일반논문 (Regular Paper)
방송공학회논문지 제24권 제5호, 2019년 9월 (JBE Vol. 24, No. 5, September 2019)
https://doi.org/10.5909/JBE.2019.24.5.870
ISSN 2287-9137 (Online) ISSN 1226-7953 (Print)

어텐션 알고리듬 기반 양방향성 LSTM을 이용한 동영상의 압축 표준 예측

김 상 민[a], 박 범 준[a], 정 제 창[a)‡]

Video Compression Standard Prediction using Attention-based Bidirectional LSTM

Sangmin Kim[a], Bumjun Park[a], and Jechang Jeong[a)‡]

요 약

본 논문에서는 어텐션 알고리듬 (attention algorithm) 기반의 양방향성 LSTM (bidirectional long short-term memory; BLSTM) 을 동영상의 압축 표준을 예측하기 위해 사용한다. 자연어 처리 (natural language processing; NLP) 분야에서 순환적 신경망 (recurrent neural networks; RNN) 의 구조를 이용하여 문장의 다음 단어를 예측하거나 의미에 따라 문장을 분류하거나 번역하는 연구들은 계속되어왔고, 이는 챗봇, 음성인식 스피커, 번역 애플리케이션 등으로 상용화되었다. LSTM 은 RNN에서 gradient vanishing problem 을 해결하고자 고안됐고, NLP 분야에서 유용하게 사용되고 있다. 제안한 알고리듬은 BLSTM과 특정 단어에 집중하여 분류할 수 있는 어텐션 알고리듬을 자연어 문장이 아닌 동영상의 비트스트림에 적용해 동영상의 압축 표준을 예측하는 것이 가능하다.

Abstract

In this paper, we propose an Attention-based BLSTM for predicting the video compression standard of a video. Recently, in NLP, many researches have been studied to predict the next word of sentences, classify and translate sentences by their semantics using the structure of RNN, and they were commercialized as chatbots, AI speakers and translator applications, etc. LSTM is designed to solve the gradient vanishing problem in RNN, and is used in NLP. The proposed algorithm makes video compression standard prediction possible by applying BLSTM and Attention algorithm which focuses on the most important word in a sentence to a bitstream of a video, not an sentence of a natural language.

Keyword : Deep Learning, Attention algorithm, LSTM, NLP, Codec

a) 한양대학교 전자컴퓨터통신공학과(Department of Electronics and Computer Engineering, Hanyang University)
‡ Corresponding Author : 정제창(Jechang Jeong)
E-mail: jjeong@hanyang.ac.kr
Tel: +82-2-2220-4370
ORCID: https://orcid.org/0000-0002-3759-3116

※ 이 연구는 방위사업청 및 국방과학연구소의 재원에 의해 설립된 신호정보 특화연구센터 사업의 지원을 받아 수행되었음.
※ This work was supported by the research fund of Signal Intelligence Research Center supervised by Defense Acquisition Program Administration and Agency for Defense Development of Korea.
· Manuscript received July 24, 2019; Revised August 12, 2019; Accepted August 12, 2019.

Ⅰ. 서 론

최근 들어 자연어 처리 (natural language processing; NLP) 분야의 연구 성과는 챗봇, 음성인식 스피커, 번역 애플리케이션 등으로 상용화되어 괄목할만한 성과들을 보여주고 있다. 이러한 트렌드에 따라, 무수한 음성과 문자로 된 시계열 데이터를 순환적 신경망 (recurrent neural network; RNN)[1]에 학습시켜 문장의 다음을 예측하거나, 문장을 의미에 따라 분류하고 번역하는 기능을 부여하는 알고리듬이 많이 제안되어 왔다. 하지만 기존의 RNN의 구조는 출력으로부터 멀어질수록 역전파 (backpropagation)[2,3]가 힘들어지면서 신경망의 학습이 저하되는 현상인 gradient vanishing problem을 발생시키기 쉽다. 이를 해결하기 위해 NLP 분야를 포함한 RNN을 필요로 하는 모든 분야에서는 이를 보완한 LSTM (Long Short-Term Memory)[4] 을 이용한다. 활성화 함수로 tanh를 이용하여 gradient vanishing problem을 발생시켰던 RNN과 다르게, sigmoid와 tanh로 이루어진 신경망을 조합하여 만든 셀들을 나열한 LSTM은 더욱 효율적인 학습이 가능하다.

자연어 처리 알고리듬에서는 각각의 단어의 '의미'를 학습시켜 이를 통해 자연어 처리를 하고자 했다. 이러한 경향은 최근에 문장 전체에서 가장 중요한 역할을 하는 단어의 '위치'를 알아내어 결과를 결정할 때 더욱 비중을 두고자 하는 것으로 발전하였다. 이러한 알고리듬을 어텐션 알고리듬(Attention algorithm) 이라고 부르며, 이는 문장의 분류나 번역에 유용하게 사용되었다[5,6,7,8]. 본 논문에서는 문장의 위치를 확실하게 파악하기 위해 시계열 데이터의 현재 입력의 다음 방향만을 향하는 기존의 LSTM에서 입력의 이전 방향까지 확인할 수 있도록 더욱 개선된 양방향 LSTM (Bidirectional Long Short-Term Memory ;BLSTM)을 어텐션 알고리듬을 기반으로 하여 활용한다.

