KGU
경기대학교

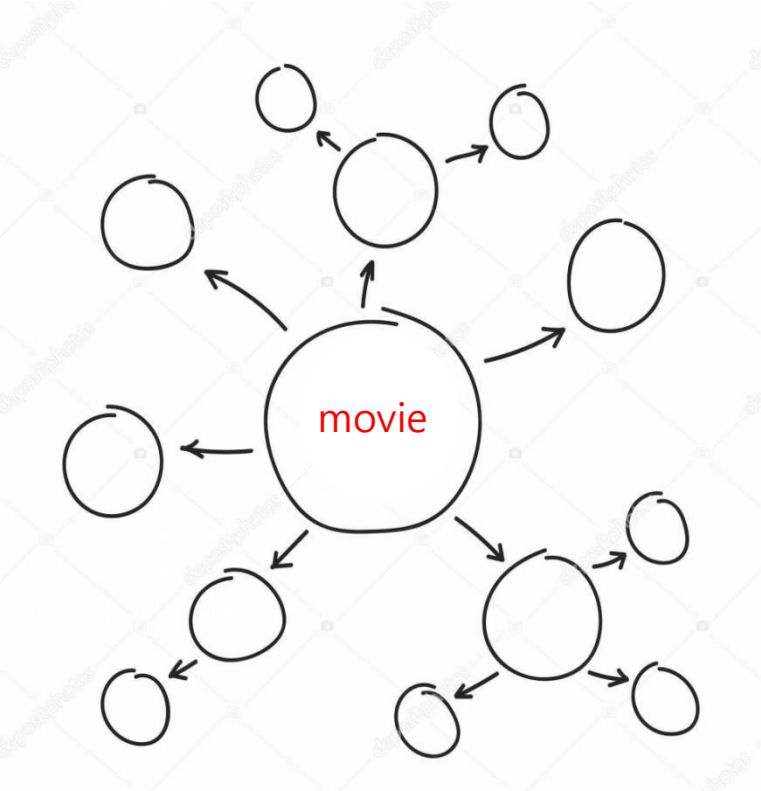
word2vec과 t-sne로 알아본 단어들간의 의미상 유사도

경기대학교
응용통계학과
201411106권용한
201511398박소담
201611423김다은

CONTENTS

KGU 경기대학교

1 소개 - 주제



네이버 영화 리뷰
단어들 간의 유사도가 있을까?

영화 리뷰를 보면 특정 단어 및 문구가 자주 나오는 것을 확인할 수 있다.
예를들면, '어벤져스'라고 하면 '로다주', '아이언맨','헐크' 등 등장인물이나 '꿀잼',
'노잼' 등 영화 자체에 대한 평가를 하는 단어들이 자주 나온다.
그렇다면 각 영화에서 어떤 단어들이 유사도가 높게 나오는지 확인하고자 한다.

*유사도란? 동질성의 정도를 수치화 한 것

KGU 경기대학교

1 소개 – 사용 툴 및 데이터 출처

Google colaboratory

구글에서 제공하는 툴로 파이썬 주피터 노트북을 클라우드 환경에서 사용할 수 있는 툴이다.클라우드 환경에서 사용되기 때문에 구글은 colaboratory를 사용할 때,성능이 좋은 GPU,TPU를 제공한다. 이를 통해 로컬 컴퓨터에서 오래 걸리는 작업을 보다 빠르게 작업할 수 있다.

네이버 영화

네이버 영화에서는 다양한 영화리뷰가 존재한다. 영화 선정 기준은 모든 영화의 평점 순으로 1~150위를 조사하였고, 추가로 애니메이션 장르를 추가하였다.(장르 다양성 추구를 위해) 따라서 크롤링 한 영화의 수는 169개이며, 총 리뷰수는 약 53만 개 이다.

KGU 경기대학교

2 Word 2 Vec – 이론

Word2Vec data generation (skip gram) (window size =2)

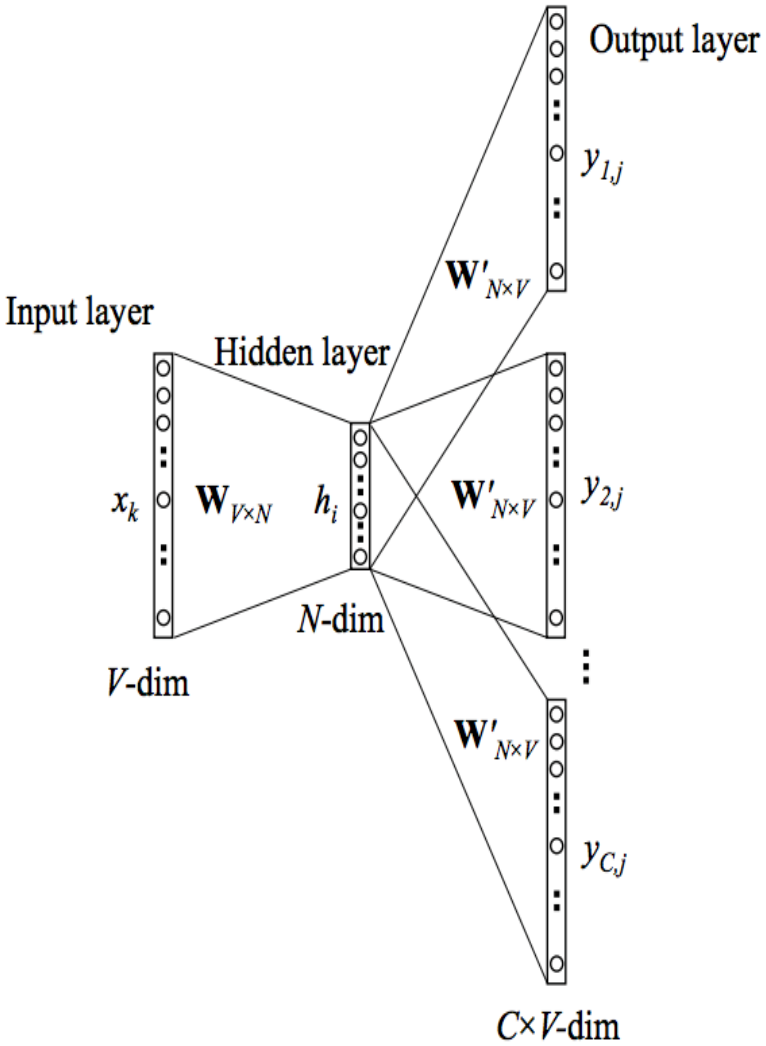
“I am Iron-man”

