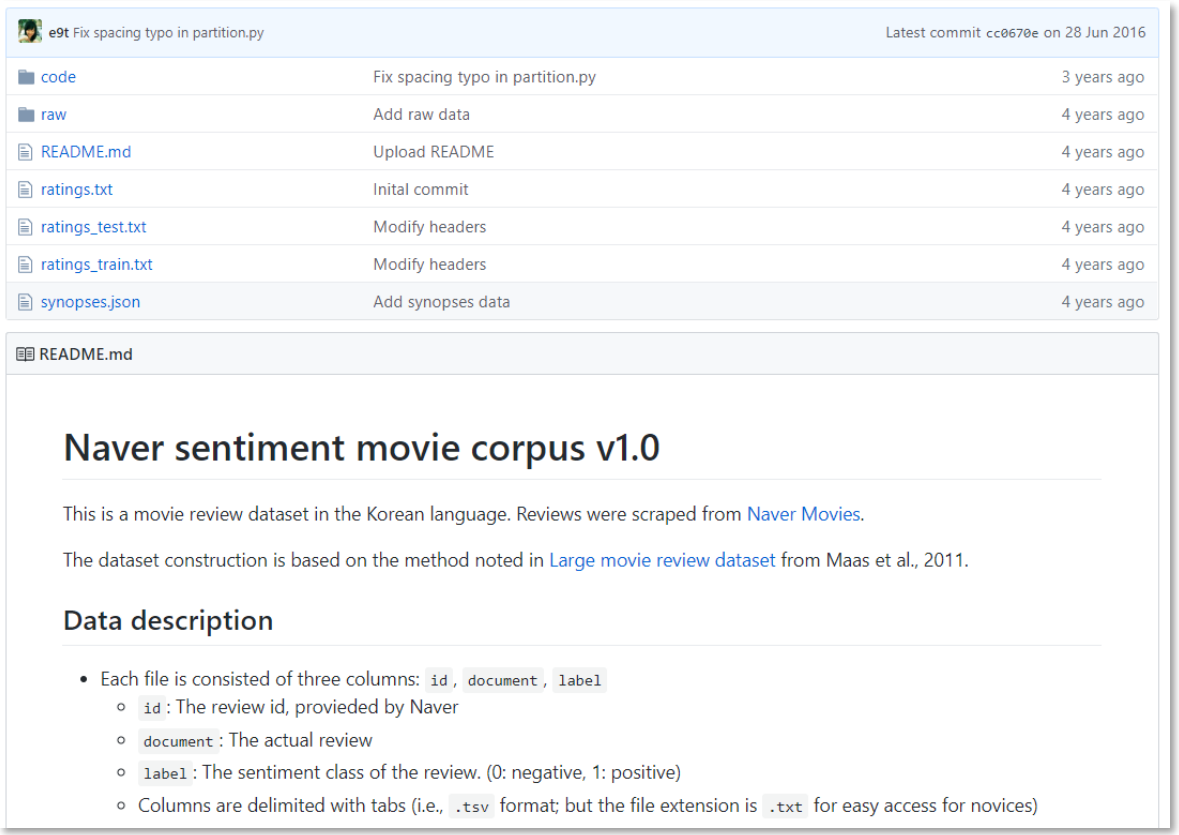
다중 감정 분류
REPORT

진명훈 김호현 〉

활용데이터 – NSMC

https://github.com/e9t/nsmc

https://movie.naver.com/



e9t Fix spacing typo in partition.py — Latest commit cc0670e on 28 Jun 2016

code	Fix spacing typo in partition.py	3 years ago
raw	Add raw data	4 years ago
README.md	Upload README	4 years ago
ratings.txt	Inital commit	4 years ago
ratings_test.txt	Modify headers	4 years ago
ratings_train.txt	Modify headers	4 years ago
synopses.json	Add synopses data	4 years ago

README.md

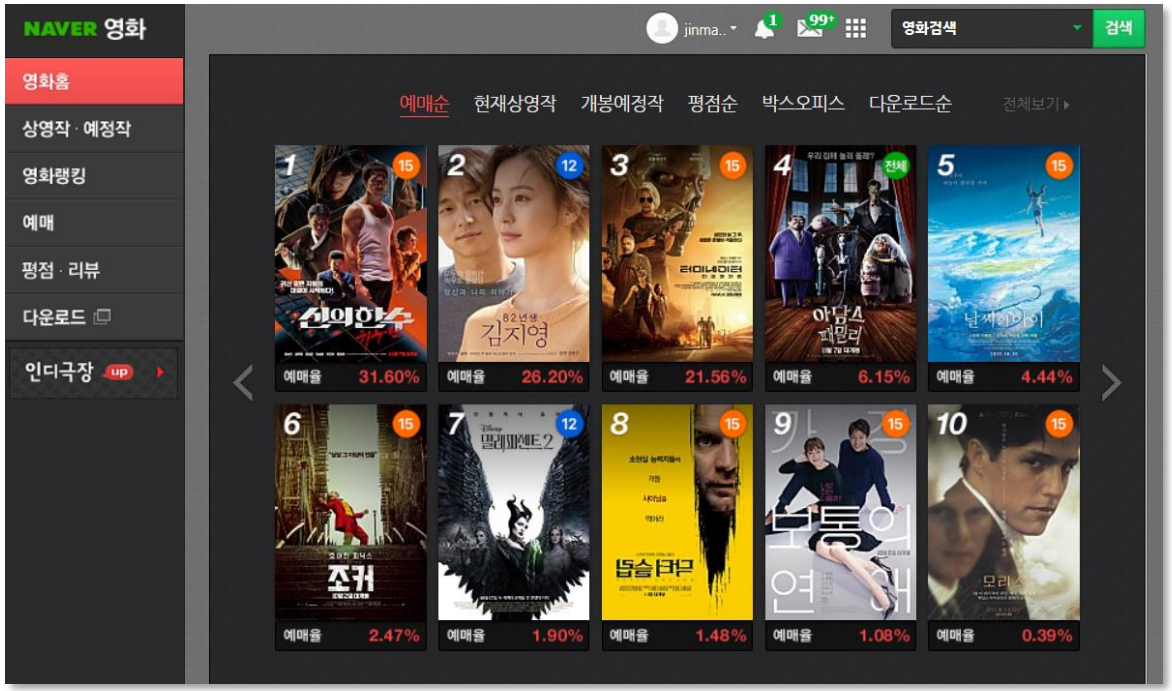
Naver sentiment movie corpus v1.0

This is a movie review dataset in the Korean language. Reviews were scraped from Naver Movies.

The dataset construction is based on the method noted in Large movie review dataset from Maas et al., 2011.

Data description

- Each file is consisted of three columns: `id`, `document`, `label`
  - `id`: The review id, provieded by Naver
  - `document`: The actual review
  - `label`: The sentiment class of the review. (0: negative, 1: positive)
  - Columns are delimited with tabs (i.e., `.tsv` format; but the file extension is `.txt` for easy access for novices)



총 71만 영화 리뷰 데이터

총 253만 영화 리뷰 데이터

감정사전 – KSenticNet

README.md

KSenticNet: 한국어 감성 사전 (Korean sentiment resource)

How to use

- Just download 'ksenticnet_kaist.py' file :)

Overview

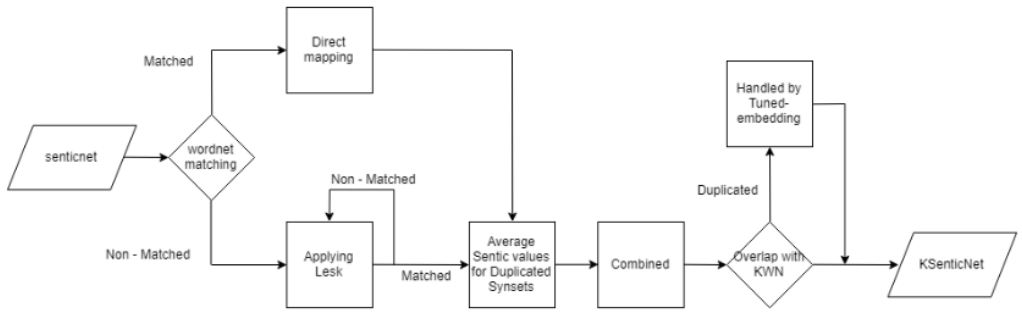
- There are several Korean sentiment analysis resources such as KNU SentiLex, KOSAC.
- However, sentiment lexicons like them require a lot of time and human resources.
- So I decided to make it easier and automated by combining SenticNet and KAIST Korean wordnet(KWN).

Example Image

```
ksenticnet["가교"] = ['0.791', '0.857', '0', '0', '#joy', '#interest', 'positive', '0.824', '다리']
ksenticnet["가구"] = ['0.514', '-0.97', '0', '0.899', '#surprise', '#admiration', 'positive', '0.797', '세간', '비품']
ksenticnet["가금"] = ['-0.9', '0.0', '-0.874', '0.0', '#sadness', '#fear', 'negative', '-0.884', '닭', '암탉', '사조']
ksenticnet["가난"] = ['-0.373', '0.0', '0.14', '-0.911', '#disgust', '#sadness', 'negative', '-0.784', '빈곤', '궁핍']
ksenticnet["가난뱅이"] = ['-0.05', '0', '-0.11', '-0.04', '#sadness', '#fear', 'negative', '-0.06', '빈자']
ksenticnet["가난하"] = ['0.0', '0.0', '-0.855', '-0.755', '#fear', '#disgust', 'negative', '-0.805']
```

- You can get words' sentic values, sentiments, polarity value and semantics.
- I recommend you to use it with POS tagger(such as Kkma).

