

Analyzing Correlations between Movie Characters Based on Deep Learning

Kyo Jun Jin*, Jong Wook Kim*

*Student, Dept. of Computer Science, Sangmyung University, Seoul, Korea

*Professor, Dept. of Computer Science, Sangmyung University, Seoul, Korea

[Abstract]

Humans are social animals that have gained information or social interaction through dialogue. In conversation, the mood of the word can change depending on the sensibility of one person to another. Relationships between characters in films are essential for understanding stories and lines between characters, but methods to extract this information from films have not been investigated. Therefore, we need a model that automatically analyzes the relationship aspects in the movie. In this paper, we propose a method to analyze the relationship between characters in the movie by utilizing deep learning techniques to measure the emotion of each character pair. The proposed method first extracts main characters from the movie script and finds the dialogue between the main characters. Then, to analyze the relationship between the main characters, it performs a sentiment analysis, weights them according to the positions of the metabolites in the entire time intervals and gathers their scores. Experimental results with real data sets demonstrate that the proposed scheme is able to effectively measure the emotional relationship between the main characters.

▶ **Key words:** natural language processing, sentimental analysis, language style, movie scripts, LIWC

[요 약]

인간은 사회적인 동물로서, 대화로써 정보를 얻거나 사회적인 교류를 해왔다. 대화는 두 사람 이상의 작은 모임에서 서로 말을 편하게 주고받는 것으로, 한 사람이 다른 사람에게 가지는 감성에 따라 그 말의 분위기가 달라질 수 있다. 영화에서 인물들과 인물들이 펼치는 이야기는 중요한 요소로 작용하며, 인물들 간의 관계는 이야기와 인물 간의 대화를 이해하는데 꼭 필요하다. 그러나 이런 정보를 영화에서 자동으로 추출하는 방법은 아직까지 연구되지 않아서 관객들에게 제공되고 있지 못하고 있다. 따라서, 영화 속 양상을 자동으로 분석하는 모델이 필요하다. 본 논문에서는 딥 러닝 기법을 활용하여 각 영화 등장 인물들 간의 감성을 측정하여 영화 속 인물들 간의 관계를 효과적으로 분석하는 방법을 제안한다. 제안 방법은 먼저 영화 대본으로부터 주요 인물들을 추출하고, 주요 인물들 간의 대화를 효과적으로 찾는다. 그런 다음, 주요 인물들 간의 관계를 분석하기 위하여, 감성 분석을 수행하여 전체 시간 간격 내 대사의 위치에 따라 가중치를 부여하고 점수를 수집한다. 또한, 실험데이터를 이용한 실험을 통하여 제안 기법이 효과적으로 영화 등장 인물들 간의 감성을 분석할 수 있음을 보인다.

▶ **주제어:** 자연어 처리, 감성 분석, 언어 스타일, 영화 대본, LIWC

-
- First Author: Kyo Jun Jin, Corresponding Author: Jong Wook Kim
 - *Kyo Jun Jin (ytom2607@gmail.com), Dept. of Computer Science, Sangmyung University
 - *Jong Wook Kim (jkim@smu.ac.kr), Dept. of Computer Science, Sangmyung University
 - Received: 2021. 07. 12, Revised: 2021. 09. 06, Accepted: 2021. 09. 15.

I. Introduction

인간은 사회적인 동물로서, 대화로써 정보를 얻거나 사회적인 교류를 해왔다. 대화는 두 사람 이상의 작은 모임에서 서로 말을 편하게 주고받는 것으로, 한 사람이 다른 사람에게 가지는 감성에 따라 그 말의 분위기가 달라질 수 있다. 좋아하는 사람에게는 긍정적으로 말을 할 확률이 높고, 싫어하는 사람에게는 긍정적으로 말을 할 확률이 그보다 낮은 것이다. 이처럼 사람이 하는 말을 통해 어떤 사물이나 객체를 향한 감성을 알아내는 연구를 감정 분석(sentimental analysis)이라고 한다 [1]. 감성은 감정과 다른 개념으로, ‘화남’이나 ‘놀람’과 같은 인물이나 사건에 대한 직접적인 평가를 할 수 있는 단어보다는 긍정(positive)이나 부정(negative), 중립(neutral)과 같은 단어로 설명될 수 있으며, 감정은 고조(arousal)를 y축으로, 양에 해당하는 수치(valence)를 x축으로 하는 2차원의 평면 안에 표현될 수 있다.

영화에서 인물들과 인물들이 펼치는 이야기는 중요한 요소로 작용한다. 또한 인물들 간의 관계는 이야기와 인물 간의 대사를 이해하는데 꼭 필요하다. 관계가 좋고 나쁨에 따라서, 각 인물이 주고받는 대사는, 좋은 표현, 혹은 나쁜 표현을 쓰더라도 그 안에 있는 의미는 관객에게 다르게 다가갈 수 있다. 사람들은 영화의 표면적인 플롯에서는 드러나지 않는 악연이나 인연을 이해하고 파헤치곤 한다. 관객들이 이야기에서 가장 강하게 연결되는 것은 관계이다. 인물 사이의 관계를 어떻게 상정하느냐에 따라서 관객들은 영화에 대한 이해를 다르게 할 수 있다 [3]. 이렇듯 영화 내 인물들의 관계는 영화를 이해하고 분석하는 데 도움을 준다.

영화나 문학 작품에 대해서 감정 분석으로 영화의 장르나, 분위기를 알아내거나 자연어 생성으로 요약문이나 대본을 쓰는 방법에 대한 연구가 많이 진행되어왔다. 그러나 영화 내의 인물 관계를 심층적으로 분석하는 연구는 진행되어 오고 있지 않아, 관객들은 이러한 정보를 얻을 수 없다.

IMDb (Internet Movie Database)는 세계 최대의 온라인 영화 데이터베이스이다 [4]. 2020년 12월 기준으로 데이터베이스 내 대략 7,500,000편의 영화에 대한 정보가 저장되어 있다. IMDb는 영화 각각의 데이터와 영화의 개봉 정보, 순위, 평론과 별점을 제공하고 있다. IMDb가 제공하는 영화의 정보로는 기본적으로 감독, 작가와 제작진, 배우와 예고편, 사진, 관련 영화와 줄거리, 나아가 박스오피스 성적과 기술정보 등이 있다 (그림 1). 그러나 영화 내 인물들의 관계에 대한 정보는 제공해 주지 않는다.

