특집논문 (Special Paper)
방송공학회논문지 제24권 제4호, 2019년 7월 (JBE Vol. 24, No. 4, July 2019)
https://doi.org/10.5909/JBE.2019.24.4.553
ISSN 2287-9137 (Online) ISSN 1226-7953 (Print)

채팅과 오디오의 다중 시구간 정보를 이용한 영상의 하이라이트 예측

김 은 율[a)], 이 계 민[a)‡]

Video Highlight Prediction Using Multiple Time-Interval Information of Chat and Audio

Eunyul Kim[a)] and Gyemin Lee[a)‡]

요 약

최근 개인방송 플랫폼을 통해 업로드 되는 콘텐츠가 증가함에 따라 시청자의 편의를 위해 하이라이트 영상을 제공하는 서비스에 대한 수요가 증가하고 있다. 이에 본 논문에서는 영상의 하이라이트 위치를 자동으로 예측하는 모델을 제안한다. 제안하는 모델은 채팅과 오디오 정보를 이용하며 양방향 LSTM을 사용해 영상의 흐름을 이해한다. 또한 콘텐츠의 종류에 따라 단기적 흐름과 함께 중장기적 흐름을 파악하는 다중 시구간 모델도 함께 제안한다. 제안한 모델은 개인방송 플랫폼을 통해 중계된 e스포츠와 야구경기 영상들을 이용하여 평가하였으며, 다중 시구간 정보를 활용하는 것이 하이라이트 예측에 유용함을 보였다.

Abstract

As the number of videos uploaded on live streaming platforms rapidly increases, the demand for providing highlight videos is increasing to promote viewer experiences. In this paper, we present novel methods for predicting highlights using chat logs and audio data in videos. The proposed models employ bi-directional LSTMs to understand the contextual flow of a video. We also propose to use the features over various time-intervals to understand the mid-to-long term flows. The proposed Our methods are demonstrated on e-Sports and baseball videos collected from personal broadcasting platforms such as Twitch and Kakao TV. The results show that the information from multiple time-intervals is useful in predicting video highlights.

Keyword : Video highlight, Multiple time-interval models, Bi-directional LSTM, Chat logs, Audio

a) 서울과학기술대학교 나노IT디자인융합대학원 정보통신미디어공학전공(Dept. of Broadcasting · Communication Fusion Program, Graduate School of Nano IT Design Fusion, Seoul National University of Science and Technology)
‡ Corresponding Author : 이계민 (Gyemin Lee)
E-mail: gyemin@seoultech.ac.kr
Tel: +82-2-970-6416
ORCID: https://orcid.org/0000-0001-6785-8739
※ 이 논문의 연구결과 중 일부는 “IPIU 2019”에서 발표한 바 있음.
※ 이 연구는 서울과학기술대학교 교내연구비의 지원으로 수행되었습니다.
※ This study was supported by the Research Program of Seoul National University of Science and Technology.
· Manuscript received April 30, 2019; Revised July 5, 2019; Accepted July 5, 2019.

Ⅰ. 서 론

최근 Afreeca TV, Kakao TV, Youtube와 같은 개인방송 플랫폼을 보는 사람들이 증가하면서 축구와 야구 같은 스포츠부터 e스포츠까지 이들 플랫폼을 통해 중계하는 경우가 늘어나고 있다. 이와 같은 경기 영상들은 대체로 길이가 길기 때문에 시청자의 편의를 위해 경기의 주최자나 중계자는 하이라이트 영상을 제작하여 제공하기도 한다. 하지만 하이라이트 영상을 제작하기 위해서는 전문적인 편집 기술과 장비가 필요하고 시간과 비용이 많이 소요되는 문제가 있다. 이에 본 논문에서는 영상에서 하이라이트의 위치를 자동으로 예측하는 방법을 제안한다.

그림 1에서 보이는 것과 같이 대다수의 개인방송 플랫폼은 영상과 채팅이 함께 화면에 자리하며 시청자들은 채팅창에서 영상에 대한 의견을 함께 나눌 수 있다. 특히 다수의 시청자가 흥미를 느끼는 부분에서는 채팅창에서도 활발한 의견 교류가 이루어진다. 이러한 경향은 채팅 내역이 하이라이트 예측에 유용할 수 있음을 의미한다. 또한 경기가 진행됨에 따라 해설자와 관중들은 환호하거나 탄식하면서 반응을 하므로 오디오 역시 영상의 흐름을 파악하는데 중요한 단서를 제공한다고 볼 수 있다. 우리는 이러한 점을 이용하여 채팅과 오디오를 하이라이트 예측에 사용하는 방법을 제시한다.

하이라이트를 추출하는데 있어 고려해야할 또 하나의 사항은 콘텐츠의 특성에 따라 다른 흐름을 보일 수 있다는 것이다. 즉, 콘텐츠의 종류에 따라 한 이벤트가 미치는 영향의 시간적 길이는 차이를 보일 수 있다. 예를 들어, e스포츠의 경우는 대게 경기가 빠른 속도로 진행되어 현재 발생한 이벤트의 중요도는 즉각적인 전후관계를 파악함으로 판단할 수 있다. 하지만 축구나 야구와 같은 전통 스포츠 경기에서는 대체로 현재 발생한 이벤트가 득점으로 이어지는지 파악하기 위해서는 중장기적인 흐름을 파악할 필요가 있다. 이에 본 논문은 여러 길이의 시간 정보를 함께 이용하는 다중 시구간 모델을 제안한다. 제안하는 모델은 실제 개인방송 플랫폼에서 중계된 e스포츠와 야구경기 영상을 이용해 평가한다.