NLP 분야에서는 인간이 사용하는 언어를 대상으로 그 알고리듬을 사용하는 것이 일반적이다. 그러나 본 논문에서는 위 문단에서 설명한 일련의 알고리듬들을 영상의 비트스트림을 대상으로 사용한다. 인간이 사용하는 언어로 되어 있는 데이터가 26개의 알파벳 혹은 다른 언어의 글자로 되어있듯이, 영상의 비트스트림을 16진법으로 표현하면 0부터 F까지 총 16개의 글자로 되어있는 데이터라고 볼 수가 있다.

현재, 대부분의 동영상은 국제 표준으로 지정된 동영상 압축 표준에 따라 저장되어 임의의 경로로 유포되었을 때 그 정보의 유출을 막는 것이 불가능하다. 이러한 정보를 개인적 혹은 군사적 이유로 암호화 방법을 더하여 보호하고 영상을 재생하고 싶을 때 동영상의 압축 표준을 예측하여 그에 따른 암호를 해독하면 효과적으로 동영상의 정보를 보호할 수 있다.이 점에 주목하여 본 논문에서는 동영상 압축 표준으로 압축 및 저장되어있는 동영상의 비트스트림 및 암호화 된 그것을 분석하여 동영상 압축 표준을 알아내는 알고리듬을 제안한다.

Ⅱ. 관련 이론

1. 양방향 LSTM (BLSTM)

기존의 RNN은 시계열 데이터, 즉 고정된 시간 안에서 일정 시간간격으로 배치된 데이터를 이용하는 데에 사용되어왔다. 시간에 따라 증감하는 데이터를 분석하기 위한 수단이었던 RNN은 최근 들어 시계열 데이터에 속하는 인간의 언어로 된 문장을 다루기 시작했고 현재 상용화된 챗봇, 음성인식 스피커, 번역 애플리케이션 등의 기본이 되고 있다.

그러나 그 이전에 데이터가 많아지고 신경망이 깊어지면서 출력과 멀어질수록 역전파 (backpropagation) 가 힘들어지는 gradient vanishing problem[9,10]에 의해 인공신경망은 침체기를 겪고 있었다. 이러한 문제를 해결하기 위해 구조를 개선한 것이 Hochreiter 등이 제안한 LSTM[4]이다. 기존의 RNN에서는 활성함수를 tanh 하나만 이용하였기 때문에, 신경망을 거치는 과정에서 중간 단계의 출력이 -1과 1 사이로 매핑 되는 과정이 반복되어 gradient vanishing problem을 심화시킨다. LSTM은 활성함수 sigmoid를 포함한 신경망을 추가하여 입출력을 제한하는 게이트의 개념을 통해 이를 방지했다.

한편, RNN과 LSTM 등 시계열 데이터를 분석하는 신경망들은 현재 입력의 다음 방향만을 향하는 단방향적 (unidirectional) 구조를 지니고 있다. RNN 구조를 통해 이루어

졌던 연구들이 현재 입력의 이전 방향을 향하는 LSTM까지 반영하는 양방향적(bidirectional) 구조를 채택함으로써 더욱 좋은 결과를 내는 경우는 이미 이전의 연구들을 통해 증명되어 있다[5,6,11,12]. 따라서 본 논문에서는 BLSTM을 이용하여 영상 비트스트림을 분류한다.

2. 어텐션 알고리듬 (Attention algorithm)

어텐션 알고리듬은 최근 NLP 분야에서 질문 응답, 기계 번역, 음성 인식 등에서 많이 사용되고 있다. 해당 알고리듬이 각광받고 있는 이유는 서론에서 언급하였듯이 문장에서 가장 중요한 위치를 찾을 수 있고 문장의 분류 및 번역에서 좋은 성과를 보이기 때문이다.