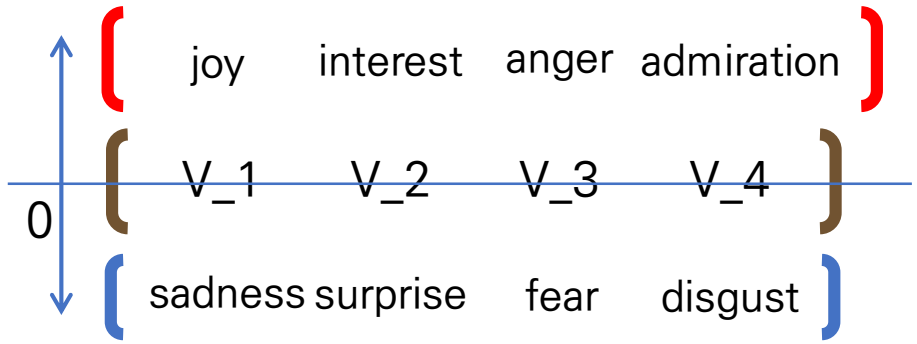
Building Process



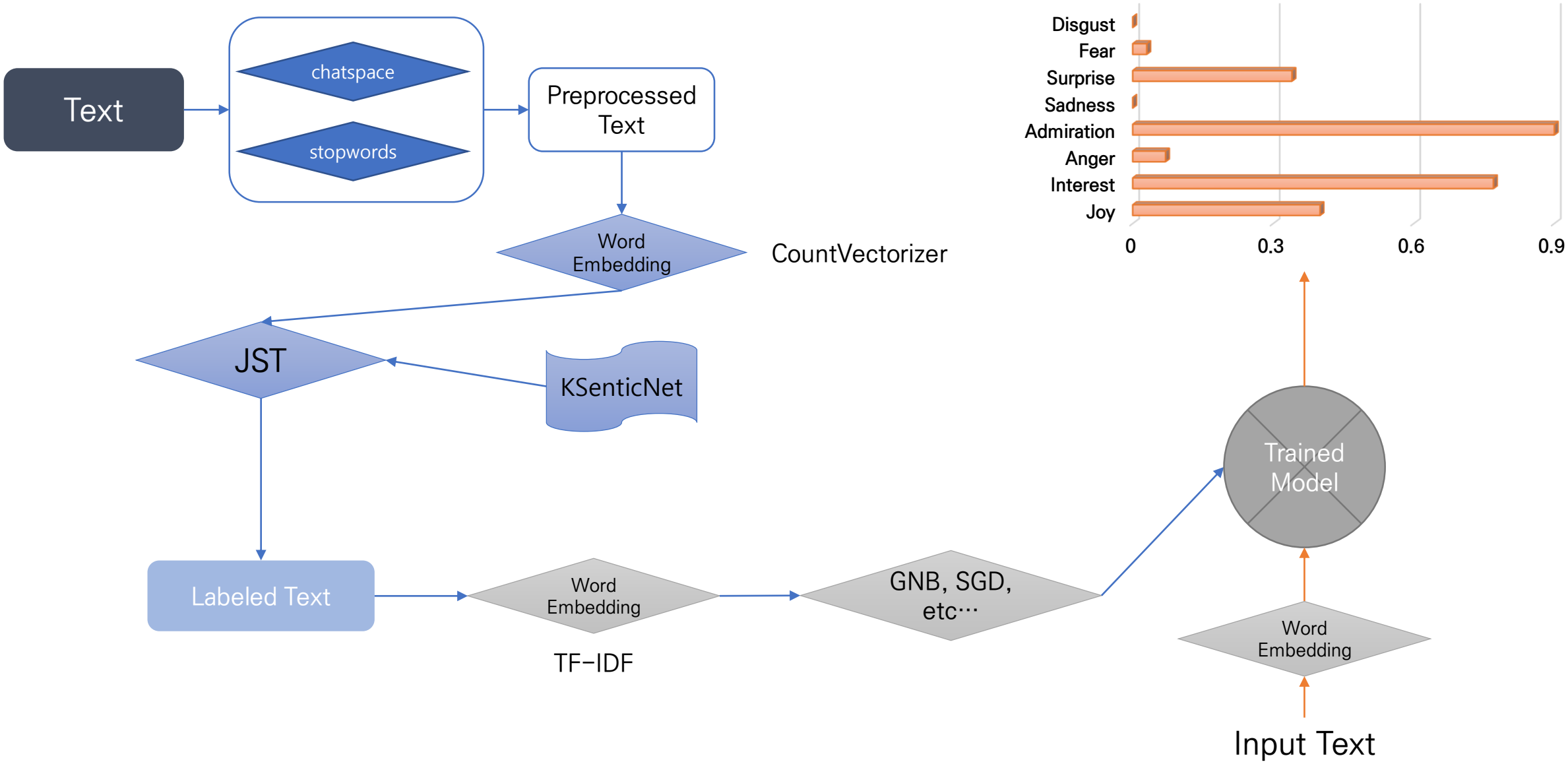
단어	Sentic_values
'예제1'	[0.70, 0.50, −0.90, 0.65]
'예제2'	[0.00, 0.23, 0.70, −0.29]

'예제1' : ['fear', 'joy']

'예제2' : ['anger', 'disgust']

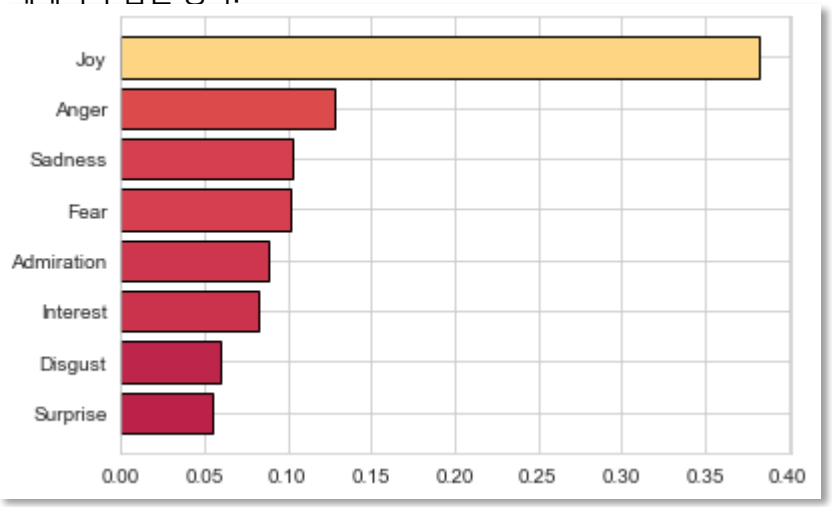


Process

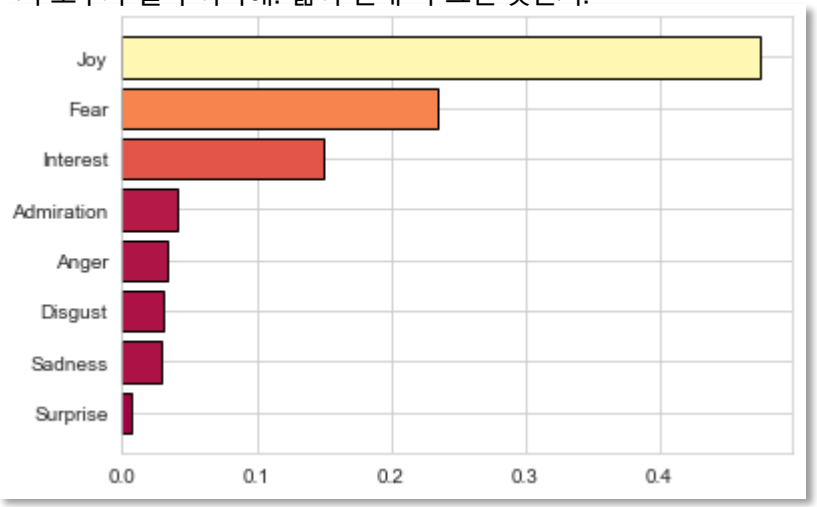

Text
chatspace
stopwords
Preprocessed Text
Word Embedding
CountVectorizer
JST
KSenticNet
Labeled Text
Word Embedding
TF-IDF
GNB, SGD, etc…
Trained Model
Word Embedding
Input Text
Disgust
Fear
Surprise
Sadness
Admiration
Anger
Interest
Joy
0
0.3
0.6
0.9

결과 시연 – Joy

로버트 패틴슨의 탐욕스런 눈빛.... 내가 다 불편하고 역겹게 느껴질 지경이었다. 연기가 좋았다 모든 배우가. 오락성으로 볼거라면 비추. 메세지가 담긴 영화.

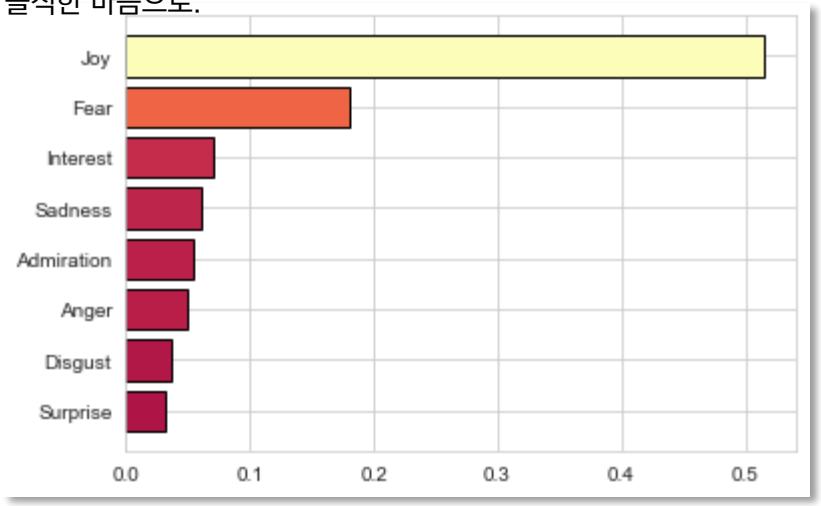


'우리의 어제 뿐만 아니라 오늘을 보게 된다. 등장하는 주요 인물들 모두에게 공감하게 된다. 현실과 욕망이라는 거대한 소용돌이 속에서 모두가 결국 아파해. 삶이 원래 다 그런 것인지.'

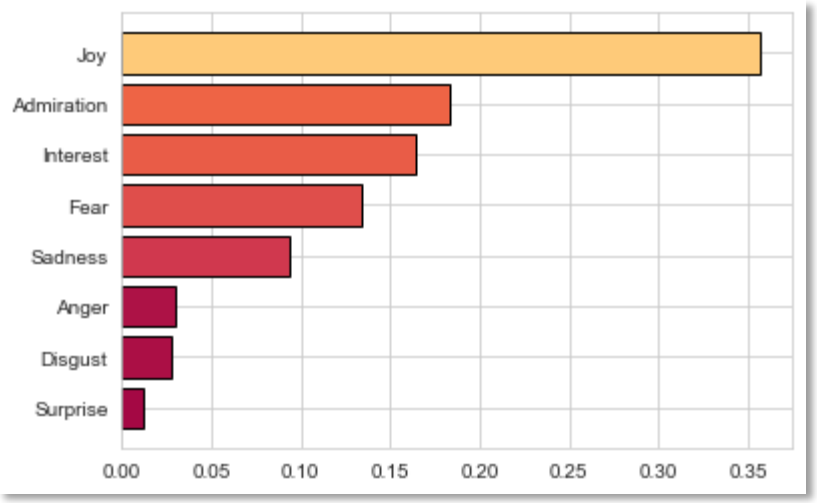


 # Joy

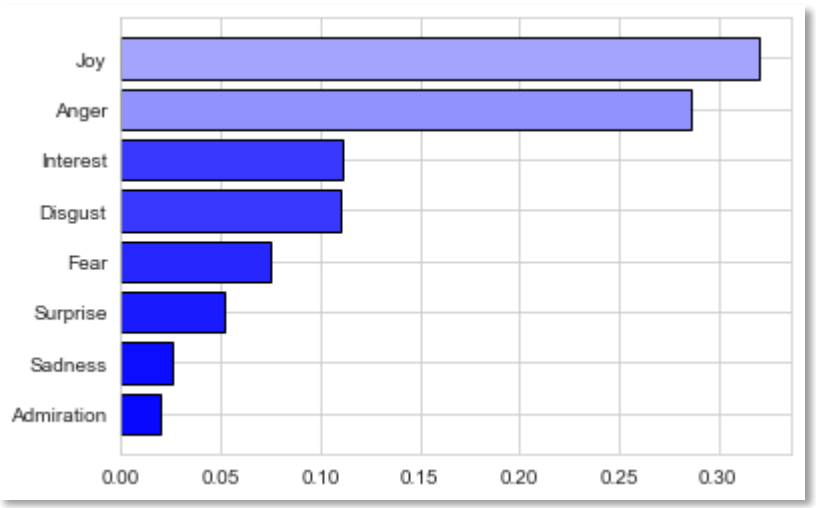
남자와 여자가 서로 믿음과 신뢰로 유혹과 맞선다지만 남자는 역시 더 힘든 건가..? 전반적으로 현실적인 면이 큰 영화라 잘 본 거 같다 솔직한 마음으로.



