N, 1),
            return_sequences = True))
model.add(Dropout(p))
model.add(LSTM(256))
model.add(Dropout(p))
model.add(Dense(OUTPUTS, activation = 'softmax'))
model.compile(loss = 'categorical_crossentropy',
              optimizer = 'adam')
    
```

IV. 성능 평가

4.1. 실험 설정

실험을 위해 3.1절의 가정을 기반으로 인공적인 사용자의 동영상 탐색 데이터를 생성하였다. 사용자가 60분 (3600초) 길이의 동영상을 시청하는 동안 원하는 정보를 찾기 위해 탐색을 시작하는 다음과 같은 시나리오를 가정한다.

- 사용자는 전체 3600초 중 특정 위치에서 ± 60 초 범위 안의 임의의 위치로부터 탐색을 시작한다.
- 목적하는 지점에 가까워질수록 사용자는 작은 범위를 점프하여 탐색을 시도한다. 목적하는 지점과 탐색 지점의 차이의 1.5배 시간 범위에서 임의로 점프하는 간격을 선택함을 가정한다.

- 도착한 지점이 목적 지점과 차이가 10초 이하일 경우 사용자는 목적 지점에 도달하여 탐색을 멈추고 시청을 계속한다.

위와 같은 시나리오에서 5000명의 사용자가 각기 다른 정보 검색을 진행한 데이터를 생성한다. 전체 목적 지점의 수는 10개로 설정한다. 검색 과정에서 각 탐색 지점을 나타내는 값은 전체 길이 3600초로 정규화하여 0과 1 사이의 값을 가지도록 한다. 생성한 데이터 중 75%를 학습에 사용하고 나머지 25%를 검증에 사용한다. 사용자가 목적인 최종지점의 범주화는 학습 데이터에만 기반하여 수행한다. 검증 데이터에서 사용자 목적 지점의 범주는 학습 데이터에서 생성한 범주 중 가장 가까운 범주로 정한다. 생성한 데이터에서 얻어진 사용자의 탐색 지점 순서의 최대 사용자 검색 순서의 길이 N은 15로 설정한다. 길이가 15 미만인 순서는 패딩을 적용하여 모델에 동일 길이로 입력할 수 있도록 한다. 나머지 파라미터인 Dropout 확률 p는 0.2로 설정하였다.

또한, 아래의 예와 같이 각각의 순서에서 시작 지점이 아닌 중간 탐색 지점을 순차적으로 제거한 부분적인 순서 또한 학습과 검증 데이터에 추가한다.

- 생성 순서 예: 0.2 → 0.1 → 0.04 → 0.03 → 0.01
- 추가하는 순서 예: 0.2 → 0.04 → 0.03 → 0.01, 0.2 → 0.03 → 0.01, ...

이와 같은 방법으로 학습과 검증에 사용할 10개의 실험 데이터를 생성하여 LSTM 모델을 100 epoch 동안 학습시킨 후 가장 작은 손실을 보인 모델을 선택하여 성능을 확인한다.

4.2. 실험 결과

첫 번째로 테스트 탐색 순서 입력에 대한 최종지점 예측의 정확도를 확인한다. 그림 4는 각 검증 데이터를 이용한 10번의 실험에서 제안 지점 범주와 실제 목적 범주가 일치한 정확도를 나타낸 그래프이다. 차이를 분명하게 확인하기 위해 y축 원점을 0.9로 설정하였다. 그림 4에서 보이는 바와 같이 제안한 방법은 모든 실험에서 0.9 이상의 정확도를 나타내었다. 평균 정확도는 0.98, 표준 편차는 0.02로 제안한 방법이 성공적으로 사용자의 탐색 순서로부터 목적 지점을 예측함을 증명한다. 또

한, 4.1절에 보인 바와 같이 검증 데이터에 전체 순서가 아닌 부분적인 순서도 포함함을 고려할 때, 전체 탐색 순서가 아닌 몇 개의 지점 탐색 순서만으로도 성공적으로 최종지점을 예측할 수 있음을 알 수 있다.