Y가 (B)LSTM의 output vector라고 하자. 그렇다면 Y는 문장의 단어의 수를 N개라고 하면, 총 N개의 output을 가지는 vector가 되는 것이다. 각각의 output은 임의의 embedding size d만큼의 성분을 가지는 벡터이다(식 1)[5,13].

$$Y=[y_1\ y_2\ y_3\ \cdots\ y_{N-1}\ y_N],\ Y\in R^{d*N} \tag{1}$$

어텐션 알고리듬의 첫 번째 과정은 해당 Y의 '점수'를 매기는 일이다. Y의 점수를 객관적인 기준으로 매기기 위해 'alignment model'를 사용하며, 이는 논문에 따라 단순히 피드포워드 신경망 (feedforward neural network) 이라고 표현하는 경우도 있고[6,14], 신경망의 trained parameter vector를 w라 할 때 아래[5,7,8]와 같이 표현하기도 한다(식 2).

$$(alignment\ \mathrm{model}) = w^T M = w^T \tanh(Y) \tag{2}$$

어텐션 알고리듬의 두 번째 과정은 첫 번째 과정에서 얻은 '점수'의 벡터를 softmax를 이용하여 총합이 1인 벡터로 바꾸어 준다. 이 벡터를 '어텐션 벡터 (attention vector)' 라고 하며 두 번째 과정을 통해 어텐션 벡터 자체는 가장 중요한 단어 또는 출력일 확률을 시사한다(식 3).

$$\alpha = softmax(w^T M) \tag{3}$$

어텐션 알고리듬의 마지막 과정은 분류를 위한 과정이다. 기존의 output vector Y에 어텐션 벡터 α을 곱해준 다음 활성함수 tanh을 거치면 분류 시 사용되는 Y의 대푯값이 될 수 있다. 이를 활성함수 softmax를 가진 신경망에서 훈련을 시켜주면 문장 S가 원하는 라벨을 가질 확률의 벡터를 얻을 수 있게 된다(식 4). 이 때, 해당 벡터에서 가장 높은 확률을 가지는 라벨이 알고리듬의 목적일 가능성이 가장 높다(식 5).

$$\hat{p}(y|S) = softmax(W^{(S)}(Y\alpha^T) + b^{(S)}) \tag{4}$$

$$\hat{y} = \arg max_y \hat{p}(y|S) \tag{5}$$

3. 동영상 압축 표준의 syntax

본 논문에서는 기존에 NLP 분야에서 문장의 분류에 사용되던 알고리듬들을 영상의 비트스트림을 16진법으로 나타낸 것에 사용한다. 기존에 영상의 비트스트림을 RNN 구조의 신경망에 입력하여 동영상 압축 표준을 분류한 예시는 존재하지만[15], 그 결과는 납득할 만큼 좋지는 못했다. 본 논문에서는 성능 향상을 위해 동영상 압축 표준에 따른 영상의 비트스트림의 구조를 파악하고 이 문제에 접근하였다.

인간의 언어로 된 문장에 비해 영상의 비트스트림은 인간이 동영상 압축 표준을 구별할 수 있는 핵심이 되는 '위치'를 찾는 것이 불가능하다. 그러나 동영상 압축 표준의 비트스트림에는 동영상 압축 표준을 설계하는 과정에서 정한 '스타트 패턴'이 존재한다. 해당 비트들을 읽으면 어떤 동영상 압축 표준인지 알 수 있는 것이 일반적이지만, 본 논문에서는 영상의 비트스트림이 임의의 방법에 의해 변형이 되어도 그 특징을 잡아내어 동영상 압축 표준을 분류할 수 있다는 가설을 세우고 이를 실험 결과에서 입증할 것이다.

본 논문에서는 MPEG-2, H.263, H.264, 총 3개의 동영상 압축 표준을 이용한다. 그림 1과 같이 세 표준 모두 복잡한 구조로 이루어져 있지만 영상의 포괄적인 정보가 담긴 부분은 모두 비트스트림의 앞쪽에 존재한다는 공통점을 이용한다.

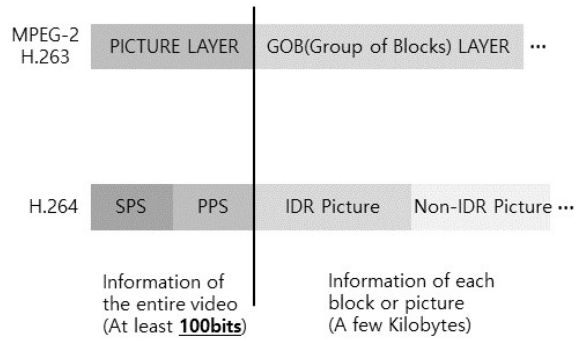


그림 1. MPEG-2, H.263, H.264 영상 비트스트림의 개괄적 구조
Fig. 1. The structure of bitstream of videos compressed by MPEG-2, H.263, H.264

Ⅲ. 제안하는 알고리듬

제안하는 알고리듬은 영상의 비트스트림을 마치 하나의 문장과 같이 여러 단어들로 나눈 다음에[15], Zhang 등이 NLP 분야에서 문장의 분류에 쓰고자 제안한 BLSTM[12]와 Zhou 등이 제안하여 NLP 분야에 본격적으로 사용하기 시작한 어텐션[5] 알고리듬을 결합한 형태이다. 영상의 비트스트림을 분류한 기존의 연구에서는 비트스트림의 구조나 비트스트림이 한 번에 학습되는 양을 고려하지 않았으며, 기존의 RNN 구조를 그대로 사용하였기 때문에 만족스런 결과가 나오지 않았다.