Ⅱ. 관련 연구

영상을 요약하거나 하이라이트를 예측하는 방법에 관한 많은 연구가 이루어지고 있다. 대부분의 연구는 영상의 시각적 정보를 이용하는데 초점을 맞추고 있다. [3]은 영상을 짧은 길이의 세그먼트로 구분하고 세그먼트끼리 비교하여 하이라이트에 포함될 점수를 매긴 후, 점수가 높을 세그먼트를 하이라이트로 분류하는 방법을 설명한다. Tang 등은 영상을 클립 단위로 나눈 다음 각 클립이 하이라이트인지 판단하기 위해 low-level 시각적 특징(색상 히스토그램과 HOG)을 기반으로 한 이벤트 통계를 추출하는 방법을 제안하였다[4]. Szegedy 등은 CNN[5]을 이용해 영상으로부터 시각적 특징을 추출한 후 LSTM(Long Short-Term Memory)을 사용해 하이라이트를 찾는 방법을 보였다[6].

한편 Xiong 등은 스포츠 비디오는 특정 장면에서 관객들의 함성이 크다는 특징에 주목하여 음향정보를 이용해 하이라이트를 찾는 방법을 제안한다[7]. 최근에는 자연어 처리 방법을 이용하여 영상을 분석하는 연구도 늘어나고 있다. [8]은 스포츠 영상의 특정 이벤트를 찾기 위해 트위터의 트윗을 이용하는 방법을 설명하였다. 또한 [9]와 [10]은 시간 동기된 코멘트를 이용하여 하이라이트를 검출하였는데, 각각 토핑 모델과 concept-emotion mapping 방법을 사용하여

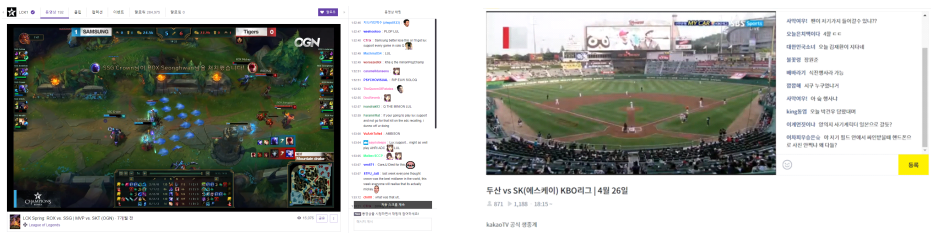

그림 1. 개인방송 플랫폼 구성 (좌: Twitch[1], 우: Kakao TV[2])
Fig. 1. Examples of live streaming platforms (left: Twitch[1], right: Kakao TV[2])

분석하는 방법을 소개한다.

앞장에서 소개한 바와 같이 개인방송의 차별화된 특징 중 하나인 채팅을 이용해서 하이라이트를 찾아낸 연구도 존재한다. [11]은 채팅 트래픽의 변화를 이용하는 방법을 소개하였으며, [12]는 영상의 시각적 정보와 함께 채팅 정보를 이용하여 LSTM을 통해 하이라이트를 찾아내는 방법을 제안하였다. 우리의 모델은 채팅과 오디오를 함께 이용한다는 점에서 시각적 정보에 주로 의존하는 기존의 연구들과 차이를 보인다. 더구나 콘텐츠의 특성에 따른 다중 시구간 정보를 이용하는 방법은 본 논문에서 새롭게 제안하는 방식이다.

Ⅲ. 하이라이트 예측 모델

이 장에서는 원본 영상에서 하이라이트를 예측하기 위해 본 논문에서 제안하는 모델을 설명한다. 채팅이나 오디오 데이터를 이용하여 하이라이트를 예측하는 기본 모델 Single Time Interval Model(STIM)을 제안한다. 그 다음, 영상의 단기적 흐름과 중장기적 흐름을 함께 고려하는 Multiple Time Interval Model(MTIM)을 소개한 후, 채팅과 오디오 정보를 함께 사용하는 방법을 제시한다.

1. 특징 벡터 추출

본 논문에서 제안하는 모델들은 시간과 메모리의 효율적인 사용을 위해 사전에 데이터로부터 특징을 추출한 후 학습한다. 채팅의 경우 FastText[13], 오디오의 경우 MFCC (Mel Frequency Cepstral Coefficient)[14]를 이용할 수 있다.

FastText는 자연어 처리 도구 중 하나인 word2vec[15]을 확장한 모델로 텍스트를 벡터 형태로 표현하며 다양한 언어로 학습된 모델을 제공한다. FastText는 일정한 시간(예, 1초) 내에 등장하는 모든 채팅 내용을 300차원으로 이루어진 하나의 벡터로 표현한다. 본 논문에서는 같은 시간 내에 등장한 채팅의 수에 대한 정보도 함께 이용하기 위해 모든 채팅의 끝에 특정 문자를 추가하여 구분하였다. 이 과정은 다음과 같이 나타낼 수 있다.

$$x_t \leftarrow FastText(chat_t) \tag{1}$$

MFCC는 오디오 인식에서 많이 사용되는 방법으로 소리를 일정 구간으로 나눈 뒤 각 구간에 대해 소리의 스펙트럼을 분석하여 해당 구간에 대한 하나의 특징벡터를 만들어낸다. 예를 들어, 입력받은 오디오 데이터를 1초의 길이를 갖는 구간으로 나눈 후, 각 구간에 대한 소리의 스펙트럼 분석을 한 번씩 진행하여 구간별 20개의 특징을 추출한다. 이 과정은 아래와 같다.

$$x_t \leftarrow MFCC(audio_t) \tag{2}$$

이렇게 추출된 채팅 특징벡터는 해당 구간의 채팅 수와 특정 키워드와 같은 정보를 가지며, 오디오 특징벡터는 해설자나 관중의 소리에 대한 정보를 포함한다.