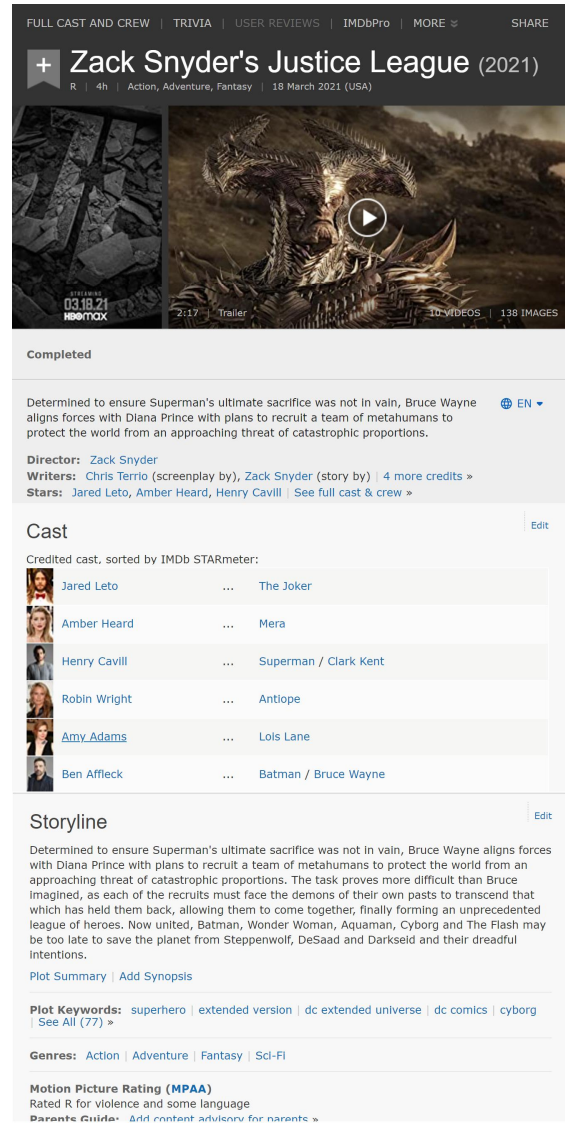


Fig. 1. Information in IMDb

현재 영화 인물들 간의 관계를 얻으려면 영화를 보거나 각본을 읽어 전체 내용을 이해하고 각각의 관계를 평가해야 한다. 복미 내 기준으로 매년 약 700편의 영화가 새로 개봉을 하는데, 사람들이 이 영화들을 보고 관계를 내리기에 현실적인 어려움이 있으며 주관적인 평가가 내려질 수 있다. 그러므로 본 논문은 이전에 제공되지 않던 정보인 영화 속 인물 간 관계를 관객에게 제공하고, 이를 얻는 작업을 자동화하기 위한 기법을 제시한다. 본 논문의 기여는 다음과 같다.

- 먼저 영화 대본으로부터 주요 인물들을 추출하고, 주요 인물들간의 대화를 효과적으로 찾기위한 방법을 제시한다 (IV. 2절).
- 영화의 주요 인물들 간의 답러닝 기반 대화 감정 분석을 통하여, 인물들 간의 상관관계를 구하기 위한 방법을 제시한다 (IV. 3절).

- 또한, 실데이터를 이용하여 제안 기법의 우수성을 평가한다.

본 논문은 다음과 같이 구성되어 있다. 2장에서 본 논문과 연관된 관련 연구에 대해서 설명하고, 3장에서 딥러닝 기반의 영화 인물들 간의 상관관계 분석 기법들을 제안한다. 4장에서는 해당 기법들의 성능을 평가하고, 5장에서 결론을 맺는다.

II. Related Work

영화의 대본에 나타나는 감정과 영화 리뷰에서 표현하는 감정을 합쳐서 영화가 전체적으로 어떠한 감정을 가지고 있는지를 자연어 처리 기술을 통해 알 수 있다. 예를 들어 VADER(Valence Aware Dictionary and sEntiment Reasoner)는 사전 기반의 감정 분류기 모델로, 문장을 입력으로 받아 4가지의 값인 부정, 긍정, 중립, 복합으로 분류한다 [5]. NRC(National Research Council)는 감정과 감성을 분석하는 것으로 문장을 입력으로 받아 8가지 감정(분노, 공포, 걱정, 신뢰, 놀람, 슬픔, 기쁨, 혐오)와 2가지 감정(긍정, 부정)으로 문장을 분류한다 [6]. 영화의 대본으로 영화의 감성을 알게 되면 영화의 적절한 전체적인 분위기가 배우와 각본에 주입이 되고 있다는 것을 확인할 수 있다. 스릴러 영화의 대본에서 굉장히 긍정적인 점수를 보이게 된다면 분노, 공포와 같은 감정보다 놀람, 기대와 같은 감정이 더 많이 나타나게 된다면 대본에 문제가 있다고 생각할 수 있다. 실제로, 스릴러 영화의 대본을 NLTK(Natural Language Toolkit)의 감정 분석 모델을 통한 감정 태깅과 수동으로 태깅된 감정은 분노와 공포 감정에서 높은 일치율을, 그리고 놀람, 기대, 혐오 감정에서는 낮은 일치율을 보임을 확인되었다 [7].

또한 이러한 기술은 영화 추천과 장르 분류 시스템에 사용하기 위해 개발되고 있다. 영화가 내비치는 감정 혹은 감정을 분류하는 작업에 대하여 많은 연구가 진행되고 있다 [8 - 10]. 이를 통해 영화사는 영화에 대한 평가를 분석하여 긍정적인 의견, 부정적인 의견을 분류해 추후 영화에 피드백을 줄 수 있다. 리뷰 문장 내에서 형용사, 부사, 동사를 추출하여 감정 분석을 실시했을 때 단순한 문서 레벨의 감정 분석보다 좀 더 높은 정확도를 보임을 보였다.