이를 해결하기 위해서 본 논문에서는 영상의 비트스트림을 넣어주는 전처리 과정에서 비트스트림을 일정한 길이만큼 얻은 후 일정 간격마다 띄어쓰기를 해주면서 비트스트림을 여러 단어로 이루어진 문장처럼 취급하였다. 또한, 일정한 길이만큼 얻는 과정에서 이전 문장과 그 다음 문장이 어느 정도 중복될 수 있게 샘플링 함으로써 각 동영상 압축 표준별로 중요한 단어를 어텐션 알고리듬을 통하여 쉽게 찾아낼 수 있도록 하였다.

1. 전처리 과정

영상의 비트스트림을 16진법으로 변환하게 되면 0부터 F까지 총 16가지의 글자로 이루어진 하나의 문자열이 된다. 하나의 문자가 4개의 비트를 의미하기 때문에 두 개의 문자가 1byte의 정보를 갖게 된다. 따라서 본 논문에서는 16진법 이하의 다른 진법으로 표현하는 것보다 한 글자당 많은 정보를 압축하고 있기 때문에 16진법으로 처리하였다.

또한, 관련 이론에서 언급하였듯이 제안하는 알고리듬에서는 각 동영상 압축 표준의 특성을 파악하기 위해 영상의 비트스트림의 전반부를 일련의 문장들로 쪼개어 입력으로 활용한다. 이 때, 그림 2와 같이 이전 문장과 그 다음 문장은 1 byte의 단어 하나를 제외하고는 모두 같은 단어를 가지고 있다. 이는 마치 조금씩 이동하면서 샘플링 하는 것과 같기 때문에 이 과정을 '쉬프트(shift)' 라고 하며, 쉬프트 시 움직이는 byte의 길이를 S라는 변수로 두어 실험 결과에서 활용한다. 기본적으로 $S = 1$이다.

본 논문에서는 전처리 과정에서 한 문장의 길이를 10 byte, 즉 단어의 개수 $N = 10$으로 고정해준 기존의 연구들과 다르게[15], 문장의 길이, 즉 (단어의 개수 = N) * (단어의 길이 = 1 byte) = N byte를 변수로 두어 N의 변화에 따른 분류의 정확도의 변화를 알아볼 것이다.

그리고 임의의 방법으로 변형된 비트스트림을 분류하기 위해 변형된 비트스트림을 대상으로도 실험을 진행한다.

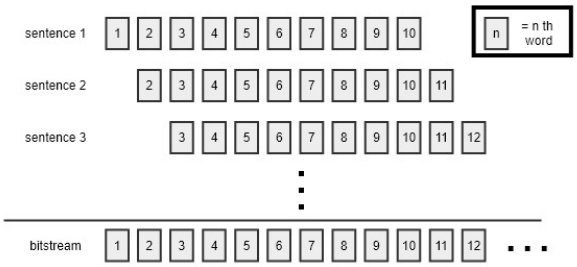


그림 2. 영상의 비트스트림의 데이터 샘플링 (N = 10)
Fig. 2. The data sampling of a bitstream (N = 10)

2. 어텐션 알고리듬 기반 BLSTM

N개의 단어로 샘플링 된 영상의 비트스트림의 문장을 input vector X로 정의한다. 즉, X는 총 N개의 input을 가지는 vector가 된다. 각각의 input은 임의의 embedding size d만큼의 성분을 가지는 벡터이다(식 6)[5,13].

$$X=[x_1\ x_2\ x_3\ \cdots\ x_{N-1}\ x_N], X\in R^{d*N} \quad (6)$$

영상의 비트스트림 입력을 벡터로 embedding하는 과정이 끝나고 나면, 그 벡터를 두 개의 LSTM 신경망에 입력한다. 그림 3과 같이 한 단어 당 하나의 LSTM Cell에 입력되며, 그 결과는 output vector Y이다. 이 때, Y는 순방향 LSTM과 역방향 LSTM의 벡터의 성분 간 덧셈이다. 따라서 Y는 각 단어를 신경망에 넣어서 출력을 낼 때 마다 앞의 단어에 의한 영향과 뒤의 단어의 영향을 동시에 받게 되어 더욱더 객관적인 결과를 얻을 수가 있다.

마지막으로 output vector Y을 대상으로 어텐션 알고리듬을 적용해주면 입력한 문장이 어느 동영상 압축 표준에 의해 압축된 영상의 비트스트림의 일부인지 분류가 가능하다. 신경망을 영상의 비트스트림의 핵심적인 정보가 들어가 있는 전반부를 중심으로 훈련시키면 비트스트림의 전반부를 대상으로 무작위의 순서로 문장을 얻은 뒤에 이를 각각 제안하는 알고리듬을 이용하여 분류하여 가장 많이 도출되는 라벨을 그 비트스트림의 동영상 압축 표준이라고 판별한다.