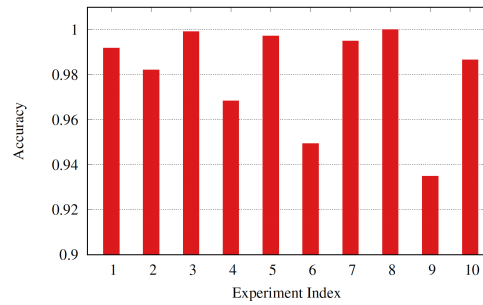


Fig. 4 Prediction Accuracy

다음으로 제안한 방법이 얼마나 사용자가 목적인 지점에 가까운 지점을 제시했는지를 확인한다. 그림 5는 각 실험에서 실제 목적 지점과 알고리즘이 제시한 지점의 평균 시간 차이를 나타낸 그래프이다.

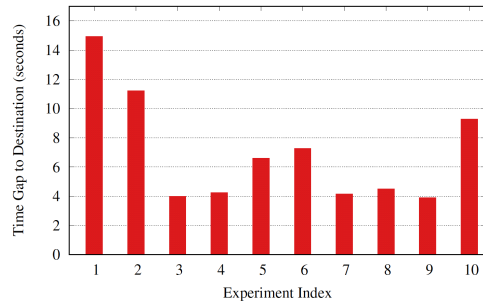


Fig. 5 Time Gap between Prediction and Ground Truth

그림 5에서 확인할 수 있듯이, 최대 15초, 최소 4초의 시간 차이가 있었으며, 평균적으로 7초의 시간 차이를 보였다. 이는 DBSCAN 클러스터링으로 인하여 비슷한 검색 시작 지점과 목적 지점을 가지는 순서들이 같은 범주에 묶이기 때문이다. 또한, 제안한 방법은 사용자에게 각 범주에 해당하는 클러스터의 중심을 목적 지점으로 제안하므로 완벽하게 정확한 목적 지점을 제시할 수 없는 한계가 있다. 하지만 평균 7초의 시간 차이는 사용자가 해당 지점에서 원하는 정보에 도달했음을 확인하는데 소요하는 시간에 비하여 크게 길지 않으므로 추가적인 검색을 시도할 필요가 없을 가능성이 크다. 따라서 사용자는 알고리즘이 제시한 지점에서 시청을 계속함으로써 빠르게 원하는 정보에 도달할 수 있을 것이다.

마지막으로 정확도와 시간 차이에 영향을 줄 수 있는 클러스터링 결과에 대하여 살펴본다. 표 2는 각 학습 데이터에 목적 지점 범주화를 위해 DBSCAN을 적용했을 때 생성된 클러스터의 개수와 클러스터 중심의 모든 쌍 사이의 평균 유클리드 거리를 나열한 결과이다. 표 2에서 보이는 바와 같이 다수의 실험에서 설정한 10개의 목적 지점의 일부가 클러스터링 알고리즘에 의해 병합되었다. 클러스터의 수와 그림 4의 정확도 사이에는 유의미한 상관관계를 찾을 수 없었다. 그림 5의 시간 차이와 비교했을 때, 많은 수의 목적 지점이 병합될수록 (적은 클러스터의 수) 사용자의 실제 목적 지점과 알고리즘이 제안한 목적 지점의 시간 차이가 벌어짐을 확인할 수 있다. 클러스터의 중심 간 평균 거리의 경우, 정확도와 시간 차이에 유의미한 상관관계가 존재하지 않았다. 이는 시작 지점만이 아닌 사용자의 다음 탐색 지점을 포함한 순서가 알고리즘의 예측에 필요하므로 (예 - 영어 단어 예측에서 첫 글자만으로는 전체 단어의 예측이 어려움) 단순히 (시작 지점, 최종지점) 데이터의 클러스터 간 중심 거리는 성능에 영향이 적기 때문이다.

Table. 2 DBSCAN Clustering on Training Traces

Idx	# of Clusters	Avg. Euclidean Distance between Center Pairs
1	6	0.252
2	6	0.516
3	10	0.507
4	9	0.383
5	9	0.505
6	7	0.582
7	8	0.389
8	9	0.473
9	10	0.513
10	8	0.559

V. 결론

본 논문에서는 클러스터링과 LSTM 모델을 이용하여 사용자가 원하는 검색 목적 지점을 예측하는 방법을 제안하였다. 인공적으로 생성한 데이터를 이용한 실험에서 제안한 알고리즘이 평균적으로 98%의 정확도로 7초의 시간 차이를 보이는 지점을 제안함을 확인하였다.