너무나 아름답고 쓸쓸한 영화... 나도 외롭다 _흑. 보는 내내 취한 듯이 보고 있었다

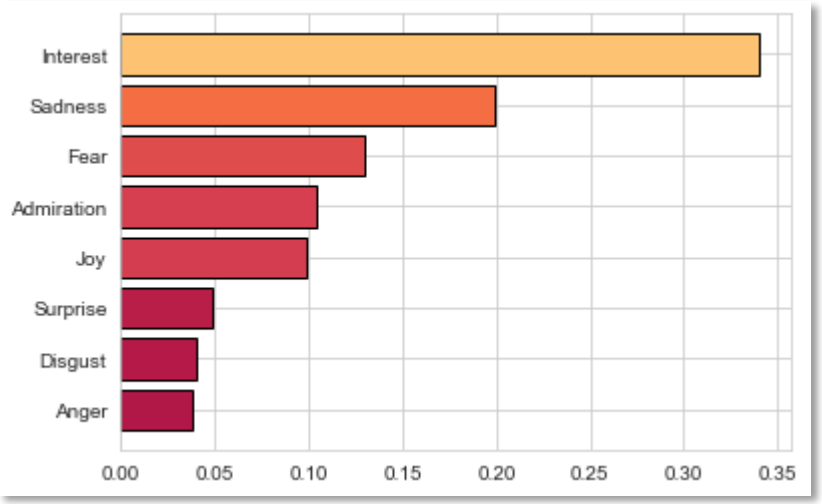


진부한 건 그렇다 치고, 너무 뜬금없이 전개되는 내용 때문에 도무지 공감이 안 간다

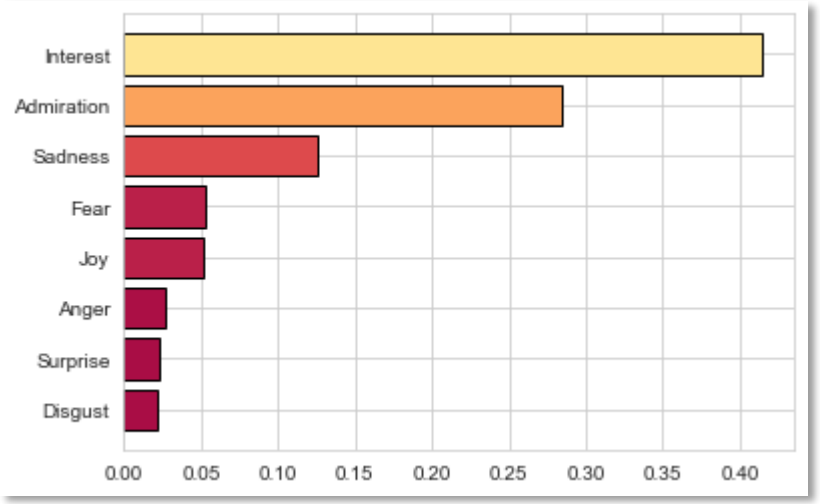


결과 시연 – Interest

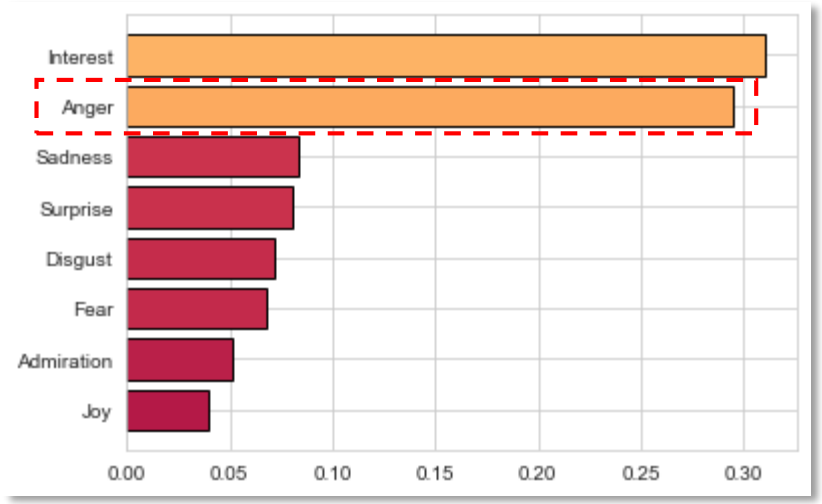
이건 진짜 길이 길이 남겨서 봐야 할 영화다. 언제 또 이 동굴을 볼 수 있을텐가. 우리 시대에 이것을 볼 수 있는 건 행운이다.



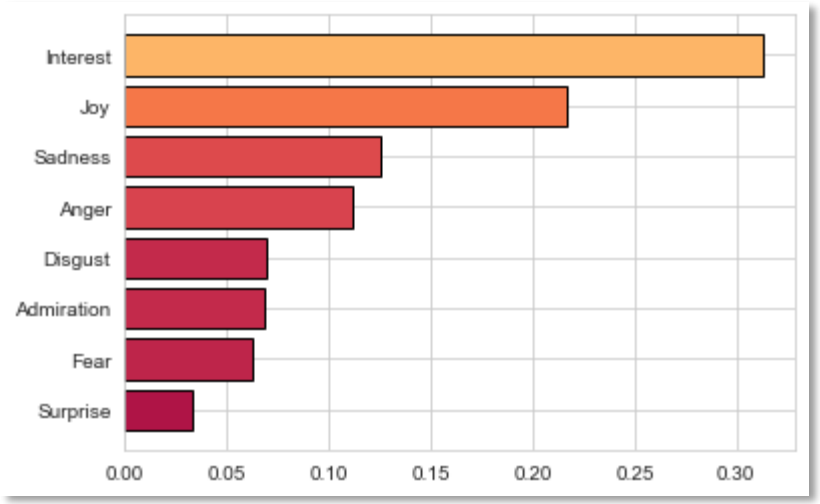
영화 보면서 울지 않는데... 슬퍼서 눈물 펑펑 쏟았네요.. 실화라서 더 감동적이네요..



반지의 제왕때를 생각하며 기대하고 봤는데, 전혀 실망감이 없었네요. 역시라는 말밖엔...



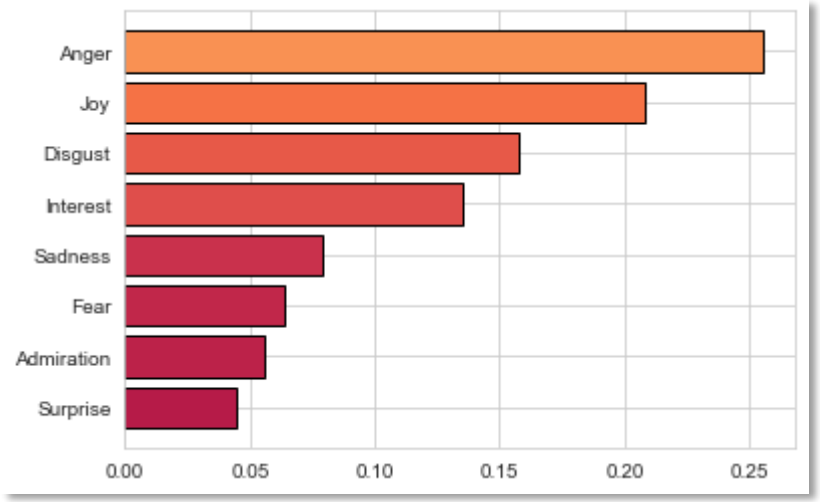
조금 더 천천히, 조금 더 느리게.. 그래서 조금 더 촉촉히 젖어드는 감성



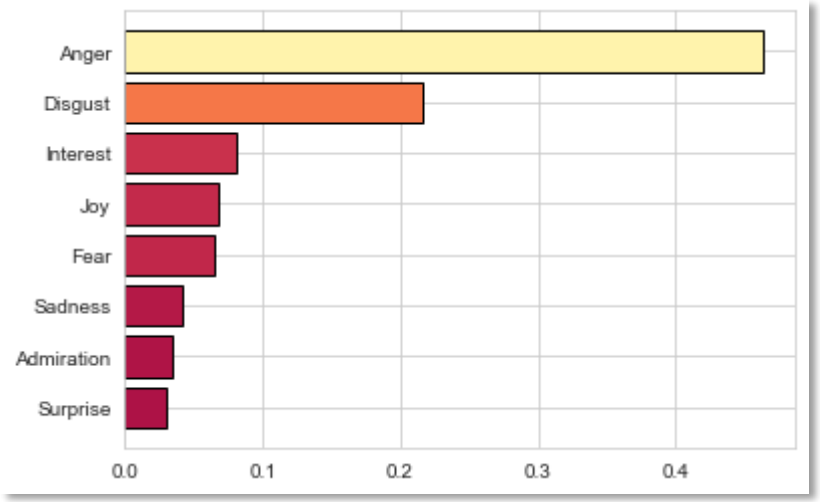
 # Interest

결과 시연 – Anger

10점 짜리 줄 영화는 절대 아닌 듯. 중간 중간 불 필요한 장면도 많고 급하게 끝내버리는 잔혹판타지 스릴러

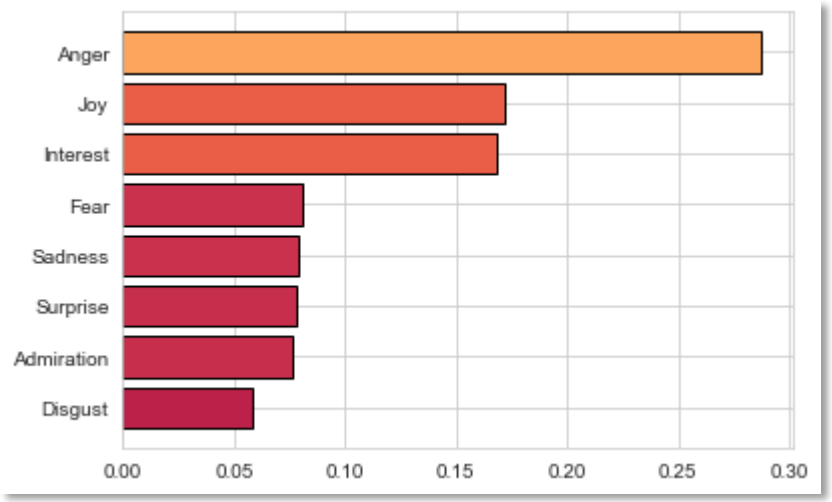


차라리 귀신 코드로 갔음 좋았을걸... 피부에 모가 나면 병원을 가야지. 어울리지 않게 왠 시답잖은 바이러스 좀비모드냐..