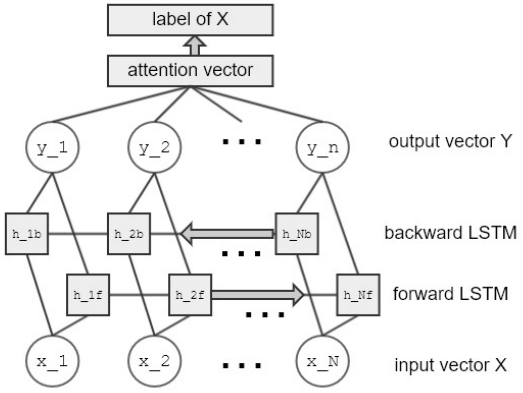


그림 3. 제안하는 알고리듬의 구조
Fig. 3. The structure of proposed algorithm

Ⅳ. 실험 결과

BLSTM을 학습시키기 위해서는 영상의 비트스트림에서 1 byte 크기의 단어 N개를 가진 문장과 그 비트스트림의 동영상 압축 표준을 대응하는 쌍으로 만드는 것이 우선이다.

본 논문에서는 표 1과 같이 총 22개의 영상을 훈련 및 검증 데이터로 사용한다[16,17]. 제안하는 알고리듬의 결과로서 나오는 라벨, 즉 동영상 압축 표준은 MPEG-2, H.263, H.264 3가지 중 하나이다. 따라서 각 영상마다 3가지 동영상 압축 표준에 해당하는 영상을 준비하였다.

표 1. 실험에 사용한 동영상의 목록
Table 1. The list of videos for training and testing

Name	MPEG-2 (.m2v)	H.263 (.h263)	H.264 (.264)
FVDO_Freeway_cif	MPEG-2	H.263	H.264 (level 4)
FVDO_Girl_cif			
FVDO_Golf_cif			
FVDO_Shore_cif			
FVDO_Plane_cif			
FVDO_Freeway_qcif			
FVDO_Girl_qcif			
FVDO_Golf_qcif			
FVDO_Shore_qcif			
FVDO_Plane_qcif			
FVDO_Freeway_4cif			
FVDO_Girl_4cif			
FVDO_Golf_4cif			
FVDO_Shore_4cif			
FVDO_Plane_4cif			
FVDO_Freeway_720p		H.263+	
FVDO_Girl_720p			
FVDO_Golf_720p			
FVDO_Shore_720p			
FVDO_Plane_720p			
akiyo_cif		H.263	H.264 (level 1.3)
news_cif			

이 때, 음성이나 다른 파일 확장자의 정보는 제외하고 순수한 동영상 압축 표준의 비트스트림만 포함하기 위해 영상의 확장자를 각각 .m2v와 .h263과 .264로 정하였다.

H.263과 H.264의 경우에는 동영상 압축 표준의 변형(variation)이 존재하기 때문에 데이터 안에 이를 포함시켰다. H.263은 HD 해상도 이상의 동영상을 대상으로 사용할 수 없기 때문에 이를 보완한 것이 H.263+[18]이다.

H.264의 경우 영상의 비트스트림 초반의 SPS (Sequence Parameter Set) 에서 비트열의 레벨을 정해주는데, level 4에서는 SPS 에서 vui_parameters_present_flag 가 0이고, level 1.3에서는 1이기 때문에 VUI 매개변수(Video Us-

ability Information parameter)의 존재 여부에 차이가 있다[19].

또한, H.264의 경우 동영상 비트스트림 초반의 PPS (Picture Parameter Set)에서 엔트로피 부호화 (entropy coding)의 종류를 정해주는데, 문맥 기반 적응적 이진 산술부호화 방식(Context-Adaptive Binary Arithmetic Coding; CABAC)을 사용할 경우 entropy_coding_mode_flag 가 0이고, 문맥 기반 적응적 가변 길이 부호화 방식(Context-Adaptive Variable Length Coding; CAVLC)을 사용할 경우 1이기 때문에 엔트로피 부호화의 종류에 차이가 있을 수 있다[19]. 실험에 사용한 .264 확장자의 동영상들은 모두 High Profile이기 때문에 두 종류 중 하나를 가질 수 있지만, 본 논문에서는 CABAC를 이용한 동영상만을 이용하였다. 관련 이론에서 H.264 동영상 압축 표준을 가지는 영상은 영상 전체의 정보를 가지고 있는 SPS와 PPS를 주로 이용한다는 가정을 하였고, SPS와 PPS의 경우 가변 길이 부호화에는 지수 골롬 부호화(exponential-Golomb coding)를 이용하기 때문에 영상의 비트스트림 초반부 이후의 엔트로피 부호화의 종류는 실험에 큰 영향이 없을 것이라 판단했다.