2. 단일 시구간 모델 STIM

영상에서 한 시점의 이벤트가 하이라이트인지 판단하기 위해서는 영상이 어떻게 진행되는지 그 흐름을 정확히 이

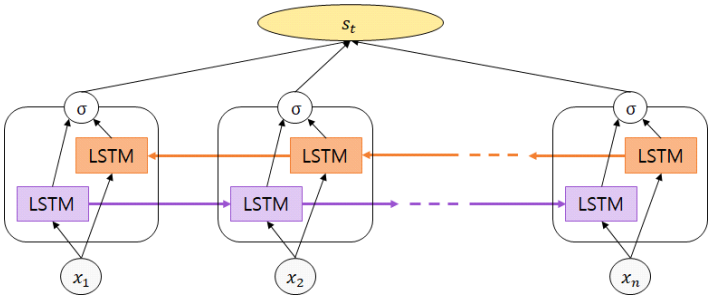

그림 2. 단일 시구간 모델 STIM
Fig. 2. Single Time Interval Model(STIM)

해하는 것이 중요하다. 영상에서 진행 흐름을 파악하기 위해 과거 정보와 현재의 정보를 함께 사용하여 분석하는 LSTM[16]을 이용할 수 있다. LSTM은 현재 이벤트를 분석하기 위해 과거 정보 중 필요한 정보를 선택적으로 사용하기 때문에 영상의 순차적 흐름을 파악하는 데 효과적이다.

하지만 한 시점의 중요도 여부는 미래에 어떠한 영향을 미치는지에 따라 결정되기도 한다. 즉, 과거로부터의 영향을 이해하는 것뿐 아니라 미래 시점에 미치는 영향을 파악할 필요가 있다. 따라서 제안하는 STIM 모델은 입력으로 구간 t의 특징벡터 x_t를 받으면, 양방향 LSTM을 거쳐 h_t^{STIM}을 얻는다. 이 결과를 바탕으로 구간 t에 대한 하이라이트 스코어 s_t가 계산된다. 이를 모든 입력 구간에 대해 반복하면 각 구간에 대한 하이라이트 스코어를 구할 수 있다. 이 과정은 알고리즘 1과 같이 나타낼 수 있다.

Algorithm 1. STIM

Input: feature x_t

1: $h_t^{STIM} \leftarrow BiLSTM(x_t)$

2: $s_t \leftarrow sigmoid(h_t^{STIM})$

Output: highlight score s_t

여기서 BiLSTM은 양방향 LSTM을 의미하며 순방향 LSTM과 역방향 LSTM을 통해 정보를 추출하는 과정이다.

$$h_t^{forward} \leftarrow LSTM_{forward}(x_t) \tag{3}$$

$$h_t^{backward} \leftarrow LSTM_{backward}(x_t) \tag{4}$$

$$h_t^{STIM} \leftarrow [h_t^{forward},\ h_t^{backward}] \tag{5}$$

이 과정을 도식화해보면 그림 2와 같다.

위와 같은 방법으로 얻어진 하이라이트 스코어를 통해 최종적으로 하이라이트 위치를 예측할 수 있다. 최종 하이라이트는 하이라이트 스코어가 상대적으로 높은 구간들을 모아 만들어진다. 따라서 하이라이트 스코어 s_t가 큰 순서대로 해당 구간이 하이라이트로 선택된다. 이때 구간들의 총 길이는 사전에 정한 길이와 같도록 만들어 준다. 이는 다음의 모델에도 모두 동일하게 적용된다.

3. 다중 시구간 모델 MTIM

위에서 제시한 하이라이트 예측 모델은 짧은 시구간에 대한 전후관계만을 파악한다. 하지만 콘텐츠의 종류에 따라 이벤트의 중요도를 판단하기 위해서는 영상의 흐름을 살펴야 하는 경우가 있다. 영상의 흐름이 빠른 경우 단기적 전후관계를 파악하는 것으로 충분할 수 있지만, 영상의 흐름이 느린 경우는 이벤트의 하이라이트 여부를 파악하기 위해 단기적 전후관계와 함께 중장기적 흐름을 파악할 필요가 있다. 이에 우리는 다중 시구간을 이용하여 중장기적 흐름을 이해하는 하이라이트 예측 모델 MTIM을 제안한다.

그림 3은 다중 시구간 모델 MTIM의 구조를 나타낸다. 채팅이나 오디오 데이터로부터 짧은 구간(예, 1초)에 대한 특징벡터 x^{short}를 추출하고 이에 대한 양방향 LSTM 결과인 h^{short}를 구한다. 마찬가지로, 긴 구간(예, 2분)에 대한 특징벡터 x^{long}으로부터 양방향 LSTM의 결과인 h^{long}을 구한다. 그 후, h^{short}의 길이에 맞게 h^{long}을 중복 나열해 h_t^{MTIM}을 만든다. 예를 들어, 짧은 구간을 1초, 긴 구간을 2분으로 설정했을 때, h^{long}을 120번 중복 나열하여 h^{short}와 길이를 맞춘 후 연결한다. 따라서 h_t^{MTIM}은 단기적 흐름에 대한 정보뿐 아니라 중장기적 흐름에 대한 정보를 포함하게 된다. 이후 h_t^{MTIM}을 MLP(Multi Layer Perception)에

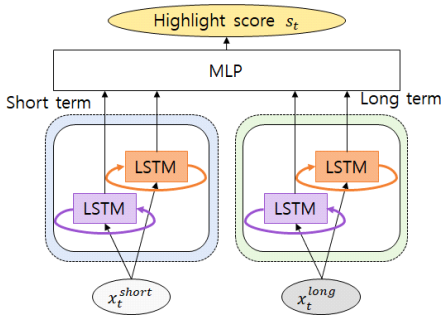

그림 3. 다중 시구간 모델 MTIM
Fig. 3. Multiple Time Interval Model(MTIM)

입력으로 주어 하이라이트 스코어 s_t를 구한다. 이 과정은 알고리즘 2와 같이 나타낼 수 있다.

