

# fastText와 OpenCV를 이용하여 크리에이터 맞춤 영상자막 수정 방법 연구

최원철<sup>1</sup>, 조세현<sup>2</sup>, 윤동우<sup>3</sup>, 우호진<sup>4</sup>, 김영종\*

<sup>1</sup>숭실대학교 전기공학부

<sup>2,3,4,\*</sup>숭실대학교 소프트웨어학부

e-mail:whj9871@hanmail.net, sybil\_3266@naver.com, osydooo@gmail.com,

whj9871@hanmail.net, youngjong@ssu.ac.kr\*

## A Study on Tools for Creator's Subtitle using fastText and OpenCV

Wonchil Choi<sup>1</sup>, Hojin Woo<sup>2</sup>, DongWoo Yoon<sup>3</sup>, Sehyeon Jo<sup>4</sup>, Youngjong Kim\*

<sup>1</sup>School of Electrical Engineering, Soongsil University

<sup>2,3,4,\*</sup>School of Software, Soongsil University

### 요 약

영상으로 콘텐츠를 개발하는 사람들을 ‘크리에이터’라 칭한다. 이들이 사람들에게 재미를 주고 이목을 끌기 위해 표준어 이외에 다양한 유행어와 신조어들을 만들어내며 이들을 영상뿐만 아니라 자막으로 활용하게 된다. 이러한 자막이 있는 영상 제작시 대부분을 제작하는데 있어 자유도가 높은 크리에이터들의 특성상 맞춤법 오류 및 오타의 문제가 생긴다. 하지만 영상제작 도구에는 맞춤법 검사 기능이 없어 검사를 미리 하기에는 어려운 점이 있다. 우리는 이 문제점을 해결하기 위해 영상을 완성하고 최종 검토를 할 때 맞춤법 검사를 하기 쉽도록 프로그램을 개발한다. OpenCV를 통해 영상의 자막을 글자로서 인식을 하고, fastText 모델을 통해 인식된 글자가 맞춤법에 맞는지 크리에이터에게 제안해주는 맞춤형 프로그램을 개발하고자 한다.

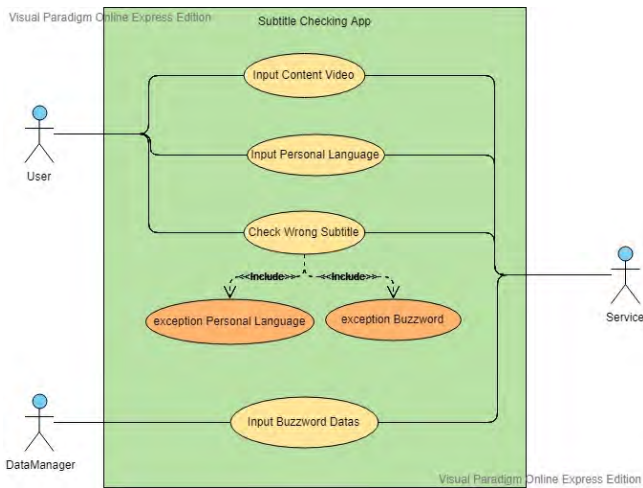
### 1. 서론

영상이 현대인들의 취미생활에 많은 영향을 준만큼 미디어에서 중요한 요소 중 하나가 되었다. 이에 따라서 영상 미디어에 자신만의 콘텐츠를 담아 만드는 사람들도 함께 늘어나고 그들을 “크리에이터”라 부른다. 크리에이터들이 영상 편집을 할 때 사람들이 보기 편하고, 특정 부분을 강조하기 위해서 자막을 주로 사용하게 된다. 이 자막을 미리 대부분을 짜고 영상을 만드는 경우도 있지만, 그렇지 않은 경우도 많기에 크리에이터들이 말하는 것을 받아 적으며 자막을 만드는 일이 많다. 이러한 경우에 맞춤법 및 오타의 문제발생 확률이 높아지고 영상을 다 만든 뒤 재검토를 하는 경우에 이 문제점을 발견하고는 한다. 최근 문서 작업 관련 앱에는 맞춤법 및 오타 검출 기능이 제공이 되지만[1] 영상 제작 앱에는 이러한 기능이 없다. 또한 영상 자막은 일반적인 String 데이터가 아닌 이미지이고 크리에이터 만의 고유 유행어나 신조어들이 있어 기존 맞춤법 검사기로는 해결이 불가능하다. 따라서 크리에이터가 영상 자막 오류를 확인, 수정하는 작업을 하기 위해서는 일일이 재검토를 해야하는 불편함이 생긴다. 우리는 크리에이터들이 겪는 위와 같은 문제점을 해결 하고자 영상을 입력으로 받고, 그 입력에 나오는 자막에 신조어가 사용되었는지, 또는 맞춤법이 틀렸는지, 틀렸다면 고쳐주는 제안을 해주는 프로그램을 연구, 개발하고자 하였다.

### 2. 자막수정 프로그램의 요구사항

사용자에게 제공되어야 할 가장 중요한 기능은 맞춤법 검사이다. 사용자가 만든 영상 자막에 오타는 없는지, 문법적으로 틀린 부분은 없는지 보기 위해 자막수정 프로그램을 사용하게 된다. text형태의 문서를 위한 맞춤법 검사 도구는 이미 기존에 다양하게 존재한다. 그러나 영상콘텐츠는 자막은 문자열(String)이 아닌 이미지 형태이다. 또한 콘텐츠 영상 특성상 사람들의 이목과 재미를 유도하기 위해 표준어에 해당하지 않는 자극적인 단어나 신조어, 유행어 등을 사용하게 된다. 이러한 영상자막의 특성으로 인해 기존 맞춤법 검사기는 해결책이 될 수 없다. 따라서 기존에 없는 영상자막 특성에 맞는 모델을 가진 맞춤법 검사기를 사용해야만 할 것이다.