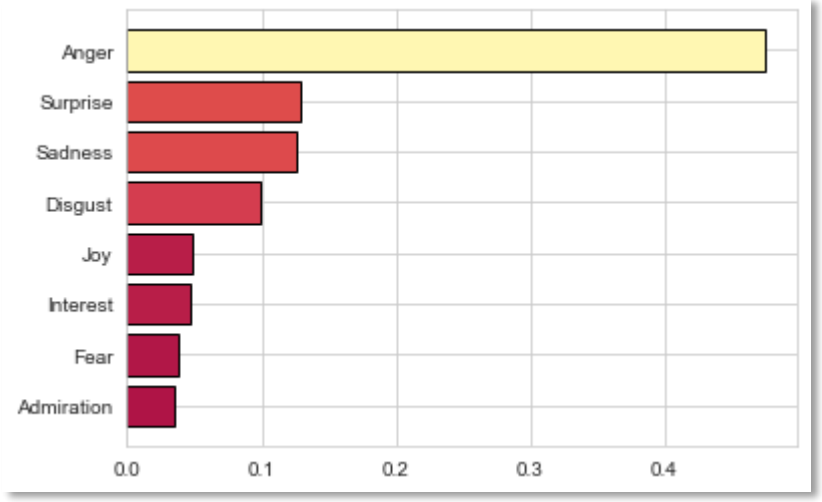


 # Anger

얇은 편지를 택배로 보낸 것 같다. 대문은 화려하지만 정작 영화의 스케일은.....

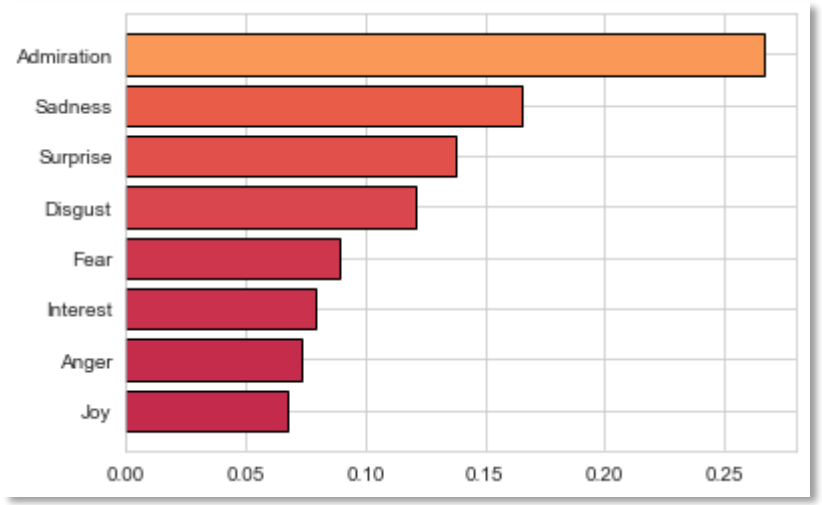


기대에 비하면 매우 실망스러운 영화이다.

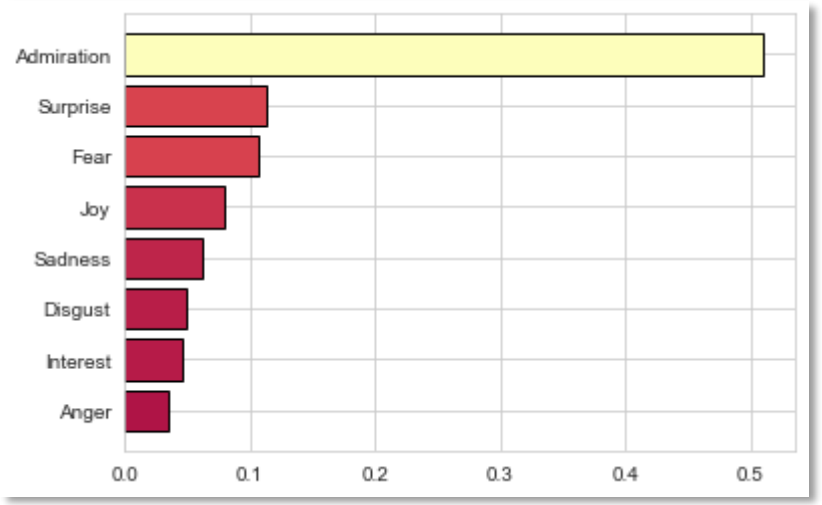


결과 시연 – Admiration

아... 저.. 이 영화 정말 너무 따뜻했어요.... ㅠㅠ

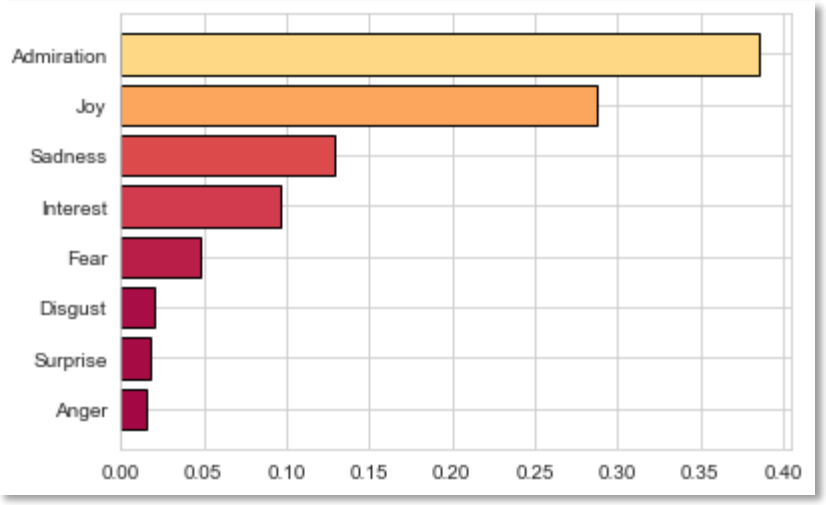


무슨 말이 더 필요하나? 최고다! 연출가 배우들은 이 드라마 시즌 제 끝가지 책임져라. 당신 들을 평생 응원하겠다!

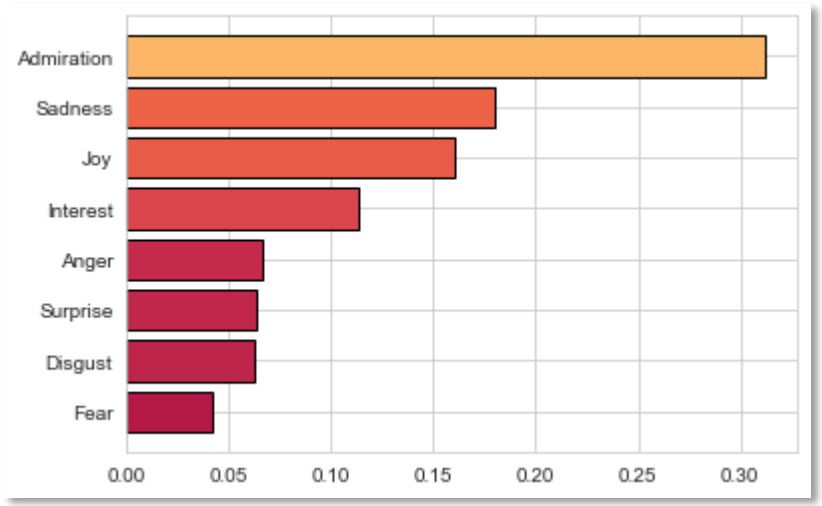


 # Admiration

와.. 정말 감동 있고, 개와 사람의 사랑과 우정, 믿음을 따뜻하게 보여준 영화 같아요. 정말 추억에 남을만한 영화입니다.

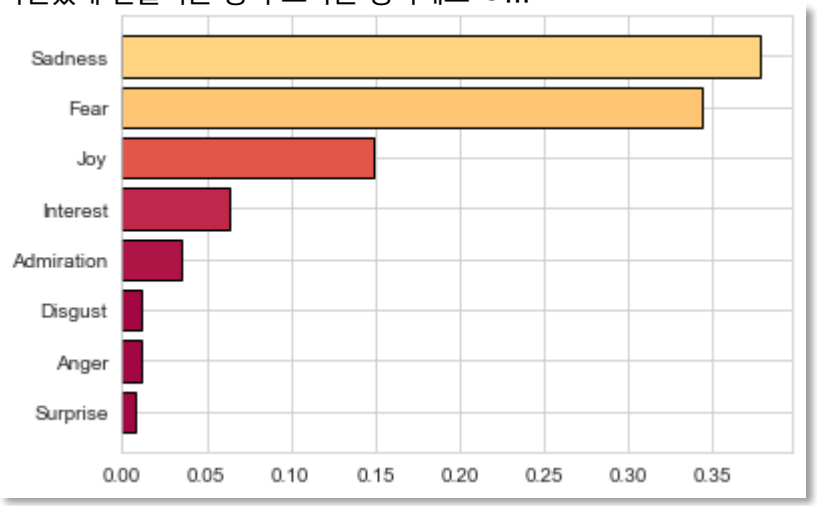


뮤지컬의 열정이 그대로 전해지는 듯~ 신선한 충격이었음..ㅎㅎ;;

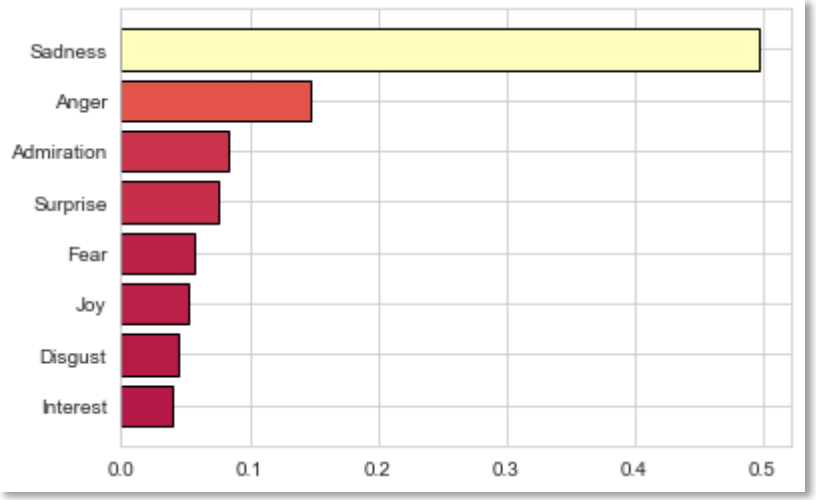


결과 시연 - Sadness

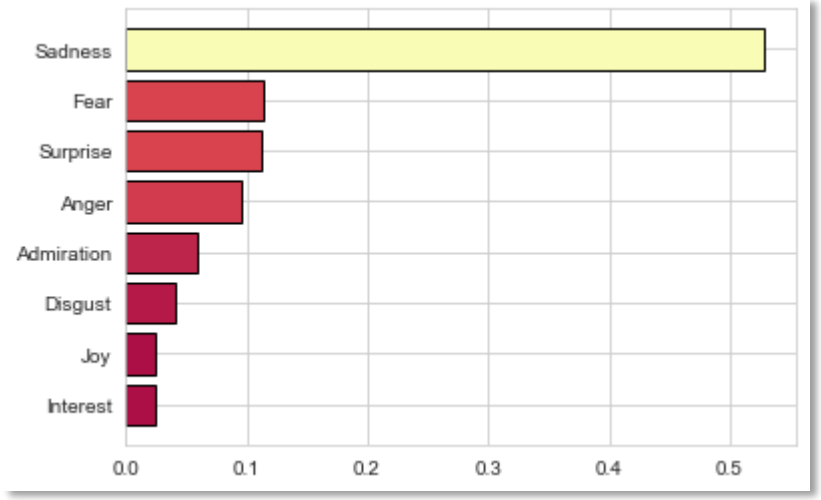
우연히 보게 된 영화 그러나 후회하지 않을 수 있게 해주는 영화 내 삶에 꼭 해보고 죽을 일 들을 만들어준 영화 그리고 죽음 앞에 덤덤할 자신있게 만들어준 영화 고마운 영화에요 ㅎ...