Algorithm 2. MTIM

Input: short-term feature x_t^{short}, long-term feature x_t^{long}

1: $h_t^{short} \leftarrow BiLSTM(x_t^{short})$
2: $h_t^{long} \leftarrow BiLSTM(x_t^{long})$
3: $h_t^{MTIM} \leftarrow [h_t^{short}, h_t^{long}]$
4: $s_t \leftarrow MLP(h_t^{MTIM})$

Output: highlight score s_t

4. 다중 데이터 이용 모델

앞에서 제안한 모델들은 채팅이나 오디오 중 하나의 데이터만을 하이라이트 예측에 이용한다. 이 절에서는 채팅과 오디오를 같이 사용하는 모델을 제시한다. 그림 4(a)는 다중 데이터를 사용하는 STIM모델을, 그림 4(b)는 다중 데이터를 사용하는 MTIM모델을 보인다. 본 논문에서는 각각을 M-STIM(Multimodal-Single Time Interval Model), M-MTIM(Multimodal-Multiple Time Interval Model)이라 부른다. 각 모델은 STIM과 MTIM을 확장한 형태이다.

그림 4(a)의 채팅 정보를 사용하는 부분과 오디오 정보를 사용하는 부분은 각각 그림 2의 STIM과 유사한 구조를 갖는다. M-STIM은 채팅과 오디오의 양방향 LSTM 결과를 결합 후, 이를 MLP에 입력하여 하이라이트 스코어를 얻는다. 이 과정은 알고리즘 3에 기술되어 있다.

Algorithm 3. M-STIM

Input: chat feature x_t^{chat}, audio feature x_t^{audio}

1: $h_t^{chat} \leftarrow BiLSTM(x_t^{chat})$
2: $h_t^{audio} \leftarrow BiLSTM(x_t^{audio})$
3: $h_t^{M-STIM} \leftarrow [h_t^{chat}, h_t^{audio}]$
4: $s_t \leftarrow MLP(h_t^{M-STIM})$

Output: highlight score s_t

이와 유사하게 그림 4(b)의 short term과 long term을 따로 보면 그림 4(a)와 구조가 동일하다. 다시말해 M-MTIM 모델은 짧은 구간에 대한 채팅과 오디오의 양방향 LSTM 결과와 긴 구간에 대한 채팅과 오디오의 양방향 LSTM 결과를 MTIM에서와 마찬가지 방법으로 연결하고 이를 MLP의 입력으로 하여 하이라이트 스코어를 구한다(알고리즘 4).

Algorithm 4. M-MTIM

Input: short-term chatting feature $x_t^{c-short}$, long-term chatting feature x_t^{c-long}, short-term audio feature $x_t^{a-short}$, long-term audio feature x_t^{a-long}

1: $h_t^{c-short} \leftarrow BiLSTM(x_t^{c-short})$
2: $h_t^{a-short} \leftarrow BiLSTM(x_t^{a-short})$
3: $h_t^{c-long} \leftarrow BiLSTM(x_t^{c-long})$
4: $h_t^{a-long} \leftarrow BiLSTM(x_t^{a-long})$
5: $h_t^{M-MTIM} \leftarrow [h_t^{c-short}, h_t^{a-short}, h_t^{c-long}, h_t^{a-long}]$
6: $s_t \leftarrow MLP(h_t^{M-MTIM})$

Output: highlight score s_t

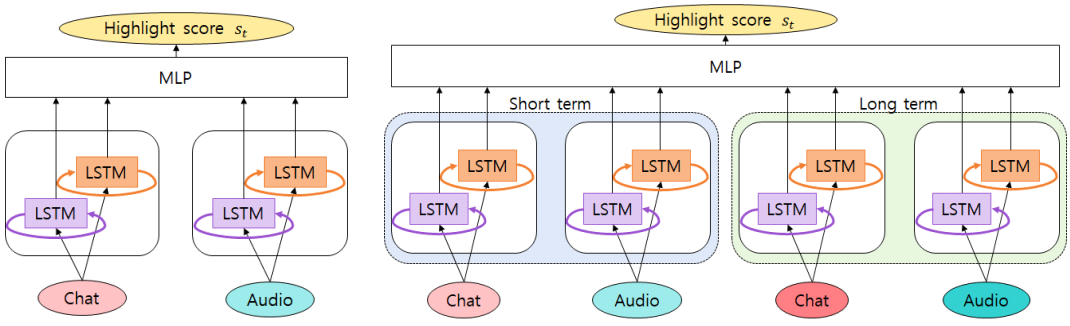

그림 4. 다중 데이터를 이용하는 하이라이트 예측 모델. (a) M-STIM, (b) M-MTIM
Fig. 4. Highlight prediction models using multimodal data. (a) M-STIM, (b) M-MTIM

최종 하이라이트는 앞서 설명한 바와 마찬가지로 하이라이트 스코어 s_t가 높은 순서대로 정해진 길이만큼 선택된다.

Ⅳ. 실험 및 평가

우리는 제안한 하이라이트 예측 모델을 평가하기 위해 e스포츠 경기와 야구경기 데이터를 수집하였다. 각각은 Twitch와 Kakao TV에서 실제로 중계된 영상이며 경기 주최자나 중계자가 제작한 하이라이트 영상이 존재한다. 제안한 모델을 통해 예측한 하이라이트와 제작되어 있는 하이라이트 영상(ground truth)이 유사한 지를 기준으로 성능을 평가하였다.

정량적 평가는 비디오 요약에서 많이 사용되는 F_1 점수를 이용하였다. F_1 점수는 정밀도(precision)와 재현율(recall)의 조화평균으로 이를 수식으로 나타내면 다음과 같다.

$$precision = \frac{\left|H_{gt} \bigcap H_{pred}\right|}{\left|H_{pred}\right|} \times 100 \quad (6)$$

$$recall = \frac{\left|H_{gt} \bigcap H_{pred}\right|}{\left|H_{gt}\right|} \times 100 \quad (7)$$

$$F_1 = \frac{2 \bullet < ision \bullet recall}{< ision + recall} \times 100 \quad (8)$$

여기서 H_{gt}와 H_{pred}는 각각 ground truth와 제안한 모델을 통한 하이라이트 예측 결과이다.