“Wonder woman”

Word	Neighbor
I	Am
I	Iron-man
Am	I
Am	Iron-man
Iron-man	I
Iron-man	Am
Wonder	Woman
Woman	Wonder

KGU 경기대학교

2 Word 2 Vec – 이론



$$[0 \quad 0 \quad 0 \quad \underbrace{1}_{\chi_4} \quad 0] \times \underbrace{\begin{bmatrix} 17 & 24 & 1 \\ 23 & 5 & 7 \\ 4 & 6 & 13 \\ 10 & 12 & 19 \\ 11 & 18 & 25 \end{bmatrix}}_{w_1} = [10 \quad 12 \quad 19]$$

$$[10 \quad 12 \quad 19] \times \underbrace{\begin{bmatrix} 1 & 2 & 5 & 8 & 9 \\ 2 & 4 & 3 & 2 & 1 \\ 5 & 6 & 9 & 2 & 4 \end{bmatrix}}_{\mathbf{w}'_2} = \begin{bmatrix} 129 \\ 182 \\ 257 \\ 142 \\ 178 \end{bmatrix}$$

$$\Rightarrow softmax\left(\begin{bmatrix} 129 \\ 182 \\ 257 \\ 142 \\ 178 \end{bmatrix}\right) = \begin{bmatrix} 0.04 \\ 0.11 \\ 0.70 \\ 0.06 \\ 0.09 \end{bmatrix}$$

$P(\chi_2|\chi_4)$

$$\left(\text{softmax}(\text{Z}_\text{j}) = \frac{e^{Z_j}}{\sum e^{Z_k}}\right)$$

KGU 경기대학교

2 Word 2 Vec – 이론

$$P(o|c) = \frac{\exp(u_o^T v_o)}{\sum_{w=1}^{W} \exp(u_w^T v_c)}$$

V는 입력층-은닉층을 잇는 가중치 행렬 w의 행벡터

U는 은닉층-출력층을 잇는 가중치 행렬 w'의 열벡터

중심단어(c) 가 주어졌을 때 주변단어(o) 가 나타날 확률

KGU 경기대학교

2 Word 2 Vec – 이론

$$\frac{\partial}{\partial \mathrm{v_c}} \ln P(o|c) = \frac{\partial}{\partial v_c} \ln \frac{\exp(u_o^T v_c)}{\sum_{\omega=1}^{W} \exp(u_\omega^T v_c)}$$

$$= \frac{\partial}{\partial v_c} \mathrm{u}_o^{\mathrm{T}} v_c - \frac{\partial}{\partial v_c} \ln \sum_{\omega=1}^{W} \exp(u_\omega^T v_c)$$

$$= \mathrm{u}_o^{\mathrm{T}} - \frac{1}{\sum_{\omega=1}^{W} \exp(u_\omega^T v_c)} \left(\sum_{\omega=1}^{W} \exp(u_\omega^T v_c) \cdot u_\omega\right)$$

$$= \mathrm{u}_{\mathrm{o}}^T - \sum_{\omega=1}^{W} \frac{\exp(u_\omega^T v_c)}{\sum_{\omega=1}^{W} \exp(u_\omega^T v_c)} \cdot u_\omega$$

$$= \mathrm{u}_{\mathrm{o}}^T - \sum_{\omega=1}^{W} P(\omega|c) \cdot u_\omega$$

KGU 경기대학교

2 Word 2 Vec – 이론

$$v_c^{t+1} = v_c^t + \alpha\left(u_o^T - \sum_{w=1}^{W} P(w|c) \cdot u_w\right)$$

중심단어 그래디언트의 반대 방향으로 조금씩 중심단어 벡터를 업데이트

α : 사용자가 지정하는 학습률 (learning rate)

KGU 경기대학교

2 Word 2 Vec – 결과

하늘과 바다

하늘/과/바다

```
model = Word2Vec(new_word2vec_corpus,
 size=300, window=10, min_count=5,
workers=10, iter=25, sg=1)
```