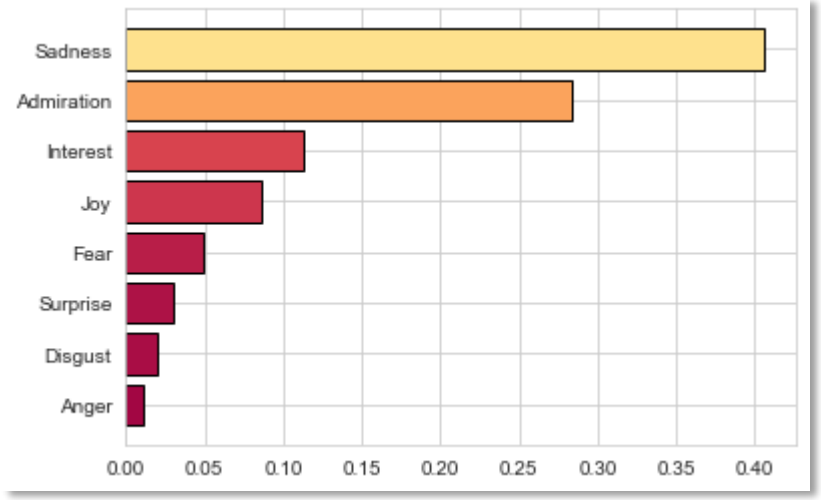
반개도 아까운 OOO 졸작 앞으로 장담컨데 이감독 영화 못만든다 왜 이렇게 좋은 배우들로 말도 안되는 졸작을 만들었으니 허점투성이다 . 리얼리즘이라고는 찾아볼 수 없다. 보는 네 네 짜증나 죽는 줄 알았다. 최근본 영화중 가장 졸작



아니 이걸 영화라고 만든거야? 화나게 하네 이렇게 말도 안되게 만드냐 진짜 화난다



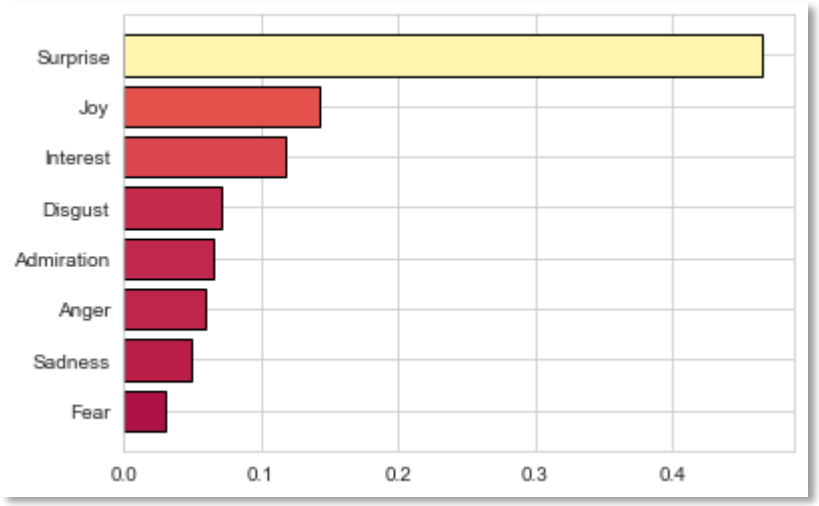
정말 보는 내내 눈물이 앞을 가리네요.. 최고의 감동이 느껴지는 영화였습니다 감사합니다!



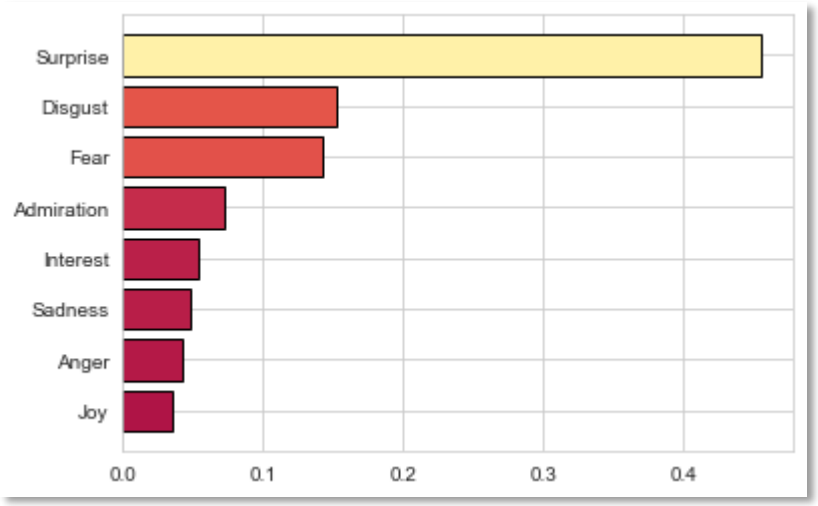
 # Sadness

결과 시연 – Surprise

히카리, 에이타 연기만으로도 10점.. 사카모토 유지 극본도 훌륭함

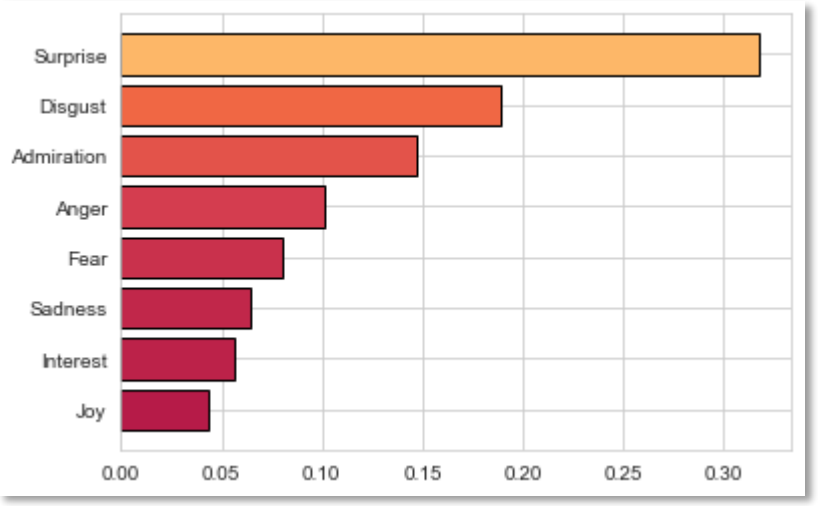


개인 취향이겠지만 생각없이 케이블에서 봤다가 재밌어서 놀랬다.
마사미 와.. 진짜 여신이구만 너무 이쁘다

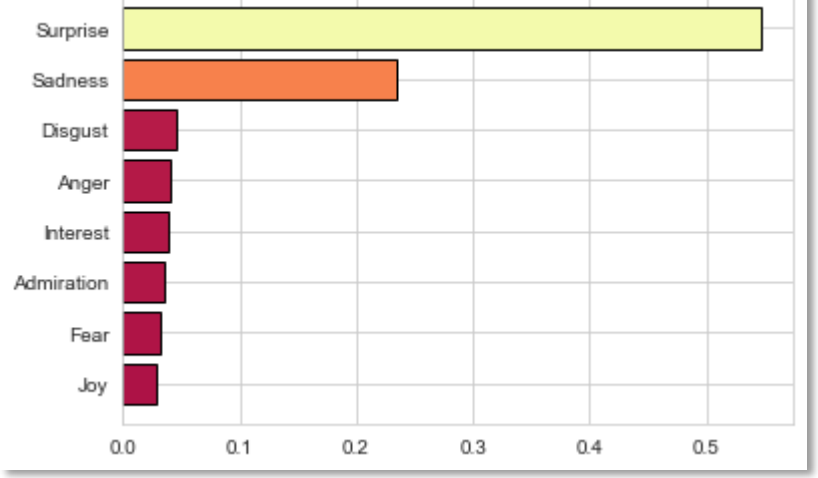


 # Surprise

후기들 보니 15세 관람가? 그건 무리인 거 같은데 ㅋㅋㅋ 역시 탐 크루즈의 코믹 연기는 명불허전...

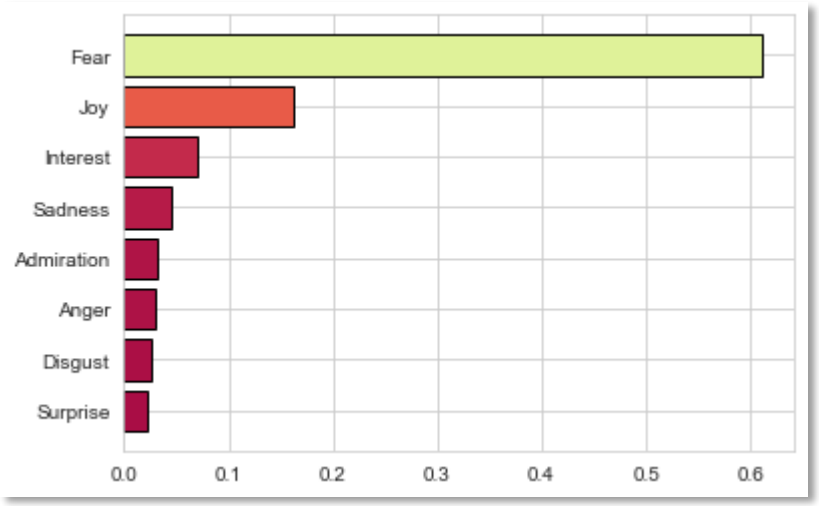


진짜 영화평점에 다 5점 이하로 주는 인간들 제정신인가? 이 정도면 솔직히 중박정도의 영화라 생각함 배우 제작사도 그쯤 으로 겨냥하고 만든 것 같고 난 좋습니다 개봉한지 1년이 넘어서야 봤지만 정확히 딱 8점 인듯 이민정씨 다음 영화 기대해요!