다른 연구로, 영화 대본 전체를 이용한 영화의 진행에 따른 감정 분석 연구가 있다 [11]. 사전 기반의 SentiWordNet [12]과 LIWC (linguistic inquiry and word count) [13]를 이용하여 감성을 분석한 것으로, 장

면 5개 단위로 영화 내의 감성을 보여준다. 연구에선 영화의 전체적인 감정 수치의 흐름에 따라 영화를 클러스터링한 결과를 보이며, LOWESS(locally weighted scatterplot smoothing) [14]와 Savitzky Golay Filter [15], Gaussian Smoothing [16]의 기법을 사용해 데이터를 스무딩 하면 보다 영화의 내러티브를 분석하는데 효과적인 데이터를 얻을 수 있음을 밝혔다.

이야기에 대해서 인물 간의 관계를 매칭하는 연구가 있다. 이는 이야기에 해당하는 문자열을 입력하고 NER(Named Entity Recognition)과 CR(Coreference Relation)으로 한 인물이 다른 인물을 부르는 호칭(관계)과 인물의 이름이 포함된 문장들을 저장한 뒤, 해당 문장들을 나이브 베이즈 분류기로 문장들을 호칭별로 분류한다. 이 과정을 통해 인물 간의 관계가 부모-자녀, 친구 등의 관계인지 분류할 수 있으며 평균적으로 85% 이상의 정확도를 가진다 [17].

위의 연구들은, 영화 내 분위기, 영화가 아닌 사람들의 영화에 대한 평가를 분석한다. 특히 마지막 연구는 인물 간의 관계가 나타나나 그 관계가 긍정적인지 부정적인지 알 수 없다는 한계가 있다. 영화의 내용을 분석하는데 영화 속 인물들의 관계 양상을 아는 것은 중요하다. 본 논문은 관련된 세 논문과 다르게, 영화 내 인물들이 서로 어떤 감정이 담긴 말을 주고 받는지를, 또 이를 통해 알 수 있는 인물 간의 관계의 긍, 부정적인 정도를 분석하고자 한다.

III. Preliminaries

1. Background

카멜레온 효과는 무의식적으로 상대방의 자세와 행동, 얼굴 표정이나 다른 기타 행동들을 흉내 내는 것이다 [18]. 예를 들어 대화의 한 참여자가 팔짱을 끼면, 다른 파트너도 주로 무의식적으로 팔짱을 낀다. 이 효과는 언어에서도 발생하여, 억양이나 속도, 피치와 같은 언어적 표현에서부터, 인접하거나 가깝게 등장하는 단어와 비슷한 단어를 쓰는 것까지 다양하게 발생한다. 또한 이러한 특징은 인위적으로 쓰인 대사에서도 동일하게 발생한다는 것이 밝혀졌다 [19].

언어의 스타일은 다양한 지표로 표현될 수 있다. 문장의 중요한 포인트가 어디에 있는가에 따라서 문장의 스타일이 갈릴 수 있다. 예를 들어, “어제 종이를 몇 장 썼지?”라는 문장에 올 수 있는 문장인 “10장 썼습니다.”라는 문장에선 둘 다 양을 가리키는 ‘몇’과 실제 수인 ‘10’이라는 숫자가 쓰였다. 억양, 속도 등의 자료를 이용할 수 없는 텍

스트 데이터에서는 이러한 단어의 사용들로 문장의 스타일을 나타낼 수 있다.

LIWC는 텍스트 내에 사용된 단어의 카테고리를 분류하여 각 카테고리 별로 사용된 단어의 수와 문장 내 비율을 계산하는 텍스트 분석 프로그램이다 [20]. LIWC가 사용하는 카테고리로는 영어의 각 품사, 그리고 나, 우리, 너, 그녀/그, 그들을 가리키는 단어, 수량과 같은 말 하는 스타일에 관련된 카테고리라 분노, 슬픔과 같은 감정, 가족, 친구, 죽음, 성과 같은 대화 주제 등이 있다. LIWC는 정적인 방법으로, 각 카테고리에 해당하는 단어들에 대한 사전이 존재한다. 문장 내에서 그 단어가 감지되면, 그 단어가 속한 카테고리의 카운트를 하나씩 올리는 방법으로 문장을 분석한다.

2. Data Collection

본 연구를 위하여 IMSDb(Internet Movie Script Database) [21]에서 총 50편에 대한 대본 데이터를 수집하였다. 영화는 Rotten Tomatoes와 IMDB의 가장 평점수가 많은 영화들을 참고하였으며, 가족, 애니메이션 장르의 영화를 선정하였다. 특히 이 중 한국 박스 오피스 성적을 참고하여, 최대한 가까운 시기에 개봉하고 성적이 높은 영화들을 우선순위로 선정하였다. 영화 대본 데이터는 IMSDb 내의 영화들을 크롤링 기법을 사용하여 얻었다.

파이썬 라이브러리인 BeautifulSoup [22]를 이용하여, 대본 페이지의 html 구조를 분석 후 대본에서 제목과 작가, 배경 설명이나 지문을 제외한 인물과 대사만을 크롤링하였다. BeautifulSoup 라이브러리는 HTML 문서를 파싱하여 사용자가 원하는 태그, 그리고 속성별로 데이터를 얻을 수 있게 해준다. 대본에선 인물은 bold 체, 그리고 bold 체 바로 다음 줄에 대사가 나온다. 이 규칙으로 크롤링을 하였으나, 몇몇 이 규칙을 따르지 않는 부분은 수동으로 제거해주었다. 영화 당 대사의 양은 평균적으로 700건을 기록하였다. 이후 각 대사에서 불필요한 기호들을 정규식을 통해 제거하였다.

IV. Proposed Method

- LIWC 분석: 한 인물이 대사를 어떤 인물에게 건넸는지를 파악하기 위하여 카멜레온 효과를 적용해 각 대사의 언어 스타일을 얻는다. LIWC 분석은 각 대사의 언어 스타일을 분석하는 과정에 해당한다.
- 인물쌍 추출: 인물쌍이란 어떤 대사를 말한 이와 이를 듣는 이의 쌍을 의미한다. 이 단계는 LIWC 분석

결과를 기반으로 인물쌍을 추출한다.

- 감성 점수 계산: LIWC 분석의 결과로 대사를 각 인물 쌍에 매칭한 후, 인물쌍의 감성 점수를 계산한다. 감성 점수는 0 이상 1 이하의 값으로, 클수록 긍정적임을 나타낸다.