1. 53만개 리뷰를 Soynlp로 형태소 분리
2. 형태소 분리한 말뭉치를 word2vec으로 학습

KGU 경기대학교

2 Word 2 Vec – 겨울왕국


```
import gensim.models as g

model = g.Word2Vec.load(model_name)
print(model.most_similar(positive= ' 겨울왕국'))
```

[('엘사', 0.6214337348937988),('렛잇고',0.5913184285163879),
('디즈니', 0.566076939422607), ('라푼젤', 0.558643162250518 8),
('조아요조아', 0.476159214973 4497), ('레릿고',0.4464509785 1753235),
('레잇고', 0.434395194053 6499), ('열풍이', 0.4255936741 8289185),
('레리꼬', 0.4059495031 8336487), ('주토피아',0.4036033153 5339355)]

KGU 경기대학교

2 Word 2 Vec – 동주


```
import gensim.models as g

model = g.Word2Vec.load(model_name)
print(model.most_similar(positive= ' 동주'))
```

[('윤동주', 0.716146588325500 5), ('강하늘', 0.619118154048919 7), ('송몽규', 0.614826142787933 3), ('몽규', 0.576820313930511 5), ('시를', 0.572955012321472 2), ('시인', 0.542444944381713 9), ('박정민', 0.529736876487731 9), ('시가', 0.513094067573547 4), ('흑백', 0.498413562774658 2), ('부끄러', 0.490002751350402 83)]

KGU 경기대학교

2 Word 2 Vec – 토이스토리


```
import gensim.models as g

model = g.Word2Vec.load(model_name)
print(model.most_similar(positive= ' 토이스토리'))
```

[('토이 스토리 3', 0.669425964355468), ('우디',0.4787367582321167), ('토이', 0.4661992192268371 6), ('장난감', 0.4617300629615783 7), ('토이 스토리 2', 0.4549075961112976), ('버즈',0.4381361603736877 4), ('토이 스토리', 0.4302777647972107), ('픽사', 0.4297353625297546 4), ('앤디', 0.4284959733486175 5), ('장난감들', 0.4189223051071167)]

KGU 경기대학교

2 Word 2 Vec – Harry Potter

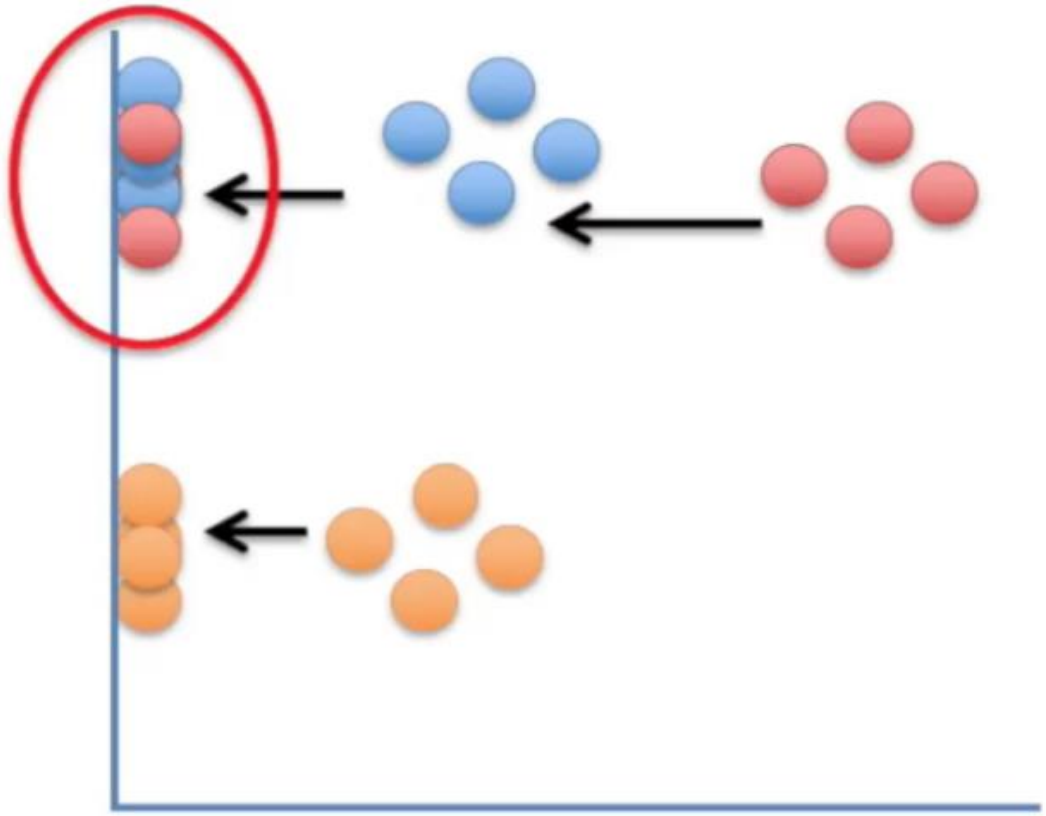

```
import gensim.models as g

model = g.Word2Vec.load(model_name)
print(model.most_similar(positive='해리포터'))
```