본 논문에서 제안한 하이라이트 예측 모델은 기존의 연구들과 달리 채팅과 오디오만을 이용하여 하이라이트를 예측한다. 따라서 제안한 모델과의 성능비교를 위해 단순 MLP 모델을 구현하였다. MLP 모델은 전후관계에 대한 이해 없이 바로 각 구간에 대한 정보만을 사용하여 하이라이트 스코어를 계산한다. 사용한 MLP 모델은 2개의 은닉층을 가지며, 각 층의 크기는 입력 벡터의 크기와 동일하게 구성하였다.

1. e스포츠 경기 영상 예측

사용한 e스포츠 영상은 2017년에 Twitch에서 중계한 'League of Legends' 대회 5개—LoL 올스타 2017, IEM 월드 챔피언십 카토비체 2017, 2017 LoL 월드 챔피언십, 2017 LoL 챔피언스 코리아 스프링, 2017 LoL 챔피언스 코리아 서머— 중 63개를 사용하였다. 이 가운데 '2017 LoL 챔피언스 코리아 스프링'과 '2017 LoL 챔피언스 코리아 서머'에 해당하는 7개 영상을 이용해 성능을 평가하였다. 사용된 데이터는 모두 e스포츠 전문 방송국인 OGN에서 제작한 하이라이트 영상이 있으며, 이를 ground truth로 하였다. 게임 영상의 길이는 평균적으로 35분이고, 하이라이트 영상은 대체로 원본 영상 길이의 10%이다. 따라서 테스트 영상 길이의 10%를 하이라이트로 검출하였다. 사용한 전체 63개 영상에 대한 정보는 표 1에 요약되어있다. e스포츠 경기는 영상의 길이에 비해 서로 다른 하이라이트 구간 사이의 간격이 짧아 중장기적 흐름보다는 직전과 직후의 흐름

표 1. e스포츠와 야구경기 데이터 요약 정보
Table 1. Summary of e-Sports and baseball datasets

Type	Statistics	Video length (sec)	Total number of chats	Number of chats per second	Length of highlights (sec)	Highlight ratio (%)
e-Sports	mean (±std)	2,096.76 (±599.10)	6,429.49 (±4,216.18)	3.08 (±1.92)	213.27 (±70.99)	10.55 (±3.78)
	max	47,850	14,145	5.96	469	22.30
	min	1,483	2,495	1.22	146	9.84
Baseball	mean (±std)	12,175.39 (±1,176.13)	15,572.32 (±4,216.18)	1.27 (±0.38)	599.25 (±225.34)	4.95 (±1.93)
	max	14,866	24,796	1.98	1,361	12.59
	min	9,909	5,562	0.53	76	0.61

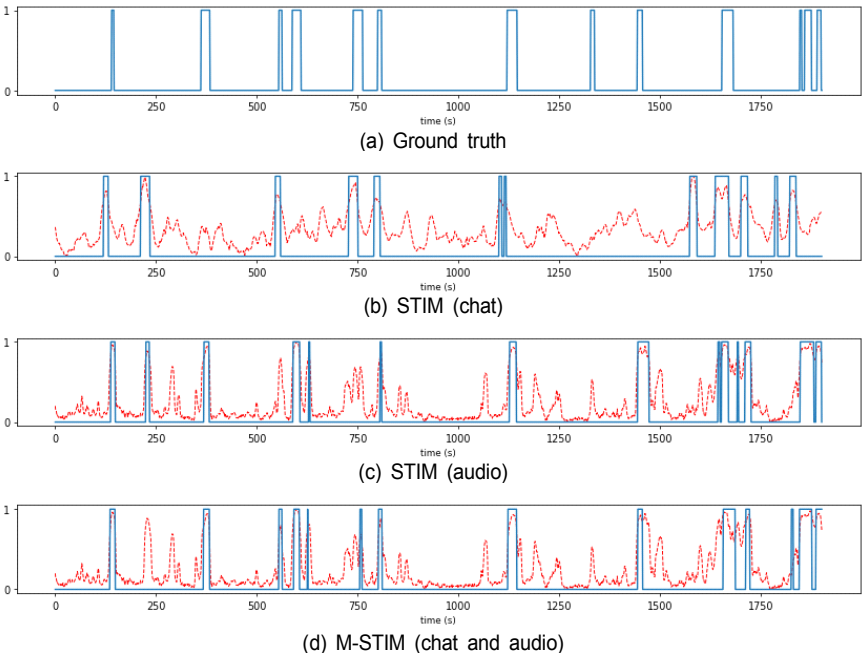

그림 5. e스포츠 영상에 대한 모델별 실험 결과 (파란 실선: 하이라이트 구간 위치, 빨간 점선: 하이라이트 스코어)
Fig. 5. Experiment results on an e-Sports test video (blue: highlight locations, red: highlight score)

을 파악하는 것이 보다 적절하므로 여기에서는 STIM 모델을 주로 고려한다.

그림 5는 평가 영상 하나에서 실제 하이라이트 구간과 제안한 모델로 예측한 하이라이트 구간의 위치를 비교한 결과를 보여준다. 영상의 하이라이트 부분을 1로, 그렇지 않은 부분을 0으로 구분하여 나타내고 있으며 하이라이트 검출 모델의 경우 하이라이트 스코어 s_t를 같이 보여주고 있다.

각각 채팅과 오디오 정보를 이용하여 하이라이트를 예측한 결과를 보여주는 그림 5(b)와 5(c)는 대체로 ground truth인 5(a)와 유사한 형태를 보이지만 230초 부근과 같이 예측에 실패한 부분도 상당히 존재한다. 하지만 채팅과 오디오를 함께 사용하여 하이라이트를 예측하는 모델인 M-STIM의 결과인 그림 5(d)에서는 이러한 부분이 많이 사라져 5(a)와 상당히 유사하게 예측하는 것을 확인할 수 있다.