그림 2는 영화 내 인물 간의 관계를 자동으로 평가하기 위한 제안 기법의 전체적인 흐름을 나타내고 있다.



Fig. 2. An overview of the proposed method

1. LIWC Analysis

본 논문의 제안 기법은 대본 내의 배경, 장소에 대한 설명, 지문 등을 제외한 인물의 대사만을 데이터로 하여 각 대사의 LIWC 분석을 시행한다. LIWC 분석으로 각 대사의 언어 스타일, 주제 등을 얻을 수 있으며, 각 대사는 93개의 지표로 분석된다 [23]. 그림 3은 영화 <해리 포터와 마법사의 돌>의 LIWC 분석의 예시 화면을 보여준다.

WC	Analytic	Clout	Authentic	Tone	WPS	Sixtr	Dic	function	pronoun	ppron	i	we	you	shehe
11	1.00	81.84	1.00	25.77	11.00	18.18	90.91	72.73	27.27	18.18	9.09	0.00	9.09	0.00
9	93.26	86.68	4.97	99.00	4.50	44.44	77.78	22.22	0.00	0.00	0.00	0.00	0.00	0.00
9	18.82	13.32	4.97	1.00	4.50	11.11	100.00	55.56	11.11	11.11	11.11	0.00	0.00	0.00
3	93.26	99.00	1.00	25.77	3.00	0.00	100.00	66.67	0.00	0.00	0.00	0.00	0.00	0.00
4	1.92	99.00	1.00	25.77	4.00	25.00	75.00	50.00	25.00	25.00	0.00	0.00	0.00	25.00
14	48.63	98.33	13.15	99.00	14.00	14.29	92.86	64.29	28.57	7.14	0.00	0.00	7.14	0.00
9	46.37	13.32	4.97	99.00	9.00	11.11	88.89	44.44	22.22	22.22	22.22	0.00	0.00	0.00
5	93.26	97.69	1.00	25.77	2.50	80.00	60.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
5	8.89	2.31	1.00	25.77	5.00	20.00	80.00	40.00	20.00	20.00	20.00	0.00	0.00	0.00
22	32.58	91.35	61.34	94.75	4.40	4.55	90.91	50.00	13.64	13.64	0.00	4.55	4.55	4.55
27	7.88	93.07	58.07	25.77	6.75	14.81	92.59	55.56	25.93	18.52	3.70	0.00	3.70	3.70
5	8.89	99.00	1.00	25.77	5.00	0.00	100.00	80.00	20.00	20.00	0.00	0.00	0.00	20.00
18	1.00	99.00	4.97	25.77	9.00	0.00	100.00	66.67	22.22	11.11	0.00	5.56	0.00	5.56
14	83.81	92.33	52.86	99.00	4.67	14.29	100.00	64.29	21.43	14.29	0.00	0.00	0.00	14.29

Fig. 3. Result of LIWC analysis

문장의 요소 중 관사, 조동사, 접속사, 부사, 비인칭 대명사, 부정사, 대명사, 전치사, 수량의 지표를 이용하여 대사의 언어 스타일을 표현하였다 [24]. 이를 이용하여 표현하는 방법은 다음과 같다.

대본 내에 모든 대사의 집합을 $L_{all} = \{L_1, L_2, \dots, L_{|L|}\}$ 이라 가정하자. 또한, 문장의 요소를 I 라고 하였을 때, 제안 기법은 다음과 같은 8개의 문장 요소를 사용한다: $I_{article}$ (관사), I_{adverb} (부사), I_{conj} (접속사), $I_{pronoun}$ (대명사), I_{negate} (부정사), I_{ppron} (고유명사), I_{prep} (전치사), I_{quant} (수량). 이때, 각각의 대사는 위의 문장 요소들을 사용하여 벡터 형태로 표현할 수 있다. 즉, i 번째 문장 L_i 는 벡터

$V_i = [I_{\text{article}}, I_{\text{adverb}}, I_{\text{conj}}, I_{\text{pronoun}}, I_{\text{ate}}, I_{\text{ppron}}, I_1, I_{\text{quant}}]$ 로 표현된다. 이때, 각 문장 요소의 값은 LIWC 분석을 통해 구하며, 최소 0부터 최대 100까지의 값을 가진다.

2. Retrieving Character Pairs

제안 기법은 대본 내에 등장하는 고유한 인물들의 대사들을 구한 후, 주연으로 추정되는 인물의 대사만을 고려한다. 이를 위해 해당 대사가 전체 대사의 10% 이상이 되는 인물들의 대사만을 사용한다. 영화 내 주연의 집합을 C 라고 할 때, 집합 C 는 $C = \{C_1, C_2, C_3, \dots, C_{|C|}\}$ 로 표현할 수 있다. 이때 $a \neq b$ 이고 $a, b \leq |C|$ 인 a, b 에 대하여 $C_a \rightarrow C_b$ 를 a 와 b 의 인물쌍이라고 정의한다. 이때, $C_a \rightarrow C_b$ 와 $C_b \rightarrow C_a$ 는 서로 다른 인물쌍이다. A가 B를 향해 가지는 감성과 B가 A를 향해 가지는 감성이 다를 수 있기 때문이다. 대본에서 한 인물의 대사가 어떤 인물을 향한 대사인지를 파악하기 위하여, 발화된 위치와 근접한 다른 인물들의 대사 중 가장 스타일이 비슷한 대사를 선택해 해당 대사들의 화자쌍과 매칭한다. 예를 들어 C_a 가 말한 대사와 C_b 가 말한 대사의 LIWC 분석 결과 스타일이 비슷하다면, 해당 대사는 각각 $C_a \rightarrow C_b$, $C_b \rightarrow C_a$ 인물쌍에 매칭된다. 3단어 이하의 대사는 인물쌍 매칭 대상에서 제외된다.