[('해리 포터와 죽음의 성물 - 2부', 0.7173735499382019), ('해리 포터와 마법사의 돌', 0.6750643849372864), ('시리즈', 0.550640106201171 9), ('포터', 0.527159810066223 1), ('마법사의돌', 0.5141783952713013), ('헤르미온느', 0.505155861377716 1), ('해리', 0.4976233243942261), ('성물', 0.4828528165817261), ('마법사의', 0.473886847496032 7), ('롤링', 0.465217232704162 6)]

KGU 경기대학교

3 t-SNE – 이론



2차원에서 1차원으로 줄인 PCA 그림

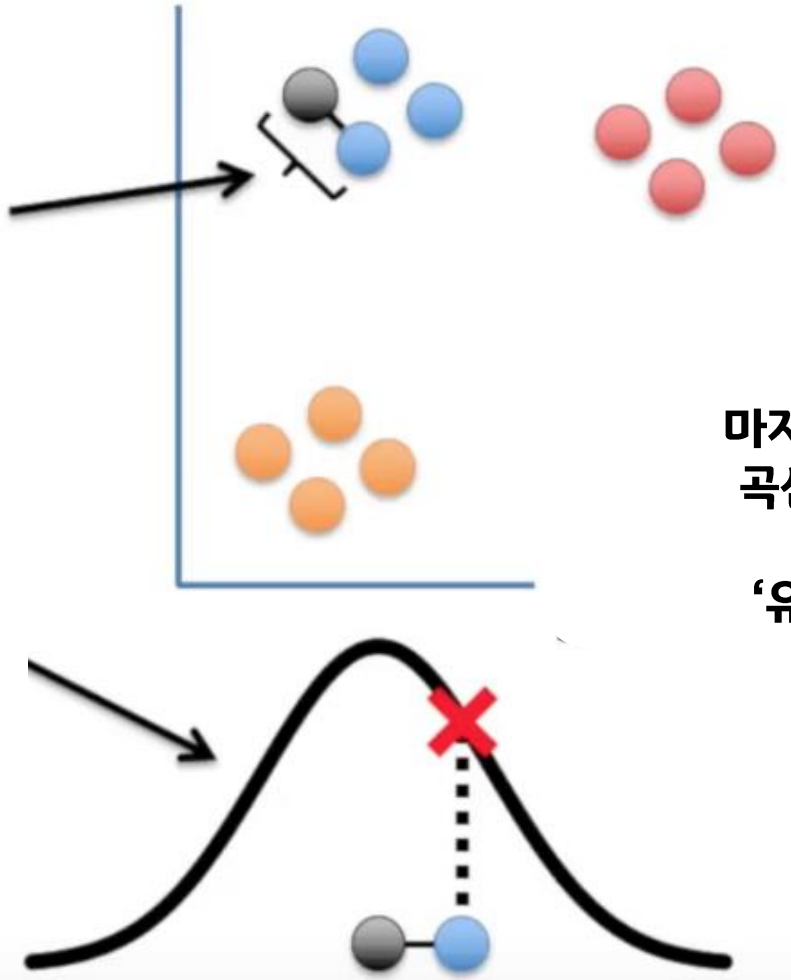
PCA의 문제점

선형 분석 방식으로 값을 사상한다.
그러므로 차원이 감소되면서 군집화
되어있는 데이터들이 뭉치면서 제대로
구별할 수 없는 문제를 가지고 있다.

KGU 경기대학교

3 t-SNE – 이론

첫째, 두 점 간의
거리를 구한다.