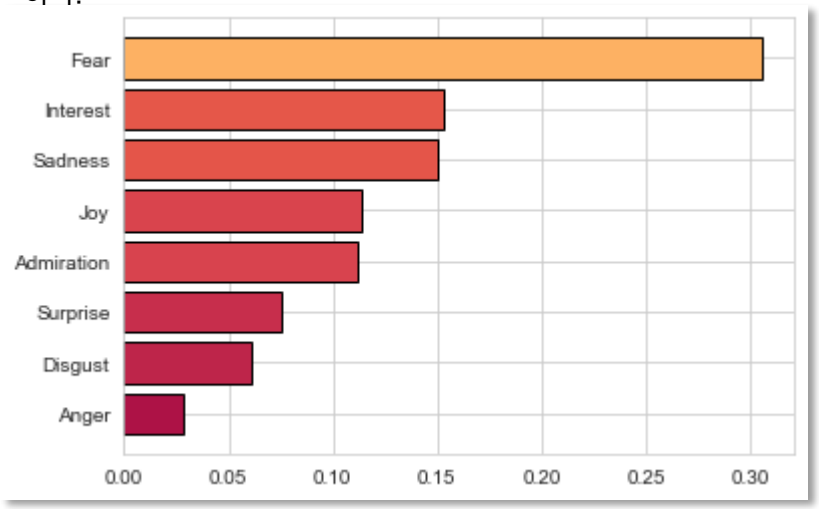


결과 시연 – Fear

당신의 의식 속에서 진실이라고 믿는 것은, 모두 진실이거나 진실이 된다.

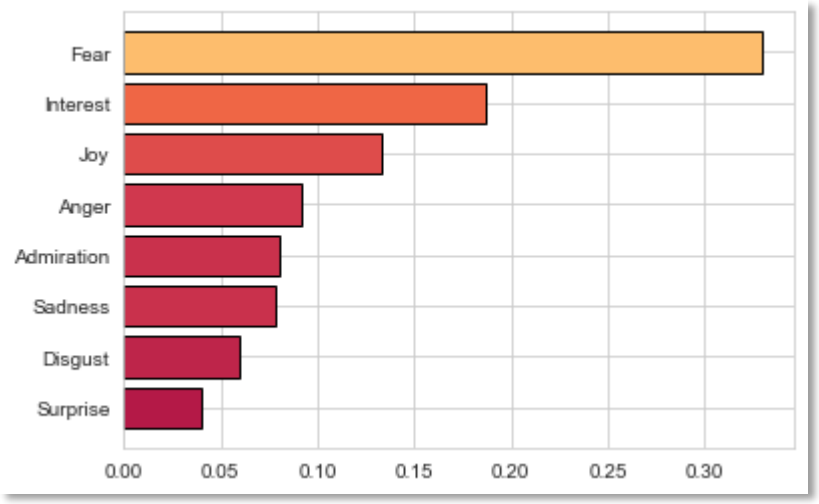


나만 볼수는 없다. 니들도 봐라. ㅋㅋㅋ 성인이 되어 초딩시절을 되볼아 보며 아직도 미성숙해 있는 주인공들을 보게 된다. 참 무의미하다.

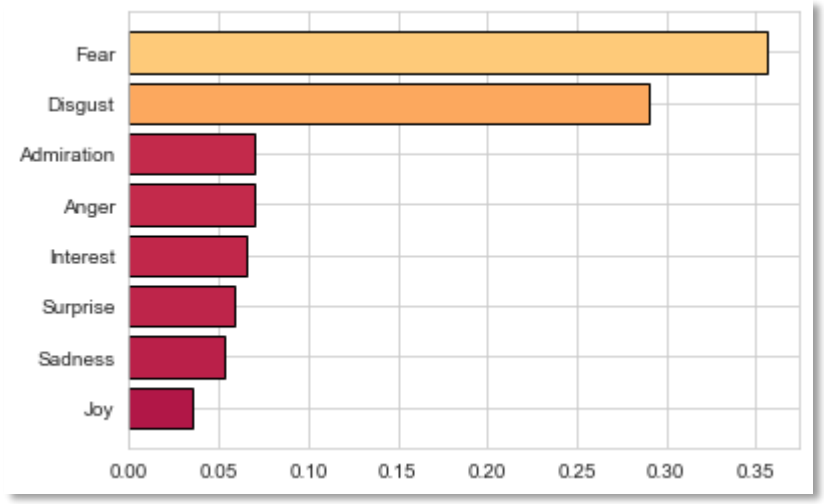


 # Fear

백구가 죽음으로 써 그에게 남아있던 온정은 사라 저버렸다.

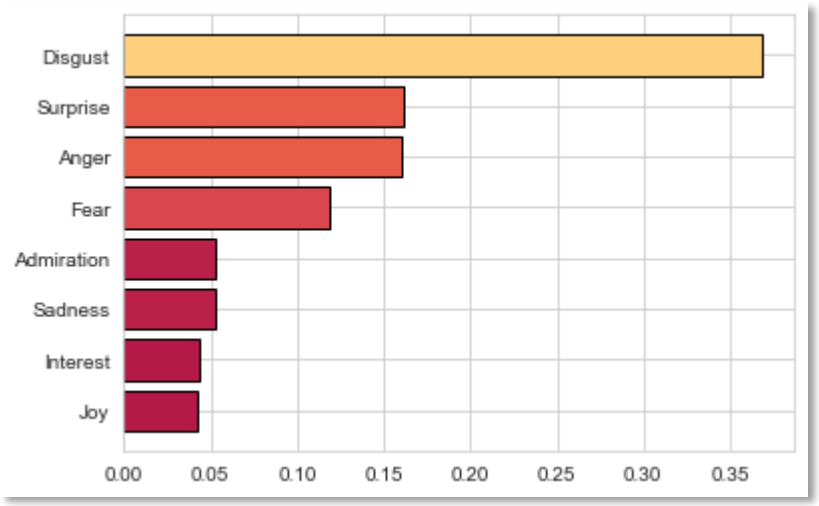


새벽에 혼자서 불 다 끄고 봐 보셈 걍 디짐 ㄷㄷㄷ 아! 소리 완전 크게 하고 봐야 함

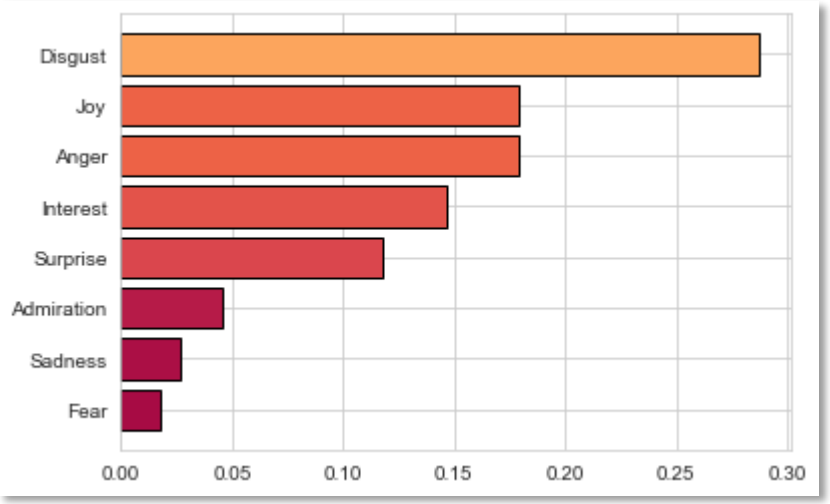


결과 시연 – Disgust

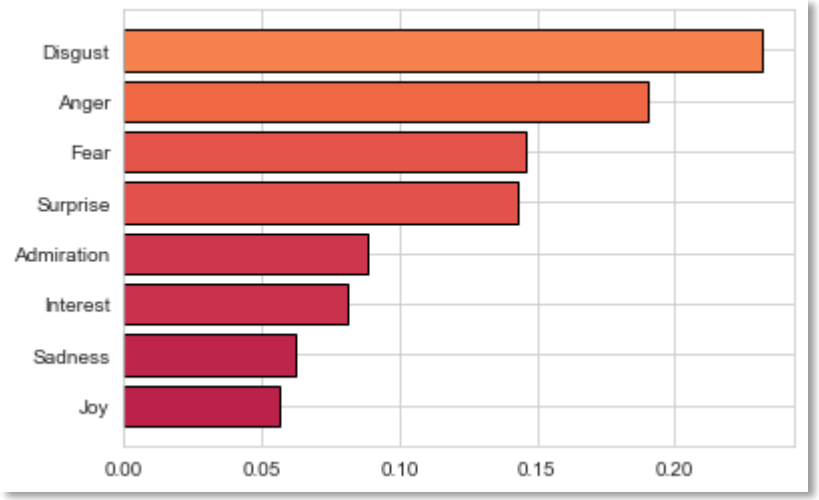
장난해? 솔트도 여자 제이슨 본이라고 하더만 그거보다 못함



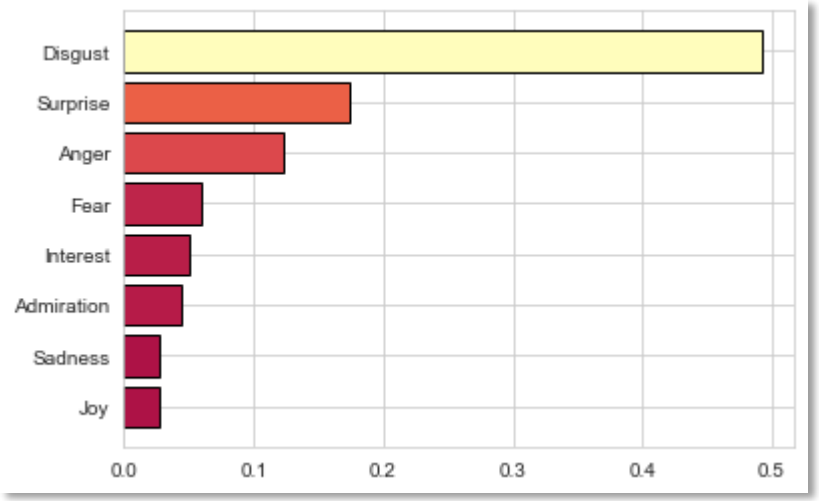
스토리도 좋고 내용도 좋지만 후반으로 갈수록 뻔한 스토리 ,.



코미디라서 봤는데... 나만 그런지 모르겠는데 진짜 재미없더라...