평가 영상 7개에 대한 실험 결과는 표 2에 나타내었다. 사용한 데이터 종류에 관계없이 모두 제안하는 하이라이트 예측 모델은 MLP 모델보다 높은 성능을 보인다. 이는 우리 모델의 양방향 LSTM이 과거와 미래 정보를 함께 이용함으로써 영상을 보다 잘 이해한다는 것을 보여준다. MLP 모델의 경우, 채팅과 오디오를 함께 이용한 경우가 오디오만을 이용하였을 때에 비해 성능이 낮게 나오는 것을 볼 수 있다. 일반적으로 채팅 정보는 시청자가 영상을 보고 채팅을 입력하고 내용이 표출되기까지 긴 지연시간을 갖는다. 반면 오디오는 해설자나 중계자가 영상을 보고 설명하기까지 상대적으로 짧은 지연시간을 갖는다. 따라서 특정 시점에서의 채팅과 오디오는 서로 다른 내용을 의미하는 경우가 존재한다. 하지만 MLP 모델은 이러한 특성을 전혀 고려하지 않기 때문에 이러한 결과가 나타난 것으로 생각할 수 있다. 이와 달리, 제안하는 하이라이트 예측 모델은 시간 정보를 고려하기 때문에 훨씬 성능이 좋은 것으로 보인다.

단일 시구간 모델인 STIM을 사용하였을 때의 F_1 점수는 채팅과 오디오를 사용하였을 때 각각 44.99와 63.19로 계산되었다. 반면 M-STIM 모델을 이용하여 채팅과 오디오 정보를 함께 이용했을 때는 65.64로 이들보다 향상된 성능을 보였다. 이를 통해 제안한 모델이 하이라이트 예측에 있어 보다 효율적이며, 특히 채팅과 오디오를 함께 활용했을 때 영상에 대한 이해도를 높일 수 있음을 알 수 있다.

표 2. e스포츠 데이터 7개에 대한 실험 결과
Table 2. Experiment results on e-Sports data

Data type	Model	Precision	Recall	F_1
Chat	MLP	12.71	15.59	13.92
	STIM	49.36	41.69	44.99
Audio	MLP	42.63	50.97	46.17
	STIM	69.58	58.44	63.19
Chat + Audio	MLP	33.17	39.39	35.83
	Simple STIM	66.23	55.54	60.09
	M-STIM	71.96	60.90	65.64

채팅과 오디오 정보를 이용하는 또 다른 방법은 채팅과 오디오 특징 벡터를 모델의 입력계층에서 단순 연결(simple STIM)하는 것이다. 반면에 제안하는 M-STIM 모델은 채팅과 오디오 각각에서 양방향 LSTM을 거쳐 추출한 정보를 이용하고 있다. 표 2는 이 두가지 방식에 대한 비교 결과를 보여주고 있다. Simple STIM 모델의 F_1 점수는 60.09, M-STIM 모델은 65.64로 M-STIM 모델을 사용하였을 때 성능이 더 좋았다. 이러한 결과로부터 특징벡터를 입력 계층에서 단순 연결했을 때에 비해 M-STIM 모델이 하이라이트 예측에 도움이 되는 정보를 보다 효율적으로 사용하는 것으로 보인다.

2. 야구경기 영상 예측

우리는 2018년 4월부터 5월 초까지 Kakao TV에서 중계된 한국 프로 야구 경기영상 중 28개를 수집하였다. 이 중 5개 경기 영상을 테스트 데이터로 활용하였으며 ground truth는 네이버 스포츠에서 제작한 하이라이트 영상을 기준으로 하였다. 사용된 야구 영상은 평균 3시간 20분 정도의 길이를 가지며, 하이라이트 영상의 평균 길이는 이의 5% 정도인 10분이다. 위의 게임 데이터에 비해 영상의 길이가 상당히 긴 반면, 하이라이트의 길이는 전체 영상에서 훨씬

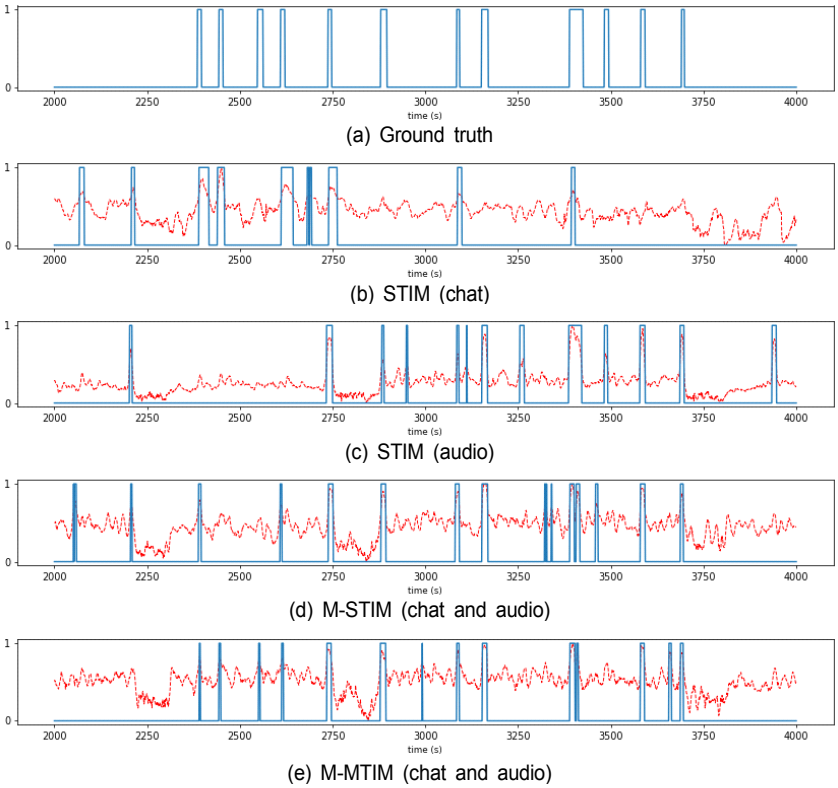

그림 6. 야구 영상에 대한 모델별 실험 결과 (2000~4000초, 파란 실선: 하이라이트 구간 위치, 빨간 점선: 하이라이트 스코어)
Fig. 6. Experiment results on a baseball video (2000~4000sec, blue: locations of highlights, red: highlight score)

적은 비중을 차지하고 있다. 야구 데이터 28개에 대한 정보는 마찬가지로 표 1에 요약하였다.