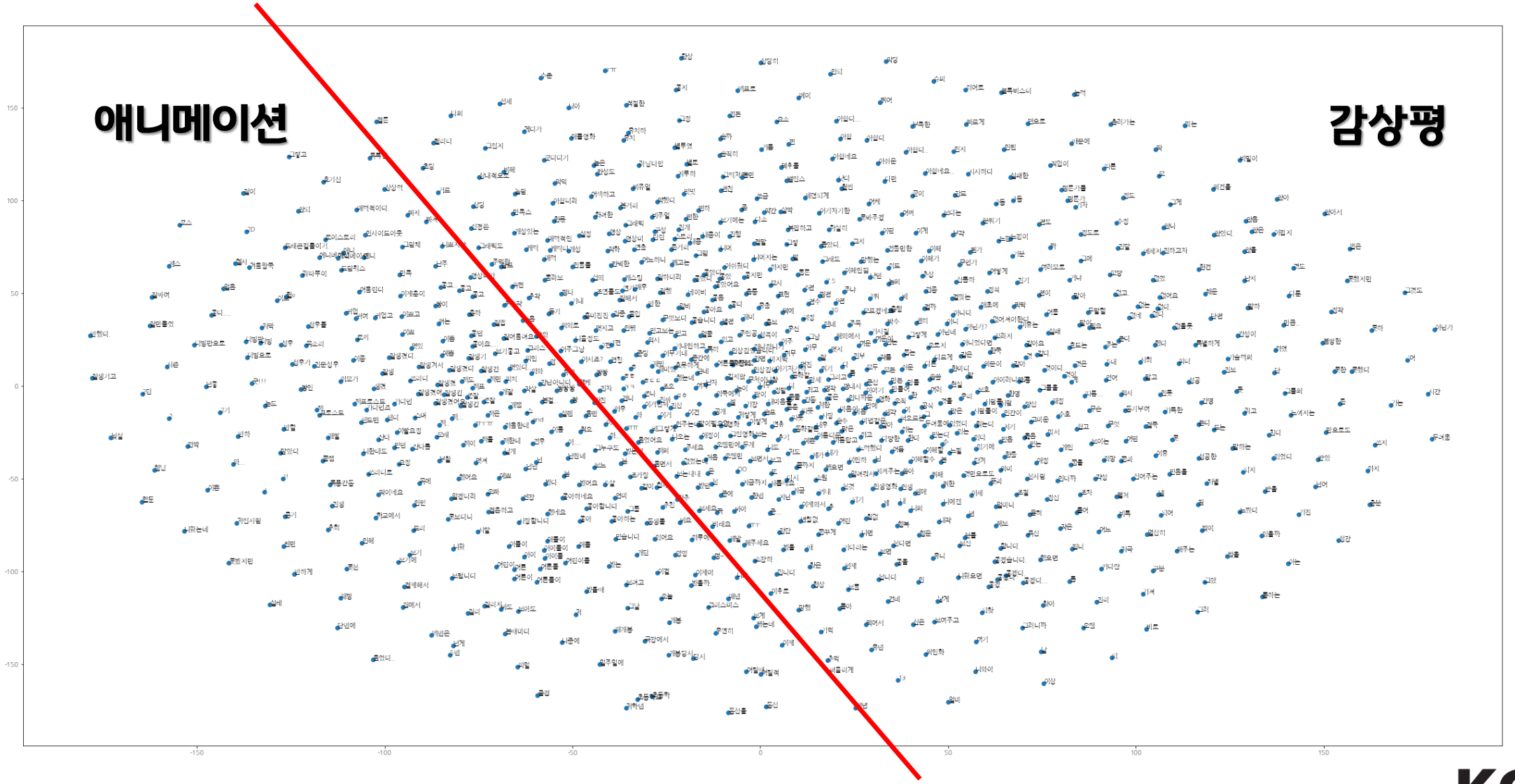


그런 다음 관심 지점을
중심으로 한 t분포 그래프에
해당 거리를 표시

마지막으로, 포인트에서
곡선으로 선을 그린다.
그 선의 길이는
'유사성의 정도'이다.

KGU 경기대학교

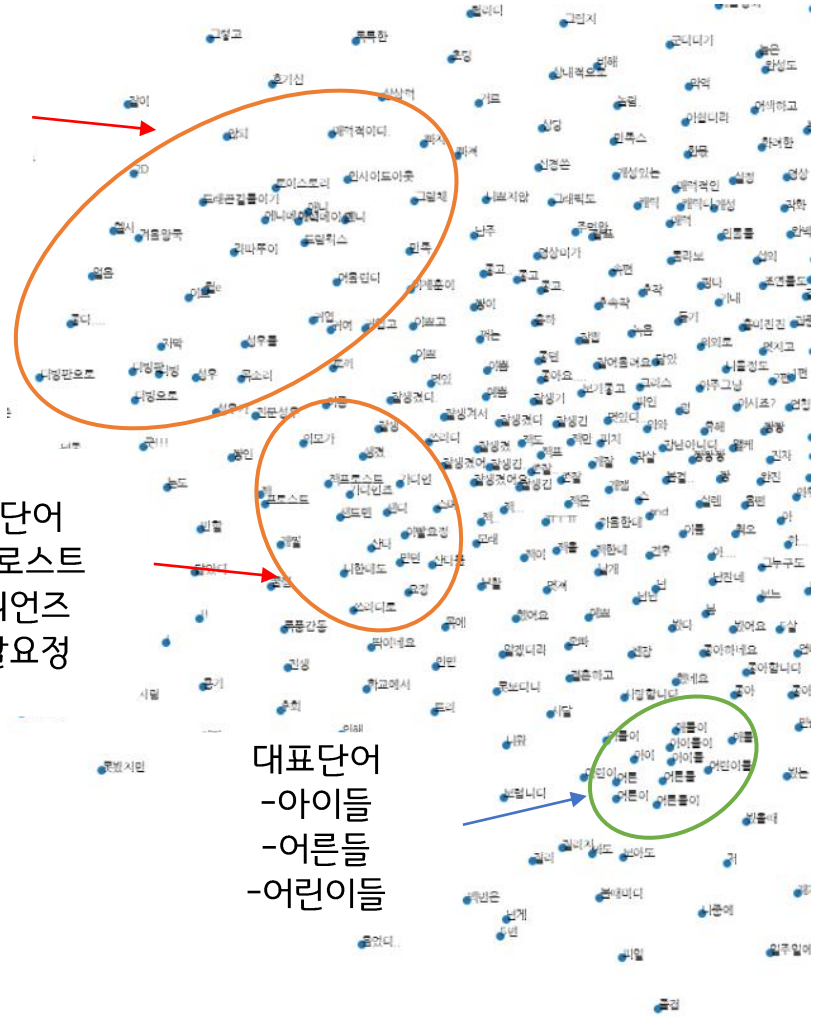
3 t-SNE – 결과



KGU 경기대학교

3 t-SNE – 결과

대표단어
-겨울왕국
-드림웍스
-인사이드아웃



대표단어
-잭프로스트
-가디언즈
-이빨요정

대표단어
-아이들
-어른들
-어린이들

애니메이션 부분(그래프 왼쪽)