서운 장면 안 나온다면서 엄청 나왔 완전 —— 낚였네 그래도 무서웠다



 # Disgust

문제점 – JST

JST

```
class SentimentLDAGibbsSampler:

……
    def run(self, reviews, st, maxIters=30, saveAs=None, saveOverride=False, do_preprocess=True):
        self._initialize_(reviews, st, saveAs, saveOverride, do_preprocess)
        numDocs, vocabSize = self.wordOccurenceMatrix.shape
        for iteration in range(maxIters):
            gc.collect()
            print('Starting iteration {} of {}'.format(iteration + 1, maxIters))
            for d in range(numDocs):
                for i, v in enumerate(word_indices(self.wordOccurenceMatrix[d, :].toarray()[0])):
                    t = self.topics[(d, i)]
                    s = self.sentiments[(d, i)]
                    self.n_dt[d, t] -= 1
                    self.n_d[d] -= 1
                    self.n_dts[d, t, s] -= 1
                    self.n_vts[v, t, s] -= 1
                    self.n_ts[t, s] -= 1

                    probabilites_ts = self.conditionalDistribution(d, v)
                    if v in self.priorSentiment:
                        s = self.priorSentiment[v]
                        t = sampleFromCategorical(probabilites_ts[:, s])
                    else:
                        ind = sampleFromCategorical(probabilites_ts.flatten())
                        t, s = np.unravel_index(ind, probabilites_ts.shape)

                    self.topics[(d, i)] = t
                    self.sentiments[(d, i)] = s
                    self.n_dt[d, t] += 1
                    self.n_d[d] += 1
                    self.n_dts[d, t, s] += 1
                    self.n_vts[v, t, s] += 1
                    self.n_ts[t, s] += 1
```

1. 초기화 작업 수행
2. Gibbs Sampling
3. 확률분포 계산 후 할당
4. 2~3 과정을 Iteration만큼 반복

```
print('--* KSenticNet으로 사전 확률 조작 중... *--')
# 감정 사전 (KSenticNEt)을 사용하여 사전 확률을 조작 중.
for i, word in enumerate(self.vectorizer.get_feature_names()):
    w = KSenticNet.keys.get(word)
    if not w: continue
    synsets = KSenticNet.scores[w, :]
    self.priorSentiment[i] = np.random.choice(
            self.numSentiments, p=synsets)
```

```
def conditionalDistribution(self, d, v):
    probabilites_ts = np.ones((self.numTopics, self.numSentiments))
    firstFactor = (self.n_dt[d] + self.alpha) / \
      (self.n_d[d] + self.numTopics * self.alpha)
    secondFactor = (self.n_dts[d, :, :] + self.gamma) / \
      (self.n_dt[d, :] + self.numSentiments * self.gamma)[:, np.newaxis]
    thirdFactor = (self.n_vts[v, :, :] + self.beta) / \
      (self.n_ts + self.n_vts.shape[0] * self.beta)
    probabilites_ts *= firstFactor[:, np.newaxis]
    probabilites_ts *= secondFactor * thirdFactor
    probabilites_ts /= np.sum(probabilites_ts)
    return probabilites_ts
```

문제점 – JST

In [2]:
```
JST.run(processed_reviews, okt, maxIters=30, do_preprocess=False)
```

Out [2]:
```
--* KSenticNet으로 사전 확률 조작 중... *--
Done.
--* initialize 작업 진행 중...          *--
    Doc  10,000 of 712,383 Reviews
    Doc  20,000 of 712,383 Reviews
              ...
    Doc 710,000 of 712,383 Reviews
Done.
--* Gibbs Sampling JST 실행             *--
    Starting iteration  1 of 30
    Starting iteration  2 of 30
              ...
    Starting iteration 30 of 30
Done. 모든 작업이 끝났습니다.
```

무엇보다 여배우의 거대한 가슴에 아낌없이 평점 10점 만점을 주고, 또 주고 싶다. (웃음) 솔직히 세트장도 정말로 인상적. 우리나라에선 죽었다 깨어나도 생각해 볼 수 없는, 탁월한 발상의 무대장치와 세트에 경탄과 찬사를 보냅니다. 정말 차원이 다르다는 말로도 부족하군요	['joy', 'surprise']
진짜 잼잇게 보고 조마조마하고 잼잇네여 마녀쌤 Vs 학생들 굿 여왕의 교실 2가 하면 좋겟네용 ㅎ	['admiration', 'joy']
부천 1분만에 매진? ? ? ? 아주 그냥 입만 열면 거짓말이야 일본 교과서나 빨리 수정해라 독도가 누구땅? 한국땅	['fear', 'sadness']
스토리가 뛰어난 건 아니지만 감각적인 영상과 뛰어난 음아으로 충분히 커버한다. 83년도 작품이라는 게 놀랍다	['surprise', 'interest']
아무리 감동적인 영화라도 나이 많은 환자들만 나오는 영화를 보는 건 좀 힘들다!	['disgust', 'sadness']

문제점 – JST

- 문장의 길이가 짧을 경우 감정 군집화가 제대로 수행되지 않았다. / 맥락을 파악할 수 없다.

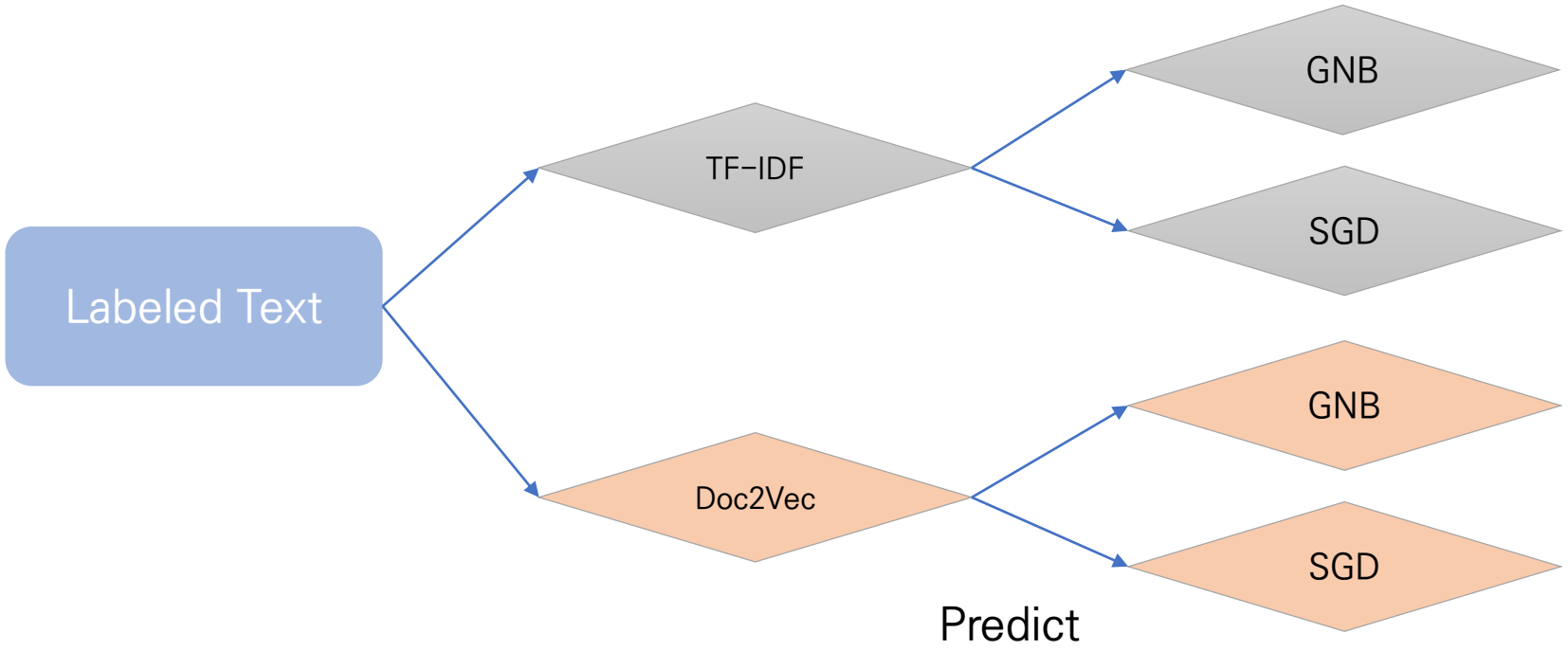
1. 뭐, 이 정도면 괜찮은 거 아닌가?	['disgust', 'fear']
2. 짱 좋아 너무 좋다 다시 나왔으면 좋겠어! !! ! ! ! !!!! !	['surprise', 'anger']
3. 시작은 좋은데 중간에 너무 지루ㅠ	['joy', 'admiration']
4. 나름 나름	['disgust', 'anger']
5. 그녀의 노래를 제외하곤 별로 남는 것이 없다.	['joy', 'interest']
6. 이거 겁나 재밌게 봤는데 ㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋ	['disgust', 'surprise']
7. 앞부분은 조금 지루했지만 좋은 교훈이 있었다	['disgust', 'sadness']

- 맥락을 이해하지 못하고 특정 단어에 비이상적으로 반응하는 경향을 보인다.

1. 사형도 수의 복장과 동일한 점이 있다는 것이 흥미로움.	['anger', 'joy']
2. 영상이 예쁘네요. 음악도 참 좋고요. 나중에 다운로드 받아서 한번 더 보고 싶습니다.	['surprise', 'disgust']
3. 너무 없이 없는 설정.. 그리멍청한 학생들과 말도 안되는 군인들.. 아무리 군대개판이래도 이건 너무 심했다. 감독이 군필인지가 의심스럽다.	['admiration', 'interest']

- Labeling이 제대로 수행되었는지 정확한 성능 평가가 어렵다.
  - ✓ Perplexity, K-folds CV로 정확한 성능 측정이 필요해 보인다.

문제점 – 분류 모델



Accuracy	Recall
49.0%	48.5%
49.3%	49.2%
13.4%	12.5%
12.9%	12.4%

Real

Joy
Interest
Anger
Admiration
Sadness
Surprise
Fear
disgust

```
array([[12158,  1280,  2130,  1469,  1559,  2371,  2143,  1650],
       [ 2435,  7369,  2433,  1547,  1477,  1502,  2124,  2156],
       [ 1662,   855, 13237,   865,  1504,  2116,  1615,  3058],
       [ 2087,  1058,  1502, 12284,  1697,  2653,  2004,  2614],
       [ 1542,   944,  1932,  1721, 11057,  2026,  1892,  1403],
       [ 1041,   577,  2000,  1747,  1441, 14917,  1298,  3934],
       [ 2506,  1069,  1967,  1509,  1939,  2260, 11924,  2767],
       [ 1198,   840,  3415,  1457,  1114,  3419,  2166, 15733]],
      dtype=int64)
```

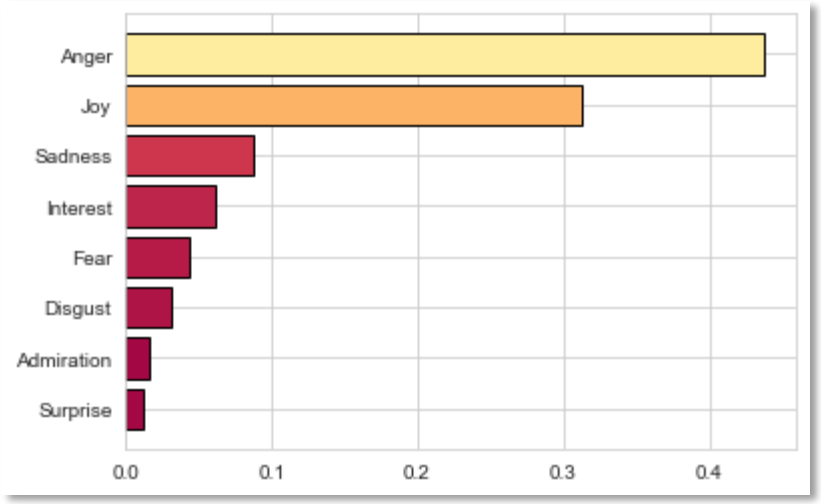
53.619385 %
45.965846 %
55.340382 %
49.105121 %
47.430403 %
53.135035 %
35.018771 %
49.103393 %