또한 모든 대사에 대하여 인물쌍을 얻는 과정에서 대본에서 대사가 발화된 구간을 추출한다. 구간은 영화가 진행됨에 따라 새로운 관계의 양상이 드러나거나, 중요한 대사가 발화되는 점을 고려하여 대사의 위치에 따라 대사의 감성에 부가되는 가중치를 다르게 만들기 위함이다. 먼저 영화의 대본을 시간 순서대로 m 개의 구간으로 균등하게 나눈다고 가정한다. 또한 t_0 은 영화의 시작 시점, t_m 은 영화의 종료 시점에 해당한다고 가정한다. 이때, i 번째 구간은 $[t_{i-1}, t_i]$ 로 표현될 수 있다.

또한, j 번째 대사 L_j 의 구간을 T_j 라 가정하자. 이때, T_j 는 $\lceil j \frac{m}{|L|} \rceil$ 에 해당한다. 예를 들어 어떤 대사가

3512번째로 발화됐고, 전체 대사 수가 7,503건이며, 5개의 구간(즉, $m = 5$)이 존재한다고 가정하자. 이때, L_{3512} 의 구간은 $T_{3512} = \lceil 3512 \times \frac{5}{7503} \rceil = 3$ 번째에 해당한다. 예를들어, 표 1은 영화 <겨울 왕국>의 대사 중 일부이다. 표 2는 표 1의 각 대사에 대해 앞에서 설명한 과정에 따라 각 대사의 T, V 값을 구한 결과를 보여주고 있다 ($m = 3, |L| = 831$).

Table 1. Example of script lines

i	화자	대사 (L_i)
480	Anna	Wow. Elsa, you look different...
481	Elsa	Thank you. I never knew...
482	Anna	I'm so sorry about what happened...
483	Elsa	No, no. It's okay...

Table 2. Example of script lines with their vectors

i	T_i	V_i
480	2	[7.14, 0, 7.14, 28.57, 0, 7.14, 0, 0]
481	2	[0, 10, 0, 40, 10, 30, 10, 0]
482	2	[0, 20, 20, 30, 0, 20, 10, 0]
483	2	[0, 6.67, 6.67, 20, 20, 13.33, 6.67, 0]

다음 단계에서는 각각의 인물쌍이 주고받은 대사들의 집합을 구한다. 그림 4는 전체적인 과정을 간략하게 보여주고 있다.

인물쌍 $C_a \rightarrow C_b$ 에 해당하는 대사들의 집합을 $L_{a \rightarrow b}$ 라 가정하자 (단, $L_{a \rightarrow b} \subset L_{all}$). $L_{a \rightarrow b}$ 를 구하기 위해 본 논문은 카멜레온 효과(즉, 두 인물이 주고받는 대사의 언어 스타일은 유사함)를 이용하였다. 카멜레온 효과를 이용하여 인물쌍 간의 대사를 구하기 위해, 각 대사에서 IV.1절에서 설명한 벡터 V 를 구한다. 이하에서 대사 L_a 와 L_b 간의 코사인 유사도를 구한다 함은 V_a 와 V_b 간의 코사인 유사도를 구한다는 것과 같다. L_n 을 C_a 가 말한 대사라 가정하자. 또한, $L_{n+1}, L_{n+2}, \dots, L_{n+5}$ 는 L_n 으로부터

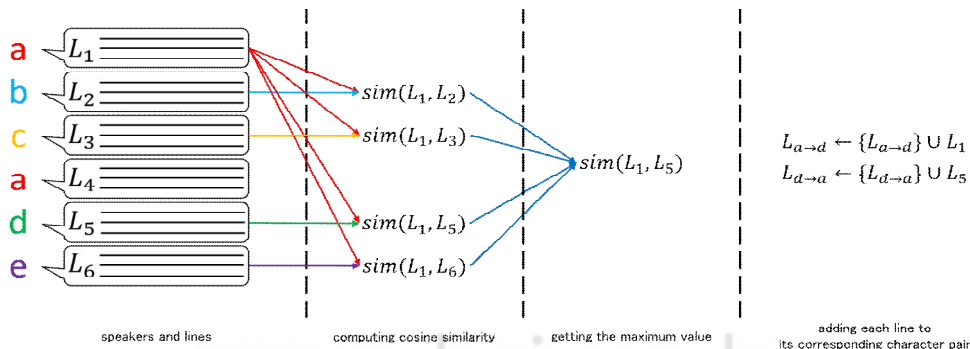


Fig. 4. Process of retrieving character pair

C_a 가 말한 대사를 제외한 다음 5개의 대사라 가정한다. 이때, L_n 과 다음 5개 각각의 대사와의 코사인 유사도를 구한다. 이중 유사도가 0.5를 넘으면서, 가장 높은 유사도를 가지는 대사의 화자를 C_b 라 가정하자. 그런 다음, 대사 L_n 은 L_{a-b} 에, L_n 과 가장 높은 유사도를 가지는 대사는 L_{b-a} 에 각각 추가된다.

그림 4에서 a, b, c, d는 등장인물을 나타내며, L_1 부터 L_6 은 각각의 등장인물이 말한 대사를 의미한다. L_1 과 L_2, L_3, L_5, L_6 의 코사인 유사도를 구한다. (L_4 은 화자가 같으므로 제외된다.) 이때 L_1 과 L_5 의 유사도가 가장 높으면서 0.5를 넘었다고 가정하자. 이 경우, a(즉, C_a)가 말한 대사 L_1 는 L_{a-d} 에, d(즉, C_d)가 말한 대사 L_5 는 L_{d-a} 에 각각 추가된다. 대사 L_1 에 대한 분류가 끝났으므로, 그 다음 대사 L_2 에 대하여 같은 과정을 반복한다. 이러한 과정은 모든 대사를 분류할 때까지 반복된다. 예를 들어, 표 1과 표 2의 예제를 이용하여 V_{480} 과 V_{481} , V_{482} 와 V_{483} 의 코사인 유사도를 구했을 때, 각각 0.8, 0.2이라는 수치가 나왔다고 가정한다. 이때, $L_{Anna} \rightarrow L_{Elsa}$ 에 L_{480} 이, $L_{Elsa} \rightarrow L_{Anna}$ 엔 L_{481} 가 추가된다. 반면 L_{482}, L_{483} 의 경우엔 0.5를 넘지 않았으므로 추가되지 않는다.