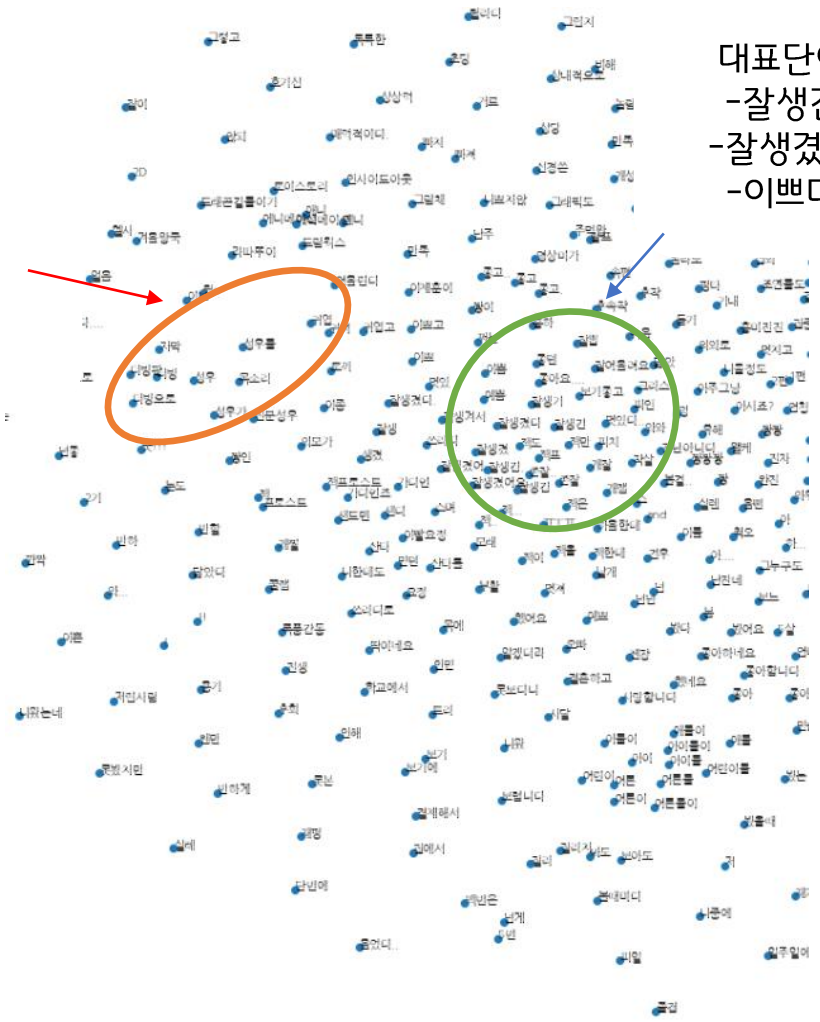
위 그래프를 볼 때, ORANGE 부분을 보았을 때, 애니메이션의 제목, 등장인물, 제작사 등의 단어들이 나왔다. 이를 통해 그래프의 왼쪽부분은 상당히 애니메이션 영화의 단어들의 유사도가 높다고 할 수 있다. 또한 GREEN 부분은 일반영화에서 나오지 않는 주어들인 '어른들이', '아이들', '아이' 등의 단어들이 나왔는데 이를 통해 애니메이션의 주 관람객 층이라고 유추할 수 있다.

KGU 경기대학교

3 t-SNE – 결과

대표단어
-성우
-더빙
-목소리

대표단어
-잘생긴
-잘생겼다
-이쁘다



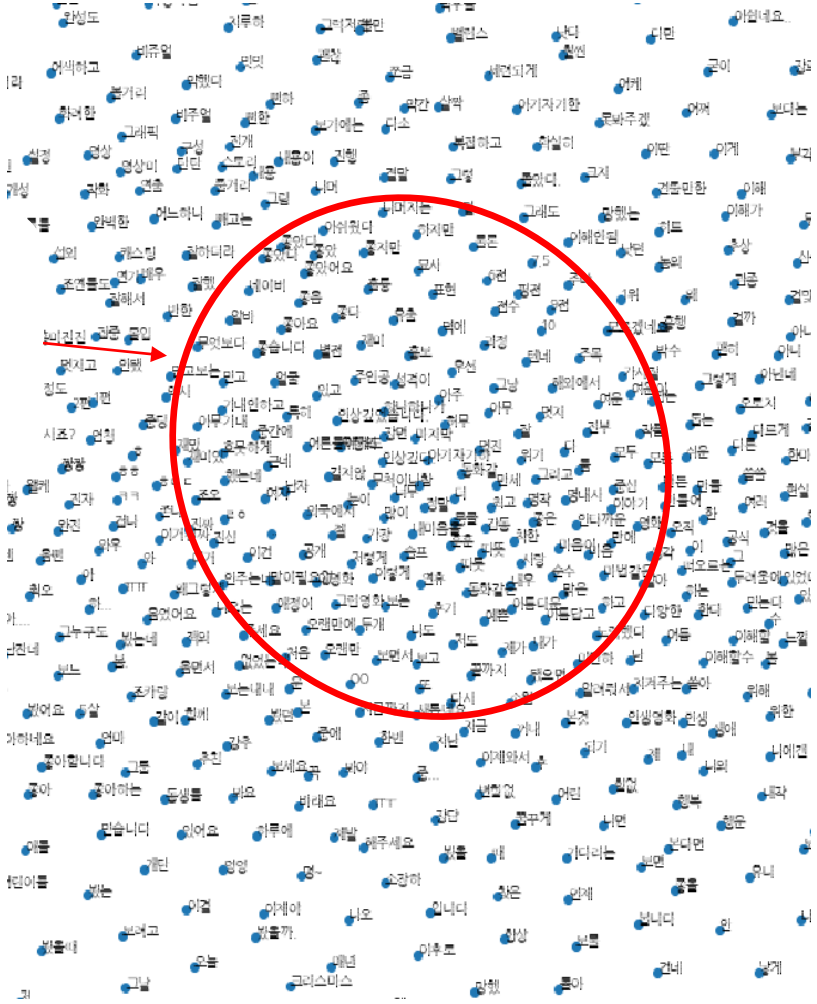
애니메이션 부분(그래프 왼쪽)