본 논문에서는 표 1의 영상 하나당 D개의 문장을 가져오도록 하였다. 한 영상마다 N byte 길이의 문장을 S byte 씩 이동시키면서 D개의 문장을 가져오기 때문에, 신경망에 들어가는 정보의 양은 $S * (D - 1) + N$ byte 이다. 이 값이 커질수록 신경망은 영상의 비트스트림의 후반부까지 입력 데이터로서 보는 것이 가능하다. 기본적으로 $D = 200$, $N = 64$이다.

따라서 신경망을 학습하고 검증하는 데에 총 4400개의 문장을 사용하게 되는데, 실험 시에 90%의 문장을 훈련에 사용하였고 나머지 10%의 문장을 검증에 사용하였다. 또한, 검증에 사용한 440개의 문장을 대상으로 혼동 행렬 (confusion matrix) 을 얻어내고 이를 이용하여 수많은 성능 지표를 얻어내었다. 그 중에서도 분류 관련 연구에 가장 활발하게 쓰이는 정확도 (accuracy) 와 F1 점수 (F1 score)를 표 2에서 정량적 지표로서 사용하였다. 정확도는 전체 문장에 대해 문장의 라벨, 즉 동영상 압축 표준을 정확히 맞춘 문장의 비를 백분율로 나타낸 것이다. F1 점수는 정밀도 (precision) 와 재현율 (recall) 의 조화평균이다. 정밀도는 A가 문장의 라벨이라고 분류한 문장들 중에서 실제로 A가 문장의 라벨인, 즉 정답을 맞힌 문장의 비율이다. 재현율은 실제로 A가 문장의 라벨인 문장들 중에서 A가 문장의 라벨이라고 분류한 문장의 비율이다[20]. F1 점수의 경우 각 동영상 압축 표준별로 도출할 수 있기 때문에 동영상 압축 표준의 수만큼 존재하며, 따라서 3개의 F1 점수를 평균한 값을 실험 결과로 사용하였다.

표 2. 검증 데이터셋에 대한 정확도와 F1 점수
Table 2. Accuracy and F1 score for test dataset

	RNN	LSTM	BLSTM	Att-LSTM	Proposed Algorithm
acc.	96.14	97.20	97.05	99.09	99.39
F1	96.16	97.19	97.04	99.08	99.39

비교하는 알고리듬은 RNN[15], LSTM[4], BLSTM[12], 어텐션 알고리듬 기반 LSTM (Attention-based LSTM; Att-LSTM)[7] 총 4가지이다. 정량적 성능 지표는 표 2와 같다.

전체적으로 높은 정확도와 F1 점수를 가지는 이유는 전처리 과정에서 동영상 압축 표준에 의해 압축된 비트스트림의 구조를 파악하고, 신경망에 입력하는 문장의 길이를 조절해주어 데이터의 특징을 늘렸기 때문이다.

또한, 단방향 신경망을 사용했을 때의 결과와 양방향 신경망을 사용했을 때의 그것은 차이가 거의 없는 것을 볼 수 있다. 이는 영상의 비트스트림이 인간의 언어로 된 문장과는 다르게 문맥을 가지고 있지 않아서 신경망의 방향이 의미가 없음을 시사한다.

하지만 RNN과 LSTM, BLSTM과 제안하는 알고리듬의 결과에는 유의미한 차이가 있다. 전자의 경우는 관련 이론에서 설명한 gradient vanishing problem을 해결하였기 때문이고 후자의 경우는 어텐션 알고리듬의 적용을 통해서 비트스트림의 동영상 압축 표준 예측에 중요한 역할을 하는 단어의 위치를 파악하는 것이 성공했기 때문이다.

이외에 본 논문에서는 영상 하나당 문장의 수 D, 문장을 샘플링 할 때 이동하는 길이 S, 한번 샘플링 할 때 문장의 길이 N을 바꾸면서 제안하는 알고리듬으로 실험해보았다.

표 3. 문장의 개수에 따른 정확도와 F1 점수
Table 3. Accuracy and F1 score for D

D	D = 100	D = 200	D = 400	D = 800
acc.	98.78	99.39	97.95	97.86
F1	98.76	99.39	97.94	97.86

표 3, 표 4와 표 5의 결과가 그것이다. 표 3은 D의 값만을 바꾼 실험 결과이고, 표 4는 S의 값만을 바꾼 실험 결과, 표 5는 N의 값만을 바꾼 실험 결과이다.