제안 방법을 실제로 사용하기 위해서는 스트리밍 서버와 클라이언트 플레이어의 정보 교환 및 학습을 위한 구현체가 필요하다. 다음 연구로는 이러한 구현체를 이용한 대학 온라인 강의 시스템에서의 실험을 통해 연구 결과를 실증할 것이다.

ACKNOWLEDGEMENT

This work was supported by the Sungshin Women's University Research Grant of H20200096.

REFERENCES

- [1] Sandvine, The Mobile Internet Phenomena Report 1H 2020 [Internet]. Available: <https://www.sandvine.com/download-report-mobile-internet-phenomena-report-2020-sandvine>.
- [2] Nasmedia, Netizen Profile Research, 2019. [Internet]. Available: <https://www.slideshare.net/nasmedia/2019-npr-f/>.
- [3] V. Srinidhi, "Classification of User Behaviour in Mobile Internet," *Asia-pacific Journal of Convergent Research Interchange*, vol. 2, no. 2, pp. 9-18, 2016.
- [4] J. Almeida, J. Krueger, D. Eager, and M. Vernon, "Analysis of educational media server workloads," in *Proceedings of International Workshop on Network and Operating System Support for Digital Audio and Video*, 2001.
- [5] A. Lobo, R. Garcia, X. G. Paneda, D. Melendi, and S. Cabrero, "Modeling Video on Demand services taking into account statistical dependences in user behavior," in *Simulation Modelling Practice and Theory*, vol. 31, pp. 96-115, 2013.
- [6] Z. Li, M. Kaafar, K. Salamatian, and G. Xie, "User Behavior Characterization of a Large-scale Mobile Live Streaming System," in *Proceeding of the 11st International World Wide Web Conference*, 2015.
- [7] C. Moldovan, F. Wamser, and T. Hoßfeld, "User Behavior and Engagement of a Mobile Video Streaming User from Crowdsourced Measurements," in *Proceeding of the 11st International Conference on Quality of Multimedia Experience*, 2019.
- [8] A. Brampton, A. MacQuire, M. Fry, I. A. Rai, N. J. P. Race, and L. Mathy, "Characterising and exploiting workloads of highly interactive video-on-demand," in *Multimedia Systems*, vol. 15, pp. 3-17, 2009.

- [9] W. Wang, T. Xu, Y. Gao, and S. Lu, "Probabilistic seeking prediction in P2P VoD systems," in *Lecture Notes in Computer Science*, vol. 5866, pp. 676-685, 2009.
- [10] F. Laiche, A. Letaifa, I. Elloumi, and T. Agulli, "When Machine Learning Algorithms Meet User Engagement Parameters to Predict Video QoE," in *Springer Wireless Personal Communications*, 2020.
- [11] S. Hochreiter and J. Schmidhuber, "Long short-term memory," in *Neural Computation*, vol. 9, no. 8, pp. 1735-1780, 1997.
- [12] D. Gao, B. Chen, R. Lu, and M. Zhou, "Recurrent Hierarchical Topic-Guided RNN for Language Generation," in *Proceedings of the 37th International Conference on Machine Learning*, 2020.
- [13] A. Katiyar and C. Cardie, "Nested Named Entity Recognition Revisited," in *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2018.
- [14] colah's blog, Understanding LSTM Networks [Internet]. Available: <http://colah.github.io/posts/2015-08-Understanding-LSTMs>.
- [15] E. Schubert, J. Sander, M. Ester, H. Kriegel, and X. Xu, "DBSCAN Revisited, Revisited: Why and How You Should (Still) Use DBSCAN," in *ACM Transaction on Database Systems*, vol. 42, 2017.
- [16] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, MIT Press, pp. 180-184, 2016.
- [17] D. Kingma and J. L. Ba, "ADAM: A Method for Stochastic Optimization," in *Proceedings of the 3rd International Conference on Learning Representations*, 2015.
- [18] Keras, [Internet]. Available: <https://keras.io>.



임연섭(Yeon-sup Lim)

성신여자대학교, 융합보안공학과, 조교수, 2020.3-현재
 IBM T. J. Watson Research Center, Research Staff Member, 2017.2-2020.2
 University of Massachusetts Amherst, Computer Science, Ph.D., 2017
 ※ 관심분야 : 컴퓨터 네트워크, 클라우드 컴퓨팅, 기계 학습, 모바일 컴퓨팅