ORANGE 부분을 볼 때, '성우', '더빙' 등 애니메이션에서 많은 비중을 차지하는 단어들은 그래프 왼쪽에 치중 되어 있다. 그러나 GREEN 부분은 '잘생겼다','예쁘다' 등 외모에 대한 평가가 써있다. 이 단어들은 애니메이션과 영화평에 대한 구분선 근처에 위치하고 있다. 이를 볼 때, 외모에 대한 평가는 애니메이션 이외의 영화들에도 발견할 수 있다는 것을 유추할 수 있다.

KGU 경기대학교

3 t-SNE – 결과

대표단어
-최고
-명작
-인상깊다



감정단어 부분(그래프 중심)

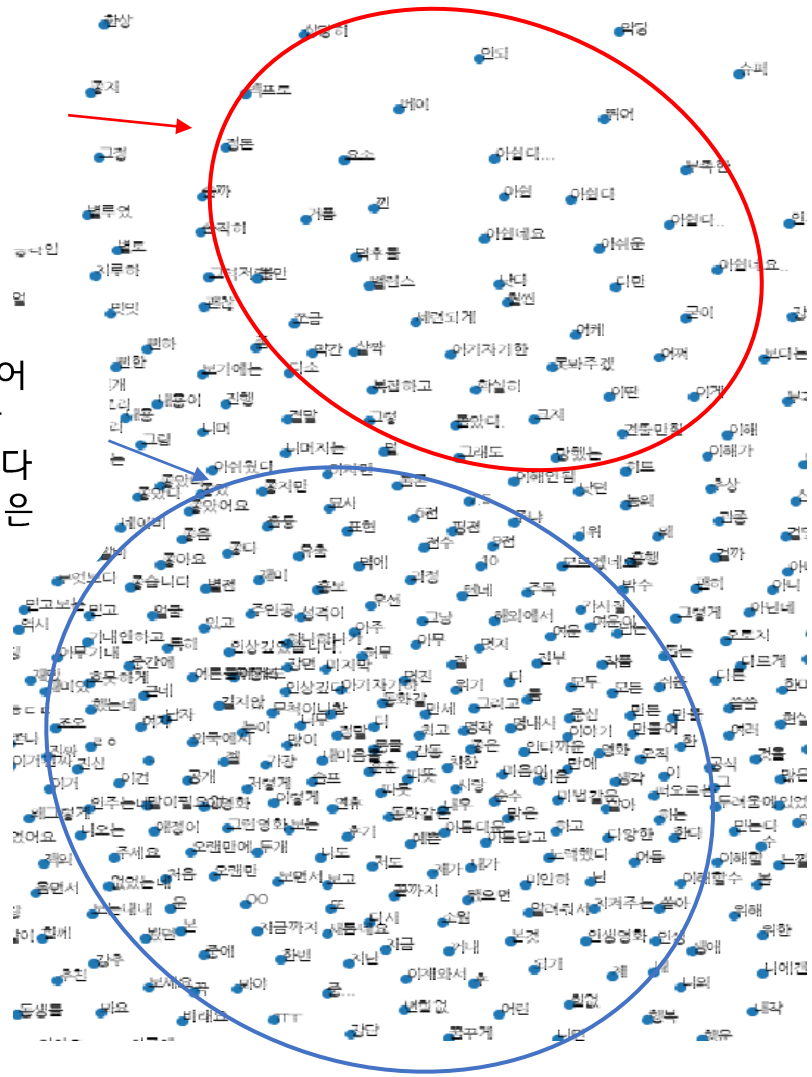
전체 그래프에서 중심 부분은 주로 영화에 대한 감정 단어들이 많이 나온다. 가장 중심단어인 '인상 깊다 ' 주위로 '최고','명작','좋았어요','아름다운 ' 등등의 영화에 대한 평가 혹은 감정단어들이 중심에 위치하고 있다.