3. Computing Sentiment Scores

각 인물쌍 내 대사들은 각각 감성 분석을 거쳐서 0부터 1까지의 감성 점수로 계산된다. 감성 점수가 1에 가까울수록 긍정적임을 의미하며, 0에 가까울수록 부정적임을 의미한다. 또한, i 번째 대사 $L_i \in L$ 의 감성 점수를 s_i 라 가정한다. 감성 점수 s_i 는 LSTM (Long short-term memory) 모델로 구현된 감성 분류기를 사용하여 구한다. 본 논문에서 사용된 모델은 Kaggle의 영화 리뷰 감성 분석 데이터셋 [25]을 이용하여 훈련하였고, 각 문장의 연속되는 단어들을 각각의 입력으로 하여 해당 문장의 감성 점수 (0- 부정적, 0.25- 약간 부정적, 0.5- 중립, 0.75- 약간 긍정적, 1- 긍정적)를 출력한다. 모델은 정확도 97%의 성능을 보이고 있다. 표 3은 모델 훈련에 사용된 하이퍼파라미터들이다.

Table 3. Hyperparameters for LSTM model used in the paper

Input size	100
Output size	5
Hidden size	128
Batch size	256
Epochs	1000
Optimizer	Adam
Learning rate	0.005
Activation function	ELU

마지막 단계에서는 각 대사의 감성 점수를 통합하여 인물쌍의 감성 점수를 계산한다. 인물쌍의 감성 점수를 계산하기 위하여 앞에서 구한 각 대사의 감성 점수에 대사가 속한 구간의 가중치를 적용한다. 즉, 각 대사의 해당 감성 점수와 구간별 가중치를 곱하고, 이들의 평균을 해당 인물쌍의 감성 점수로 정한다. $w(x)$ 를 가중치 함수라 가정하자. i 번째 대사 L_i 의 구간 T_i 가 주어졌을 때, 이 구간의 가중치를 구하기 위해 본 연구에서는 다음과 같은 3가지 종류를 가중치 함수를 사용한다: $w(T_i) = 1$, $w(T_i) = T_i$, $w(T_i) = T_i^2$. $w(T_i) = T_i$, $w(T_i) = T_i^2$ 은 대사의 가중치가 영화가 진행될수록 증가함을 의미한다. 특히, $w(T_i) = T_i^2$ 의 경우 후반부로 갈수록 전반부보다 대사의 중요성이 더 강조된다. 영화나 소설과 같은 창작물은 영화의 시작부터 후반까지 계속해서 이야기가 고조되고 중요해진다. 가중치를 주지 않았을 때, 일부 후반부에 악역임을 밝히는 등장인물과 선역의 관계의 감성 점수가 부정확하게 측정되었다. 인물쌍의 감성 점수는 해당 인물쌍의 대사의 감성 점수와 가중치의 곱들의 평균이다.

인물쌍 $C_a \rightarrow C_b$ 에 해당하는 대사들의 집합이

$L_{a-b} = \{L_1, L_2, \dots, L_a\}$ 인 경우, 해당 감성 점수, s 는 식 (1)로 계산된다.

$$s = \frac{1}{n} \times \sum_{i=1}^n \left(s_i \times \frac{w(T_i)}{W} \right) \quad (1)$$

이때, W 는 각 가중치 식에서 가능한 가중치 값의 총합을 의미하여 식 (2)로 계산된다.

$$W = \sum_{i=1}^m w(i) \quad (2)$$

표 4는 영화 <겨울 왕국>의 결과 중 일부이다. 식 $m = 3$, $w(T_i) = T_i$ 의 가중치 함수를 사용하였다.

Table 4. Result of a film <Frozen> analysis

pairs	result
Anna -> Elsa	0.226424
Elsa -> Anna	0.178713
Elsa -> Hans	0.124833
Hans -> Anna	0.187306

V. Experiment

1. Experiment Setup

본 연구에서 제안하는 영화 인물들 간의 상관관계 분석 기법을 평가하기 위해, 실제 데이터를 이용하여 실험을 진행하였다. 실험에 사용된 데이터는 IMSDb를 크롤링하여 얻은 영화 대본 50개를 사용하였다. 실험에선 H/W로는 Ryzen 2700X, GTX1070, S/W로는 운영체제에 Ubuntu 20.04 LTS, 감성 분석에 PyTorch 1.7.0, CUDA 10.2 버전을 사용하였다. 실험 1에서는 IV. 2 에서 제안한 인물쌍을 얻는 기법에 대해서 실험하며, 실험 2에서는 IV. 3에서 제안한 감성 점수를 계산하는 기법들을 실험해본다. 실험 2에서는 두 개의 가중치 함수 $w(T_i) = T_i$, $w(T_i) = T_i^2$ 을 사용하였으며, 영화의 구간 $m = 3$, $m = 5$ 를 사용하였다. 그러므로 이들로부터 다음과 같이 총 4개의 조합이 가능하여, 설명의 편의성을 위해 이들을 순서대로 M_1, M_2, M_3, M_4 라 한다.

- $M_1: (w(T_i) = T_i, m = 3)$
- $M_2: (w(T_i) = T_i^2, m = 3)$
- $M_3: (w(T_i) = T_i, m = 5)$
- $M_4: (w(T_i) = T_i^2, m = 5)$

제안 기법의 성능 평가를 위해 다음과 같이 사용자 평가 (user study)를 진행하였다. 5명의 사용자에게 영화 대본 50개의 방법별 결과를 나누어주고 인물쌍 별 감성 점수를 방법별로 1부터 5점까지 평가하게 하였다. 실험 결과 평가에 사용된 점수의 기준은 다음과 같다. 1은 매우 상반되는 감성 점수 (예를 들어, 연인 사이인데 점수가 가장 낮게 나온 경우), 2는 다소 상반되는 점수(예를 들어, 연인 사이인데 점수가 다소 낮은 경우), 3은 보통, 4는 일치하는 감성 점수, 5는 매우 일치하는 감성 점수이다. 그리고 각각의 사용자는 평가한 영화의 이해도를 1부터 3까지의 수치를 이용하여 측정하게 하였다. 1은 영화를 보지 않은 경우, 2는 영화를 보았지만 잘 기억이 나지 않거나 관계를

잘 모르는 경우, 3은 영화를 완전히 이해하는 경우를 나타낸다. 사용자로부터 데이터를 모두 취합한 후, 영화의 이해도가 2 이상인 영화들의 평가 점수를 분석하였다. 예를 들어, 표 3의 결과는 표 5와 같다.