그림 6은 평가 영상에서 일부(2000~4000초)에 해당하는 구간에서의 실험 결과를 보여준다. 단일 데이터만을 이용한 경우인 그림 6(b)와 6(c)는 2740초나 3090초, 3400초 등에서와 같은 부분을 하이라이트로 잘 예측하지만, 하이라이트인 부분을 잘못 예측하거나 찾아내지 못한 부분도 상당부분 존재한다. 이에 비해 채팅과 오디오를 함께 사용한 M-STIM의 결과인 그림 6(d)는 6(a)와 더 유사하게 하이라이트를 찾아내고 있다. 더 나아가 다중 시구간 모델 M-MTIM을 적용한 결과인 그림 6(e)는 6(c)에서 잘못 예측한 부분에 해당하는 2250초 이전 부분과 3340초 등의 부분을 하이라이트에서 배제함으로써 더욱 ground truth와 유사한 형태를 보여 성능이 향상되었음을 확인할 수 있다.

표 3은 5개의 테스트 영상을 이용한 실험의 평균 결과를 보여주고 있다. e스포츠 데이터를 사용한 실험과 마찬가지로 MLP 모델을 이용하여 제안한 하이라이트 예측 모델과 비교하였다. 표 3에서 MLP 모델의 결과는 이용한 데이터 종류에 관계없이 F_1 점수가 30이하임을 확인할 수 있다. 이는 야구경기 역시 하이라이트 예측이 쉽지 않은 문제임을 보여준다.

단일 시구간 정보만을 이용하는 STIM 모델을 이용하였을 때, 채팅 데이터만을 사용한 경우는 30.59, 오디오 데이터만을 사용한 경우는 45.84의 성능을 보였다. 하지만 M-STIM은 F_1 점수가 47.20으로, 채팅과 오디오 두 종류의 데이터를 모두 사용함으로써 영상을 보다 잘 이해하는 것으로 보인다.

표 3의 결과에서 데이터의 종류에 관계없이 다중 시구간 모델이 단일 시구간 모델보다 성능을 향상시키는 것을 확인할 수 있다. 이러한 결과는 다중 시구간 모델이 중장기 흐름을 함께 봄으로써 단일 구간만을 이용할 때 부족했던 정보를 보완하기 때문이다. 특히 채팅과 오디오를 함께 사용하는 M-MTIM 모델의 결과가 51.48로 가장 성능이 좋았다. 이는 채팅과 오디오 정보를 같이 사용하는 것과 동시에 다중 시구간을 함께 고려하는 것이 하이라이트 예측에 더욱 유용하다는 것을 보여준다.

표 3. 야구 데이터 5개에 대한 실험 결과
Table 3. Experiment results on baseball data

Data type	Model	Precision	Recall	F_1
Chat	MLP	29.79	13.11	18.16
	STIM	29.20	32.32	30.59
	MTIM	30.09	32.74	31.28
Audio	MLP	43.71	19.20	26.60
	STIM	43.23	49.17	45.84
	MTIM	46.25	53.01	49.23
Chat + Audio	MLP	32.56	14.30	19.81
	Simple STIM	41.94	47.42	44.33
	M-STIM	44.48	50.64	47.20
	M-MTIM	48.57	55.20	51.48

표 4. 야구 데이터에서 긴 구간의 길이에 따른 F_1점수
Table 4. F_1scores evaluated for various long-term intervals

Model	Long term interval	Chat	Audio	Chat + Audio
STIM	-	30.59	45.84	47.20
MTIM	1min	20.02	43.63	48.55
	2min	21.32	47.84	51.48
	3min	22.22	44.94	45.88
	4min	19.50	49.23	48.36
	5min	23.26	48.54	46.33
	6min	31.28	47.60	48.86
	7min	20.60	48.20	47.99

3. 구간의 길이에 따른 MTIM의 성능 분석

다중 시구간 모델 MTIM에서는 콘텐츠의 특성에 따라 적절한 구간 길이를 선택하는 것이 중요하다. 따라서 우리는 짧은 구간은 1초로 고정한 후 긴 구간의 길이를 변화시켜가며 구간 길이에 따른 영향을 살펴보았다. 실험은 야구경기 영상에 대해 진행하였으며 이에 대한 결과는 표 4에 정리하였다.

채팅과 오디오 데이터를 함께 이용한 경우, 긴 구간의

길이가 2분일 때의 M-MTIM 모델의 F_1 점수가 가장 높았다. 영상을 직접 확인해본 결과 투수가 공을 한 번 던지고 다음 공을 던지기까지 평균적으로 1분에서 2분정도 걸림을 알 수 있었다. 이는 하이라이트 예측에 있어 하나 또는 두 타석의 정보가 보다 깊이 연관되어 있음을 보여준다.

반면 채팅만 사용했을 때는 긴 구간의 길이가 6분, 오디오만 사용했을 때는 4분일 때 가장 좋은 성능을 보였다. 채팅과 오디오의 내용은 영상 속의 이벤트에 대한 반응인 경우가 많으므로 대체로 영상 내용에 비해 지연되어 나타난다. 또한 이러한 반응은 시간을 두고 계속해서 이어지는 경향이 있으므로 표 4와 같은 결과가 나오는 것으로 볼 수 있다. 특히 채팅의 경우 영상과의 상관관계가 오디오에 비해 적은 것을 확인할 수 있다. 따라서 채팅이나 오디오 정보를 사용하는 경우에 영상과의 지연 시간을 함께 고려하면 유용할 것으로 판단된다.