표 4. 문장의 샘플링 시 이동하는 길이에 따른 정확도와 F1 점수
Table 4. Accuracy and F1 score for S

S (byte)	S = 8	S = 4	S = 2	S = 1
acc.	75.23	87.42	96.89	99.39
F1	75.04	87.37	96.89	99.39

표 5. 문장의 길이에 따른 정확도와 F1 점수
Table 5. Accuracy and F1 score for N

N (byte)	N = 8	N = 16	N = 32	N = 64
acc.	69.92	81.89	91.59	99.39
F1	69.80	81.97	91.49	99.39

표 3은 D에 따른 성능 지표의 변화를 나타낸 표이다. 관련 이론에서 동영상 압축 표준의 syntax를 분석하는 과정에서, 동영상 압축 표준의 전반부의 구조를 이해하고 전반부 위주의 구조를 학습시키는 것이 더 좋을 결과를 낳을 것이라 기대했었다. 그러나 이를 증명하는 뚜렷한 결과는 나오지 않았고, 비트스트림의 후반부 역시 좋은 훈련 데이터가 될 수 있음을 입증하였다.

표 4는 S에 따른 성능 지표의 변화를 나타낸 것이다. S를 줄여 각 문장에 중복된 단어의 비율을 늘리는 과정은 결과적으로 신경망의 학습을 용이하게 해주었고, 그것이 결과로서 나타났다.

표 5는 N에 따른 성능 지표의 변화를 나타낸다. N을 늘려 학습하는 데이터의 특징 (feature)을 늘리는 과정은 결과적으로 신경망의 학습을 용이하게 해주었다.

마지막으로 영상의 비트스트림을 임의의 방법으로 변형했을 때에도 동영상 압축 표준을 분류할 수 있는 특징이 유지되며, 제안하는 알고리듬이 영상의 변형된 비트스트림의 동영상 압축 표준을 분류할 수 있는지 실험하였다. 영상은 비트별 반전 (bitwise inversion) 으로 변형하였다. ‘01010110’ 이라는 비트스트림이 ‘10101001’로 변형되는 것이 그 예이며, 본 논문에서는 16진법으로 된 비트스트림을 이용하기 때문에 샘플링 된 문장은 변형 후 16진법으로 바꾸는 과정을 거쳤다. 학습 데이터의 수는 기존의 4400개의 문장에서 변형된 문장 4400개의 문장을 추가하였으며 이는 같은 데이터를 다른 방식으로 신경망에 입력하는 일종의 데이터 어그멘테이션(data augmentation)이라고 할 수 있다. 따라서 검증 데이터는 그 10%인 880개이며 그 정량적 지표는 표 6과 같다.

표 6. 변형에 따른 정확도와 F1 점수
Table 6. Accuracy and F1 score for encoded bitstream

Encoded	default	bitwise inversion
acc.	99.39	98.64
F1	99.39	98.64

표 6을 통해, 제안하는 알고리듬은 영상의 비트스트림이 변형되었음에도 불구하고 만족스런 결과를 도출하였음을 보인다.

V. 결 론

본 논문에서는 기존의 NLP 분야에서 문장의 다음 단어 생성, 음성 인식과 문장의 번역 등에 사용되던 알고리듬에 문장의 가장 중요한 위치를 결과에 반영하는 어텐션 알고리듬을 도입하였다. 또한, 이를 동영상의 압축 표준 예측에 사용함으로써 딥러닝 알고리듬이 동영상의 압축 표준을 예측하는 분야에 큰 도움이 될 수 있음을 시사했다. 이전 연구[15]가 한정된 변수로 실험을 진행하고 61% ~ 75%의 전체적으로 낮고 불안정한 정확도를 보인 반면에 본 논문에서는 영상 하나당 샘플링 하는 문장의 수, 샘플링 시 이동하는 길이, 샘플링 시 문장 하나당 길이를 변수로 취급하여 더욱 최적화되고 안정적인 결과가 나오도록 하였다.

또한 영상의 변형된 비트스트림을 대상으로 실험하여 동영상 압축 표준의 특징을 잡아내어 동영상 압축 표준의 예측이 가능함을 입증하였다. 컴퓨터 파일의 확장자마다 본 논문에서 실행한 실험에서의 세 동영상 압축 표준과 같이 특징이 있는 만큼 제안하는 알고리듬은 동영상 압축 표준의 예측 이외에도 손상되거나 버려진 컴퓨터 파일의 확장자를 예측하여 정보의 선순환을 불러오는 순기능을 발휘할 수 있을 것으로도 전망한다.