영상자막 맞춤법 검사 서비스를 제공하기 위해서는 두 가지 핵심적인 기능이 필요하다. 첫번째는 영상속 이미지 형태의 자막을 인식하는 것이고, 두번째는 인식된 자막에서 유행어 및 신조어를 고려하여 오류를 검출하는 것이다. 이러한 서비스를 이용하기 위해 사용자는 제작한 영상과 자신만의 개인 언어만을 제공 하면 된다. 다음 그림은 위와같은 요구사항을 포함한 간단한 usecase이다.



(그림 1) 프로그램 서비스 기반 usecase

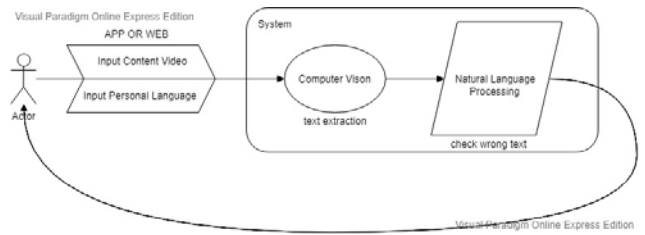
이같은 요구사항을 처리하기 위해 먼저 사용자가 제공한 영상과 개인 언어 데이터를 받을수 있는 Web 또는 App 인터페이스를 제공 할 것이다.

다음으로 인터페이스를 통해 받은 영상에서 이미지 형태의 자막을 인식하기 위해 먼저 영상을 일정 프레임 단위로 나누어서 저장하고 난 후 컴퓨터 비전 기술을 사용하여 처리한다.

사용하고자 하는 도구로는 OpenCV를 사용할 것인데, 이는 먼저 영상 속에서 이미지 글자 영역(text ROI)을 추출해내기 위해 다양한 방법이 존재하나 MSER 또는 Canny Edge Detection을 사용하여 테두리 부분을 추출하고 그 뒤에 불필요한 선을 제거하여 글자를 찾는 방식으로 진행 한다.[2][3] 글자를 찾은 뒤엔 딥러닝모델을 통해 이미지 글자를 문자열(String)형태로 예측해준다.

이제 문자열로 바뀐 영상 자막을 자연어 처리 모델을 이용하여 오타나 맞춤법의 오류가 있는지 검사를 할 수 있다. 학습하는 과정으로는 fastText를 이용하는데, 이는 타겟이 되는 단어 뭉치를 숫자 벡터값으로 변환 한 후 그 단어를 기준으로 앞뒤 문맥으로 나타나는 단어들을 맞추도록 학습한다.[4][5] 이때 사용되는 자연어 처리 모델에는 신조어나 유행어 등이 포함되어야 하므로 기존에 제공하던 한글 데이터에 더불어 전처리 단계를 이용하여 국내 커뮤니티에서의 유행어 신조어 등을 함께 학습시킨다. 이를 통해 개인언어, 유행어, 신조어등 이 고려되는 자연어 처리를 할 수 있고 인터페이스를 통해 사용자에게 틀린 맞춤법 정보를 빠르게 고쳐서 보여준다.

다음 그림은 이같은 과정을 간단하게 도식화 한 그림이다.



(그림 2) 프로그램 작동과정

따라서 사용자가 개인언어와 영상을 제공해주게 되면 입력된 영상을 먼저 일정 단위로 나눈 뒤 위에서 언급한 방법으로 글자를 추출하고, 개인언어가 고려된 검사기를 통해 추출된 글자들을 모두 검사하여 사용자에게 “어떤 프레임의 어떤 자막 내용이 이렇게 고쳐져야 한다”고 결과값을 제공하여 사용자가 자막 수정을 편리하게 진행 할 수 있도록 할 것이다.

ACKNOWLEDGMENT

“본 연구는 과학기술정보통신부 및 정보통신기술진흥센터의 SW중심대학사업의 연구결과로 수행되었음 (2018-0-00209-001)”

참고문헌

[1] 조영환, 김덕봉, 최기선, 김길창. (1990). 한글 맞춤법 오류 검사 및 교정 시스템. 한국정보과학회 학술발표논문집, 17(2), 823-826.

[2] "EAST: An Efficient and Accurate Scene Text Detector" Xinyu Zhou, Cong Yao, He Wen, Yuzhi Wang, Shuchang Zhou, Weiran He, and Jiajun Liang

[3] Neumann L., Matas J. (2011) A Method for Text Localization and Recognition in Real-World Images. In: Kimmel R., Klette R., Sugimoto A. (eds) Computer Vision - ACCV 2010. ACCV 2010. Lecture Notes in Computer Science, vol 6494. Springer, Berlin, Heidelberg

[4] 조현수, 이상구. (2017). FastText를 적용한 한국어 단어 임베딩. 한국정보과학회 학술발표논문집, , 705-707.

[5] Tomas Mikolov, Edouard Grave, Piotr Bojanowski, Christian Puhersch, Armand Joulin. Advances in Pre-Training Distributed Word Representations, 2017