KGU 경기대학교

3 t-SNE – 결과

대표단어
-거품
-아쉽다
-망했는

대표단어
-명작
-좋습니다
-인상깊은

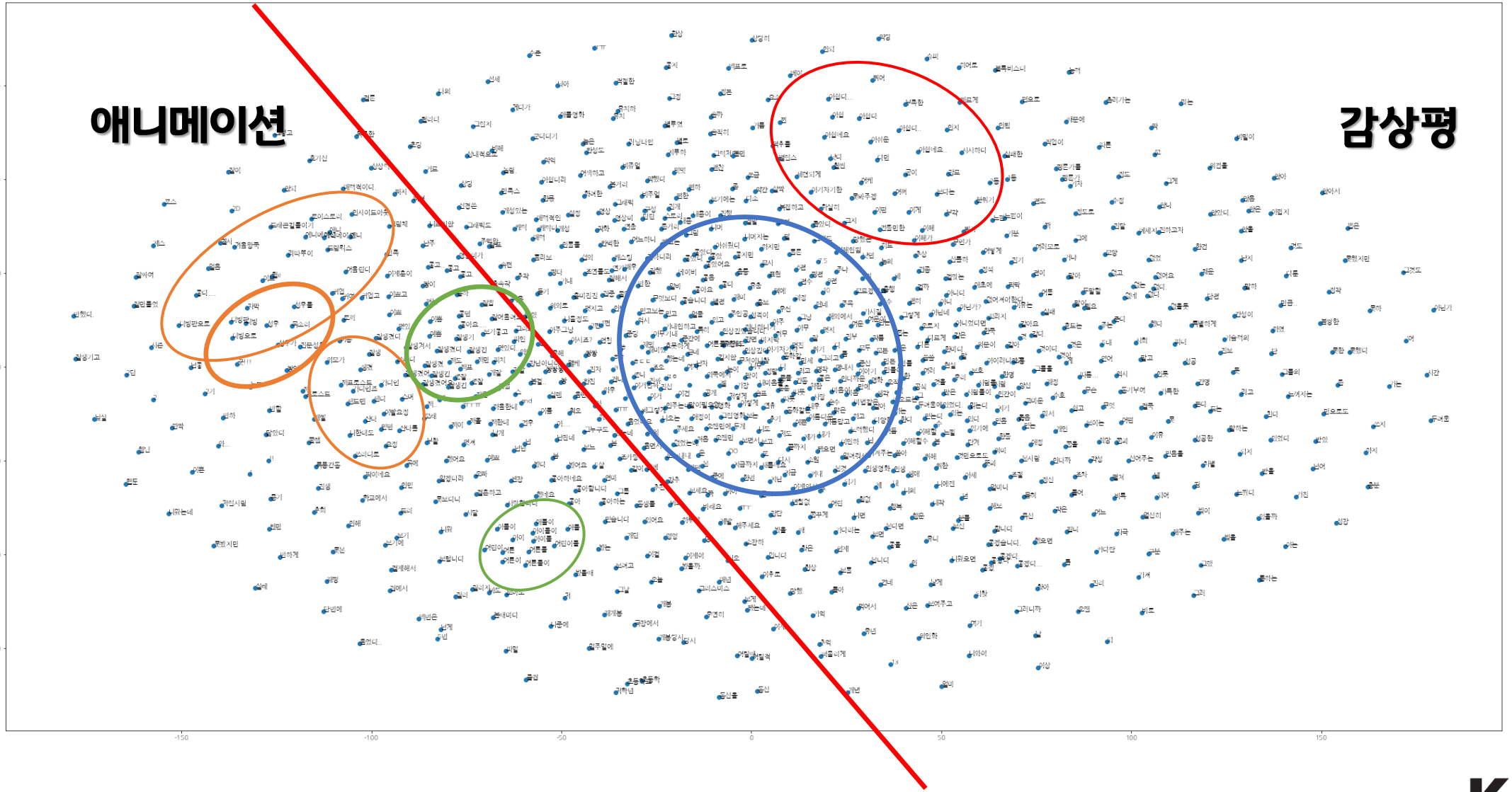


감정단어 부분(그래프 오른쪽)

RED 부분을 보았을 때, 부정적인 단어가 많이 나온 것을 확인할 수 있다. 예를 들어 '아쉽다','거품','망했는' 등 영화에 대한 부정적인 단어들이 전체 그래프의 오른쪽 상단에 위치하였다.
BLUE 부분은 '인상깊은','좋습니다.','명작' 등 영화에 대한 긍정적인 단어들이 나타났다. 이 단어들은 전체 그래프의 중심에 위치해있었다. 이를 통해 전체 영화 리뷰들은 대부분 긍정적인 평이 부정적인 평보다 많이 존재한다는 것을 유추할 수 있다.

KGU 경기대학교

3 t-SNE – 결과



KGU 경기대학교

4

후속 연구

형태소분리기

이번 분석에서 사용한 형태소분리기는 'SoyNLP'를 사용하였다.
그렇지만 다른 형태소 분리기(ex. KONLPY, khaiii)를 사용할 때,
다른 결과들이 나오게 된다. 다른 결과들을 확인하지 못한 것이 아쉽다.

사용자 주관

단어들의 벡터화를 확인 할 때, 수치적인 방법을 사용하지 않고,
사용자의 임의적인 판단으로 관계를 봤다. 다음에는 이런 관계를
수치적으로 해석하는 방법(ex.KNN)을 찾고 연구해야 할 것이다.

KGU 경기대학교

KGU
경기대학교


감사합니다