참 고 문 헌 (References)

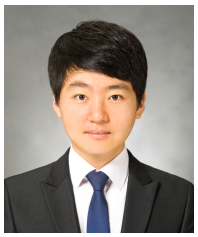
[1] J. L. Elman, “Finding structure in time,” Cognitive science, Vol.14, No.2, pp.179-211, March 1990.
[2] Y. LeCun, “A Theoretical Framework for Back-Propagation,” Proceedings of the 1988 connectionist models summer school, Pittsburgh, Vol.1, pp.21-28, 1988.
[3] F. J. Pineda, “Generalization of back-propagation to recurrent neural networks,” Physical review letters, Vol.59, No.19, pp.2229-2232, November 1987.
[4] S. Hochreiter, and J. Schmidhuber, “Long short-term memory,” Neural computation, Vol.9, No.8, pp.1735-1780, November 1997.
[5] P. Zhou, W. Shi, J. Tian, Z. Qi, B. Li, H. Hao, and B. Xu, “Attention-based bidirectional long short-term memory networks for relation classification,” Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL) (Volume 2: Short Papers), Berlin, Germany, pp. 207-212, 2016.
[6] D. Bahdanau, K. Cho, and Y. Bengio, “Neural machine translation by jointly learning to align and translate,” Proceeding of International Conference on Learning Representations (ICLR), San Diego, pp. 1-15, 2015.
[7] Y. Wang, M. Huang, L. Zhao, and Xiaoyan Zhu, "Attention-based LSTM for aspect-level sentiment classification," Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP), Austin, pp.606-615, 2016.
[8] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy, “Hierarchical Attention Networks for Document Classification,” Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies, San Diego, pp. 1480-1489, 2016.
[9] S. Hochreiter, “The vanishing gradient problem during learning recurrent neural nets and problem solutions,” International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, Vol.6, No.02, pp.107-116, April 1998.
[10] R. Pascanu, T. Mikolov, and Y. Bengio, "On the difficulty of training recurrent neural networks," International conference on machine learning (ICML), Atlanta, pp.1310-1318, 2013.
[11] M. Schuster, and K. K. Paliwal, “Bidirectional recurrent neural networks,” IEEE Transactions on Signal Processing, Vol.45, No.11, pp.2673-2681, November 1997.
[12] S. Zhang, D. Zheng, X. Hu, and M. Yang, “Bidirectional long short-term memory networks for relation classification,” Proceeding of the 29th Pacific Asia conference on language, information and computation (PACLIC), San Diego, pp.73-78, 2015.
[13] Y. Kim, “Convolutional Neural Networks for Sentence Classification,” Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, pp.1746-1751, 2014.
[14] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, “Attention Is All You Need,” Advances in neural information processing systems (NIPS), Long Beach, pp.5998-6008, 2017.
[15] S. Wee, and J. Jeong, “RNN-based bitstream feature extraction method for codec classification,” International Workshop on Advanced Image Technology (IWAIT) 2019, Singapore, Singapore, Vol.11049, p. 110493N, 2019.
[16] Download H.264 High Profile Video streams, http://ftp.arl.mil/~mike/ping/html (accessed Jun. 25, 2019).
[17] Test Sequences encoded in the H.264/MPEG-4 standard, https://pi4.informatik.uni-mannheim.de/~kiess/test_sequences/download/ (accessed Jun. 25, 2019).
[18] T. R. Gardos, “H.263+: THE NEW ITU-T RECOMMENDATION FOR VIDEO CODING AT LOW BIT RATES,” Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP'98 (Cat. No. 98CH36181), Seattle, Vol.6, 1998.
[19] S. Ookubo, H.264/AVC TEXTBOOK, (Translated by Jechang Jeong), HONGRUNG PUBLISHING COMPANY, pp.330-333, 2007.
[20] A. Luque, A. Carrasco, A. Martin, and A. Heras, “The impact of class imbalance in classification performance metrics based on the binary confusion matrix”, Pattern Recognition, Vol.91, pp.216-231, 2019.

저 자 소 개

김 상 민

- 2019년 2월 : 한양대학교 융합전자공학부 학사
- 2019년 3월 ~ 현재 : 한양대학교 전자컴퓨터통신공학과 석사과정
- ORCID : https://orcid.org/0000-0002-1692-0165
- 주관심분야 : 영상처리, 딥 러닝

저 자 소 개



박 범 준

- 2016년 2월 : 한양대학교 융합전자공학부 학사
- 2016년 3월 ~ 현재 : 한양대학교 전자컴퓨터통신공학과 석박사통합과정
- ORCID : https://orcid.org/0000-0003-3783-8272
- 주관심분야 : 영상처리, 딥 러닝

정 제 창

- 1980년 2월 : 서울대학교 전자공학과 학사
- 1982년 2월 : KAIST 전기전자공학과 석사
- 1990년 : 미국 미시간대학 전기공학과 공학박사
- 1980년 ~ 1986년 : KBS 기술연구소 선임연구원 (디지털 및 뉴미디어 연구)
- 1990년 ~ 1991년 : 미국 미시간대학 전자컴퓨터공학부 연구교수 (영상 및 신호처리 연구)
- 1991년 ~ 1995년 : 삼성전자 HDTV 연구개발 담당 수석연구원
- 1995년 ~ 현재 : 한양대학교 융합전자공학부 교수 (영상통신 및 신호처리 연구실)
- ORCID : https://orcid.org/0000-0002-3759-3116
- 주관심분야 : 영상처리, 영상압축