Table 5. Evaluation score of a film <Frozen> analysis result

pairs	result	score
Anna -> Elsa	0.226424	3
Elsa -> Anna	0.178713	2
Elsa -> Hans	0.124833	5
Hans -> Anna	0.187306	5

2. Result

2-1. Retrieving Character Pairs

먼저 IV. 2에서 제안한 대사를 인물쌍에 추가하는 기법에 관한 실험을 진행하였다. 이미 인물쌍이 매칭된 5개의 영화 대본 데이터를 이용하여, 각 영화에서의 해당 기법의 정확도를 구하였다 [17]. 정확도는 인물 A의 대사 L이 인물 B를 향해 말한 것으로 결과가 나왔을 때 실제 데이터셋에서 대사 L이 A가 B에게 말한 것으로 라벨링 되어 있는 경우의 수를 전체 대사 수로 나눈 것이다. 표 6은 실험의 결과를 나타내는 표이다.

Table 6. Accuracy of the proposed method on 5 films

	영화1	영화2	영화3	영화4	영화5	평균
정확도	94.5	93.18	94.2	90.86	97.7	94.01

가장 낮은 정확도로 90.86, 가장 높은 정확도로 97.7을 기록했으며 영화 5편의 평균은 94.01을 기록하였다. 이러한 실험 결과는 IV. 2에서 제안한 기법이 영화 내의 발화자와 청자를 매칭하는데 좋은 성능을 보임을 입증한다.

2-2. Computing Sentiment Scores

표 7은 각 방법별 사용자 평가 점수를 보여주고 있다. 그림에서 알 수 있듯이 M_1 이 사용자로부터 가장 높은 평가를 받았다. 전체 구간을 3구간으로 나눈 경우(즉, $m = 3$)과 5구간으로 나눈 경우($m = 5$)를 비교해보면, $m = 3$ 인 경우가 더 우수한 성능을 보이는 것을 알 수 있다. 이는 구간이 세부적으로 나누어지면 가중치의 차이가 구간마다 커지게 되고, 특정 구간들(즉, 시간상으로 뒤에 위치한 구간들)의 감성 점수가 전체적인 인물쌍의 감성 점수에 미치는 영향이 커지기 때문이다.

Table 7. Average evaluation score for each method

방법	M_1	M_2	M_3	M_4
점수	3.3	3.1	2.8	2.6

$m = 3$ 인 경우 사용자 평가 3과 4점의 사용자 평가 점수가 가장 많았다. 반면에, $m = 5$ 인 경우 긍정적인 평가는 떨어지고 오히려 부정적인 평가가 증가함을 실험을 통해 알 수 있었다. 즉, $m = 5$ 인 경우 $m = 3$ 보다 1점과 2점이 현저히 증가하였으며, 4점은 크게 낮아진 것을 확인할 수 있었다. 따라서 영화의 구간을 세분화하는 것은 영화를 분석하는데 오히려 방해가 된다는 것을 알 수 있다. 이는 대부분의 영화 시나리오가 설정, 대립, 해결의 3막 구조를 따르는 것에서 이유를 찾을 수 있을 것이다 [26]. 5개 구간으로 나누게 된다면 영화의 전반부와 후반부에 부여되는 가중치의 차이가 커지기 때문에 전체적인 대사들을 고려하지 않고 후반부의 대사에만 영향을 받게 되어서 이러한 결과가 나오는 것으로 보인다.

그림 5는 M_1 과 M_2 의 (즉, $m = 3$ 인 경우)의 사용자 평가 점수 분포도에 해당한다. 여기서, x축은 사용자 평가 점수, y축은 사용자 평가 점수의 해당 빈도수이다. 그림에서 알 수 있듯이 $m = 3$ 에 해당하는 M_1 , M_2 에서는 전체적으로 양호한 점수 분포를 보였다. 특히 그림 5.(a)를 보면 M_1 의 경우 1이나 5의 양극에 위치한 평가는 낮은 반면 4점과 3점이 많은 것을 확인할 수 있다. 그림 5.(b)에서 M_2 의 경우 4점이 소폭 상승한 반면 2점이 감소했다. 그러나 1점이 증가하고 3점이 감소하여 평균적으로는 M_1 보다 좋지 못한 평가를 받았다.

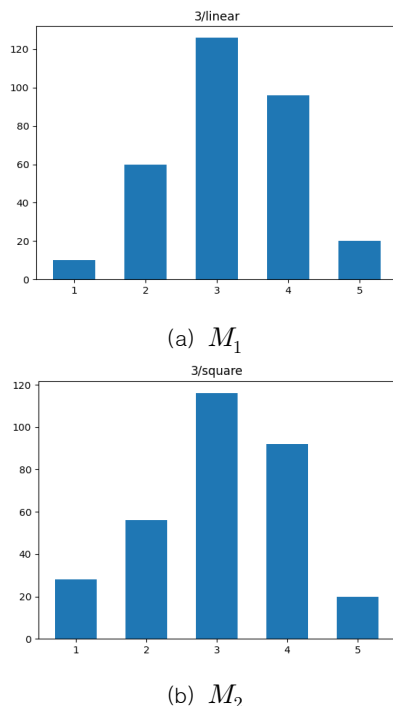


Fig. 5. Evaluation score distribution of M_1 and M_2

VI. Conclusion

본 논문에서는 딥러닝 기술을 활용해 영화 내 등장인물 간의 관계를 분석하여 감성 점수를 구하는 방법을 제안하였다. 본 논문에서 제안한 기법은 대사를 각 인물의 쌍에 매칭한 후 감성 분석을 시행하여 대사의 위치에 따른 가중치를 두어 그 점수를 취합하는 방법이다. 구간은 영화의 전통적인 구조에 따라 전체를 3개의 구간으로 나누는 것이 좋으며 후반부에 과하게 가중치를 주는 것이 아닌 선형 함수로 가중치를 증가시켜 더욱 정확하게 영화의 전체적인 흐름을 반영한 감성 점수를 확인할 수 있다. 향후 연구에서는 본 논문에서 제안한 방법에 관계 추출(Relation extraction) 기법을 도입하고 관계 점수의 정확도를 대사 뿐 아니라 영화 대본 내의 등장인물의 행동이나 독백을 고려하여 보다 정확하고 다양한 인물들 간의 관계 정보를 제공하는 방법을 고안하고자 한다.