문제점 - 분류 모델

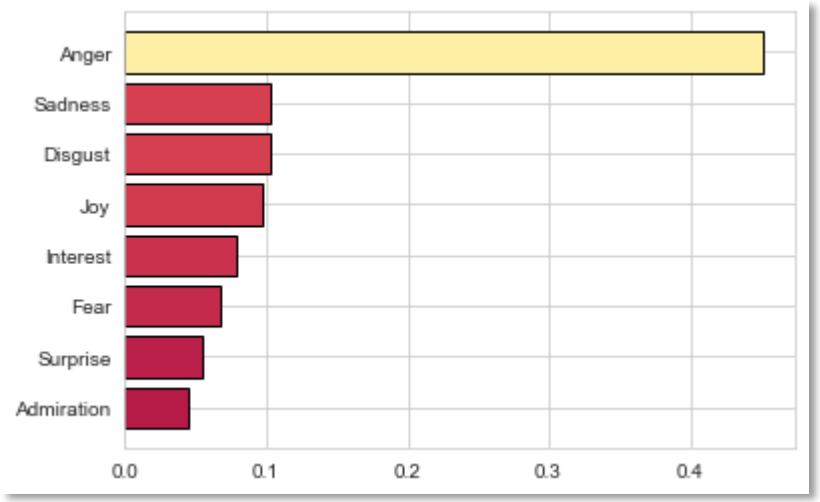
	review	Joy	Interest	Anger	Admiration	Sadness	Surprise	Fear	Disgust
119208	재미게봤다 전작을안봐서잘모르지만스토리보다배드신위주로보는데이건전부다봤다 스토리도b급영화치고탄탄하고 밑에놈들이랑다르게여주매력있게생겻다이쁜여잔아닌데매력있었다.. 그리고중간중간 가슴골이보이는거같은은꼴재밌게봤다 1편도한번봐봐야겟다	5.59%	5.29%	10.55%	5.26%	4.60%	15.92%	9.45%	43.34%
411527	지츠니 오모시로이!	6.99%	7.50%	8.88%	35.52%	9.60%	8.34%	13.50%	9.67%
575334	조혼나 재밌었다 조혼나	10.56%	10.62%	12.07%	10.52%	6.65%	14.25%	8.20%	27.14%
95526	진짜다잘하고재밌는데근데짜리몽땅은약간가수가아니라합창이더어울리는거같다만약쟤네가우승한다면가수로서할수잇는장르를어떤걸해야할지발라드랩힙합댄스;;가수는아닌듯진짜잘하긴하는데가수는아닌거같다약가알멩이나버나드박이가수로서가능성재능이잇는듯	7.97%	8.12%	14.03%	18.35%	10.50%	17.43%	11.94%	11.65%
17351	전 이거 볼빠엔 차라리 트랜스포머4를 보겠어요	5.01%	17.38%	11.95%	7.67%	8.39%	14.34%	16.90%	18.35%
137675	영화 배급사 ㅅㄲ들 진짜 제목낚시 불법화 못하나? 원제는 빅풋인데 왜 아무 상관 없는 혹성탈출 이름을 쓰냐? 진짜 영화에 대한 명예훼손이다.	3.36%	4.74%	34.95%	5.08%	13.81%	8.08%	16.66%	13.32%
637004	뻔하고 지루했어요... 하지만 엄마는 위대하다!	31.88%	8.69%	18.52%	9.06%	3.20%	4.49%	10.18%	13.98%
638779	여자주인공 섭외잘못한듯..너무나 갸냘프고 여리여리해서영화캐릭에 안어울림 . 영화에 집중이 안됨..좀더 카리스마있는여배우였으면 좋았을텐데.	20.26%	4.79%	13.78%	7.12%	3.64%	29.03%	5.11%	16.28%
46602	너무재밓었다그래서보는것을추천한다	13.95%	7.51%	7.41%	11.12%	19.51%	7.97%	24.08%	8.44%
447256	동성애 예찬 영화? 천재의 고뇌? 도대체 무엇을 말하고 싶은 건지 알 수 없는 영화.	15.64%	9.75%	20.87%	3.69%	12.00%	1.34%	35.11%	1.60%
404467	좀 아쉬웠지만 귀는 심심하지 않았다.	11.89%	15.30%	14.93%	10.76%	5.89%	8.51%	6.01%	26.70%
146225	이거 생각지도 않은 수작이다 무협의 재미를 충분히 만끽하는 작품	9.03%	8.77%	15.02%	7.51%	41.30%	9.05%	2.88%	6.44%
31242	어렸을때 봤는데 야한장면 나올때 낯뜨거웠음.	9.92%	20.95%	6.32%	7.66%	9.15%	13.72%	14.55%	17.72%
32103	영화를보는동안미스터리하면서도 가슴이미어지는 영화였다 "이 사람들 이상해"라고 하는 남주인공이 알게 모르게 와 닿았다 사람이 그사람들을 평가하고 동물들을 죽이네 살리네 하는 인간의 기준이 뭔가 그게 타당 한가에 대해서 의구심이 들었고 한편으로 안타깝다	34.57%	12.29%	4.86%	8.12%	9.10%	1.68%	25.38%	4.01%

문제점 – 분류 모델

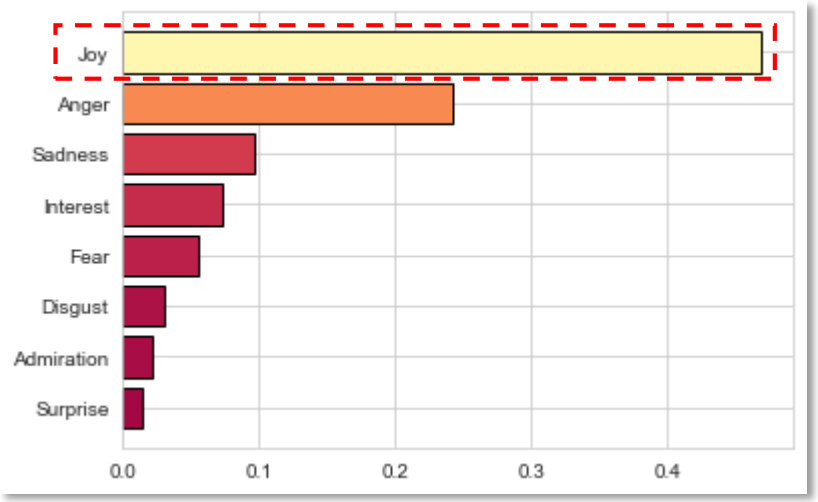
난 역시 프랑스와 맞지 않아... 그저 유럽 특유의 거침없는 표현 방식만 마음에 든다.



난 역시 프랑스와 맞지 않아...



그저 유럽 특유의 거침없는 표현 방식만 마음에 든다.



결과 분석 및 해결 방안

Text Sentiment Labeling

- Sentiment Labeling에 대한 적절한 성능 평가 필요 〉 Perplexity, K-folds CV
- Sentiment Dictionary 확장 및 Domain에 대해 상세하게 표현 〉 KSenticNet 어휘 추가
- Intent, Context를 파악하여 특정 단어 별 Weight를 변경하여 감정을 라벨링할 수 있게

  〉 BERT, XLNet의 Idea를 활용, 자체 API를 만들어 JST 방식을 맥락, 의도에 강건하게 구축

Text Sentiment Classification

- Doc2Vec은 Randomize하게 결과를 추출하는 12.5%(1/8) 성능을 보임

  〉 Labeling 방식에 맞게 Text를 Embedding해서 성능을 높여야 함 (ex) CBOW〉CBOW, Seq〉Seq
- TF-IDF Embedding은 각 label 별 50%로 군집화된 감정 재현율은 coin 던지기 정도의 결과를 보임

  〉 TF-IDF로 Embedding시 어떤 단어가 문장에서 중요한 지는 알 수 있지만 어떤 감성을 보일 지 구분할 사전 확률정보가 없음

  〉 Feature에 특정 감정일 확률을 제시하여 Labeling 결과 재현율을 높여야 함

  〉 Hybrid 모형으로 성능 향상을 도모 (ex) Topic별로 모형을 구축, Stacking한 후에 Resampling하여 Soft Voting

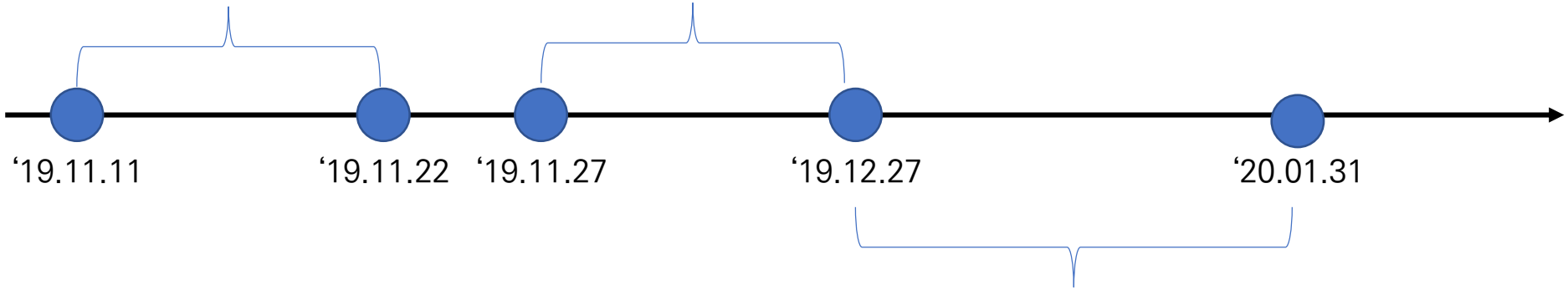
Visualization

- SHAP, LIME, Attention 기법 등을 활용하여 어떤 단어 혹은 맥락이 감정을 분류하는 데 유의한 feature로 사용 되었는지 제시해야 함
- 특히나 Attention, AdaBoost 기법을 활용하여 중요하게 여겨지는 단어들, 문맥에 가중치를 부여하는 방식으로 문제 해결을 도모

향후 계획 및 목표

향후 계획

- Senti. Labeling 성능 평가
- Hybrid Ensemble model 구축
- 결과 Table 정리 후 11.27 발표
- CBOW Labeling 성능 향상 계획 수립
- KSenticNet 감정 사전 보완
- 효율적인 Text Embedding 탐구



- 언어 모델 임베딩을 활용한 semi-supervised labeling 방법 개발
- XAI 모형을 활용하여 결과 시각화

향후 목표

- Sentiment Labeling 모형 성능 개선 및 Intent, Context, Aspect를 고려하여 Labeling
- 분류 모형이 Labeling 재현율을 80% 이상으로 구축
- 타 Domain의 Text에서도 강건하게 동작할 수 있도록 Text 수집 및 분석