V. 결 론

본 논문은 영상에서 하이라이트의 위치를 자동으로 예측하기 위해 채팅과 오디오를 이용하는 방법을 제안하였다. 이벤트가 하이라이트인지 판단하기 위해서는 과거 정보뿐 아니라 미래에 어떠한 영향을 끼쳤는지도 중요한 요소이기 때문에 제안하는 모델은 양방향 LSTM을 이용하여 영상을 보다 잘 이해하도록 하였다. 또한 콘텐츠의 특성에 따라 중장기적 흐름을 파악해야 하는 경우, 다중 시구간 정보를 함께 이용하는 것이 영상을 이해하는 데 있어 도움이 된다는 사실을 보였다. 제안한 모델들은 채팅과 오디오를 함께 사용함으로써 하이라이트 예측 성능을 보다 높일 수 있었다.

본 논문에서 제시한 모델은 학습 과정에서 얻은 정보를 단순 연결시켜 하이라이트를 예측하고 있는데, 차후에 결합 방법을 다양하게 모색해 볼 필요가 있다. 또한 데이터 종류에 따라 고려하는 구간의 길이도 다르게 적용한다면 향상된 성능을 보일 것으로 예상된다. 영상에서 채팅과 오디오뿐 아니라 이미지 정보를 함께 사용하면 보다 좋은 성능을 보일 것으로 기대하고 이에 대한 후속연구를 진행 중이다.

참 고 문 헌 (References)

[1] Twitch, https://www.twitch.tv/ (accessed Mar. 08, 2019).

[2] Kakao TV, https://tv.kakao.com/ (accessed Mar. 08, 2019).

[3] M. Sun, A. Farhadi, and S. Seitz, “Ranking Domain -specific Highlights by Analyzing Edited Videos,” European Conference on Computer Vision, Zurich, Switzerland, pp. 708-802, 2014, doi:10.1007/978-3-319-10590-1_51.

[4] H. Tang, V. Kwatra, ME. Sargin, and U. Gargi, "Detecting highlights in sports videos: Cricket as a test case," IEEE International Conference on Multimedia and Expo, Barcelona, Spain, pp. 1‑6, 2011, doi:10.1109/ICME.2011.6012139.

[5] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, “Going deeper with convolutions,” The IEEE Conference on Computer Vision and Pattern Recognition, Boston, Massachusetts, pp. 1-9, 2015, doi: 10.1109/CVPR.2015.7298594 .

[6] K. Zhang, WL. Chao, F. Sha, and K. Grauman, “Video Summarization with Long Short-term Memory,” European Conference on Computer Vision, Amsterdam, Netherlands, pp. 766-782, 2016, doi:10.1007/978-3-319-46478-7_47.

[7] Z. Xiong, R. Radhakrishnan, A. Divakaran, and TS. Huang, “Highlights extraction from sports video based on an audio-visual marker detection framework”, IEEE International Conference on Multimedia and Expo, Amsterdam, Netherlands, pp. 29-32, 2005, doi:10.1109/ICME.2005.1521352.

[8] LC. Hsieh, CW. Lee, TH. Chiu, and W. Hsu, “Live semantic sport highlight detection based on analyzing tweets of twitter,” IEEE International Conference on Multimedia and Expo, Melbourne, Australia, pp. 949-954, 2012, doi:10.1109/ICME.2012.135.

[9] J. Li, Z. Liao, C. Zhang, and J. Wang, “Event Detection on Online Videos using Crowdsourced Time-Sync Comment,” International Conference on Cloud Computing and Big Data, Macau, China, pp. 52-57, 2016, doi:10.1109/CCBD.2016.021.

[10] Q. Ping, C. Chen, “Video Highlights Detection and Summarization with Lag-Calibration based on Concept-Emotion Mapping of Crowd-sourced Time-Sync Comments,” Empirical Methods in Natural Language Processing, Copenhagen, Denmark, pp. 1-11, 2017, doi:10.18653/v1/W17-4501.

[11] E. Kim, G. Lee, "Highlight Detection in Personal Broadcasting by Analysing Chat Traffic : Game Contests as a Test Case," Journal of Broadcast Engineering, Vol.23, No.2, pp.218-226, 2018, doi: http://dx.doi.org/10.5909/JBE.2018.23.2.218 .

[12] CY. Fu, J. Lee, M. Bansal, and AC. Berg, “Video Highlight Prediction Using Audience Chat Reactions,” Empirical Methods in Natural Language Processing, Copenhagen, Denmark, pp. 972-978, 2017.

[13] A. Joulin, E. Grave, P. Bojanowski, and T. Mikolov, “Bag of Tricks for Efficient Text Classification,” European Chapter of the Association for Computational Linguistics, Valencia, Spain, pp. 427-431, 2016, doi:10.18653/v1/E17-2068.

[14] S. Davis, P.Mermelstein, “Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken

Sentences," IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol.28, No.4, pp.357-366, 1980, doi:https://doi.org/10.1109/tassp.1980.1163420.

[15] T. Mikolov, K. Chen, G. Corrado, and J. Dean. "Efficient Estimation of Word Representations in Vector Space," Journal of Biomedical Science and Engineering, Vol.9, No.1, pp.7-16 2016

[16] S. Hochreiter, J. Schmidhuber, "Long short-Term Memory," Neural Computation, Vol.9, No.8, pp.1735-1780, 1997, doi:10.1162/neco.1997.9.8.1735 .

저 자 소 개

김 은 율

- 2017년 : 서울과학기술대학교 전자IT미디어공학과 학사
- 2017년 ~ 현재 : 서울과학기술대학교 나노IT디자인융합대학원 정보통신미디어공학전공 석사과정
- ORCID : https://orcid.org/0000-0001-9023-7834
- 주관심분야 : 머신러닝, 딥러닝, 신호처리

이 계 민

- 2001년 : 서울대학교 전기공학부 학사
- 2007년 : University of Michigan EECS 석사
- 2011년 : University of Michigan EECS 박사
- 2011년 ~ 2012년 : University of MichiganResearch Fellow
- 2013년 ~ 현재 : 서울과학기술대학교 전자IT미디어공학과 부교수
- ORCID : https://orcid.org/0000-0001-6785-8739
- 주관심분야 : 머신러닝, 신호처리, 의료정보학