ACKNOWLEDGEMENT

This work was supported by Institute for Information & communications Technology Promotion(IITP) grant funded by the Korea government(MSIP) (No.2021-0-00884, Development of Integrated Platform for Untact Collaborative Solution and Blockchain Based Digital Work).

REFERENCES

- [1] Yaakub, Latiffi, Zaabar, (2019). A Review on Sentiment Analysis Techniques and Applications. IOP Conf. Series: Materials Science and Engineering. <https://doi.org/10.1088/1757-899x/551/1/012070>
- [2] Russell, (1980). A circumplex Model of Affect. (pp. 1167). Journal of Personality and Social Psychology Vol. 39, No. 6, 1161-1178. <https://doi.org/10.1037/h0077714>
- [3] David, (2014). Screenwriting and emotional rhythm. Journal of Screenwriting 5(1):47-57. https://doi.org/10.1386/josc.5.1.47_1
- [4] IMDb <https://imdb.com/>
- [5] Hutto, Gilbert, (2015). VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text. ICWSM proceedings.
- [6] Mohammad, Turney, (2013). NRC Emotion Lexicon, National Research Council, Canada.
- [7] Yu, Kim, Bae, (2017). Emotion and Sentiment Analysis from a Frilm Script: A Case Study. Journal of Digital Contents Society

- Vol. 18, No. 8, 1537-1542.
- [8] Yasen, Tedmori, (2019). Movies Reviews Sentiment Analysis and Classification, Conference: 2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT). <https://doi.org/10.1109/jeeit.2019.8717422>
- [9] Uma Ramya, V., & Thirupathi Rao, K. (2018). Sentiment Analysis of Movie Review using Machine Learning Techniques. *International Journal of Engineering & Technology*, 7(2.7), 676. <http://dx.doi.org/10.14419/ijet.v7i2.7.10921>
- [10] Baid, P., Gupta, A., & Chaplot, N. (2017). Sentiment Analysis of Movie Reviews using Machine Learning Techniques. *International Journal of Computer Applications*, 179(7), 45-49. <http://dx.doi.org/10.5120/ijca2017916005>
- [11] Lee, Yu, Cheong, (2017). Analyzing Movie Scripts as Unstructured Text. 2017 IEEE Third International Conference on Big Data Computing Service and Applications (BigDataService). <https://doi.org/10.1109/bigdataservice.2017.43>
- [12] A. Esuli and F. Sebastiani, SENTI-WORDNET: A Publicly Available Lexical Resource for Opinion Mining. (2006). Proc. of the 5th Conf. on Language Resources and Evaluation (LREC'06), Genoa, Italy, May. 2006, pp. 417-422. https://doi.org/10.1007/978-3-642-20095-3_46
- [13] Francis, M.E.; Booth, R.J, (1993). *Linguistic Inquiry and Word Count*; Southern Methodist University: Dallas, TX, USA.
- [14] William S. Cleveland, (1979), Robust Locally Weighted Regression and Smoothing Scatterplots, *Journal of the American Statistical Association*, Vol. 74, No. 368 (Dec., 1979), pp. 829-836.
- [15] Abraham Savitzky, M. J. E. Golay, (1964). Smoothing and Differentiation of Data by Simplified Least Squares Procedures, *Anal. Chem.* 1964, 36, 8, 1627-1639. <https://doi.org/10.1021/ac60214a047>
- [16] Haddad RA, Akansu AN, (1991). A class of fast Gaussian binomial filters for speech and image processing. *IEEE Trans Signal Process* 1991;39(3):723-737. <https://doi.org/10.1109/78.80892>
- [17] Devisree, V., & Raj, P. C. R. (2016). A Hybrid Approach to Relationship Extraction from Stories. *Procedia Technology*, 24, 1499-1506. <https://doi.org/10.1016/j.protcy.2016.05.101>
- [18] Lakin, J.L., Jefferis, V.E., Cheng, C.M., Chartrand, T.L. (2003) The chameleon effect as social glue: Evidence for the evolutionary significance of nonconscious mimicry. *J. Nonverbal Behav.* 145-162. <https://doi.org/10.1023/a:1025389814290>
- [19] Danescu-Niculescu-Mizil, Lee, (2011). Chameleons in imagined conversations: A new approach to understanding coordination of linguistic style in dialogs. *ACL*, Portland, OR, USA, 23 June 2011.
- [20] Francis, M.E, Booth, R.J, (1993). *Linguistic Inquiry and Word Count*; Southern Methodist University: Dallas, TX, USA.
- [21] *IMSDb* <https://imfdb.com/>
- [22] *BeautifulSoup*, <https://www.crummy.com/software/BeautifulSoup/bs4/doc/>
- [23] Halteren, (2002). Writing Style Recognition and Sentence Extraction, *Proceedings of the workshop on automatic summarization*, pp:66-70. (2002)
- [24] Ireland, Slatcher, Eastwick, (2011). Language Style Matching Predicts Relationship Initiation and Stability, *Psychological Science* 22(1):39-44. <https://doi.org/10.1177/0956797610392928>
- [25] *Kaggle*, <https://www.kaggle.com/c/sentiment-analysis-on-movie-reviews/data>
- [26] Syd Field, (1979), *The Foundations of Screenwriting*, Dell Publishing Company

Authors



Kyo Jun Jin received the B.S. student in Department of Computer Science from Sangmyung University, Seoul, Korea, in 2021. He is interested in natural language processing. Such as Emotion and Sentiment

Analysis, Named Entity Recognition and Text Pseudomization. And he is studying natural language processing.



Jong Wook Kim received the Ph.D. degree from the Computer Science Department, Arizona State University, in 2009. Dr. Kim was a Software Engineer with the Query Optimization Group at Teradata, from 2010

to 2013. He is currently an Assistant Professor of computer science with Sangmyung University. His primary research interest is in the area of data privacy, distributed databases, and query optimization. He is a member of the ACM.