

# 가짜 맛집 리뷰 판별

이석범

# 동기

음식점에서 직접 음식을 먹어보고 리뷰를 남기는 ‘맛집 리뷰’에 대한 수요와 공급이 많이 있는 상황

솔직한 리뷰도 있지만, 음식점으로부터 금전적 지원을 받고 쓰는 광고성 ‘거짓리뷰’ 또한 존재

리뷰를 구성하는 텍스트 정보만으로 진짜리뷰와 거짓리뷰를 구분할 수 있을 것이라 가정

# 데이터 마이닝

리뷰에 대한 데이터가 존재하지 않기 때문에 직접 수집해야 함

네이버 **API**와 **Python Beautiful-soup** 라이브러리를 이용해 네이버 블로그 맛집리뷰를 크롤링

리뷰가 진짜인지 거짓인지 **Label**이 달려 있지 않은 문제점  
-> 일정 기준으로 직접 **Labeling** 진행

# 데이터 마이닝

거짓 리뷰 : 금전적인 지원을 받았다는 언급이 존재하는 경우로 한정

서비스를 무상으로 지원받아 식사하고 솔직하게 작성한 포스팅 입니다.

본 포스팅은 소정의 원고료를 제공받아 작성하였습니다

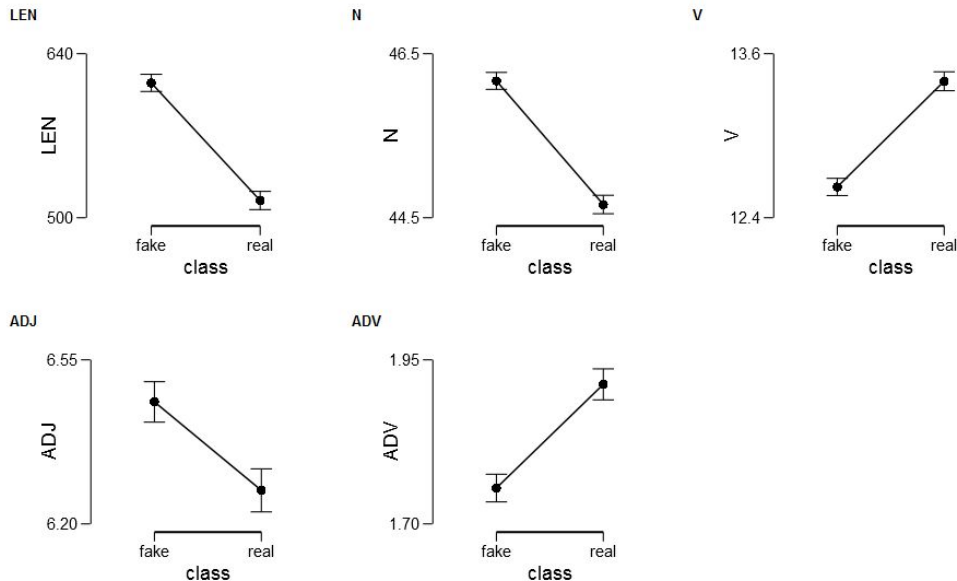
본포스팅은 업체로부터 서비스를 지원받아 작성하였습니다

진짜 리뷰 : ‘사비를 지출했다’, ‘직접 사먹었다’는 언급이 존재하는 경우로 한정

# 데이터 마이닝

수집한 두 Class 간 언어적인 차이가 존재하는지 검증

Descriptives Plots



리뷰의 길이, 품사의 비율에서 통계적으로 유의미한 차이가 존재했음

# 데이터 전처리

Labeling 에 사용된 문장은 전부 제거함

‘본 포스팅은 소정의 원고료~’, ‘제 사비로 직접 사먹었습니다~’

모든 문장을 단어의 (어근, 품사) 로 변환

‘오늘은 맛있는 피자집에 다녀왔어요~ ㅎㅎ’ ->

[('오늘', 'Noun'), ('은', 'Josa'), ('맛있다', 'Adjective'), ('피자', 'Noun'), ('집', 'Noun'), ('에', 'Josa'), ('다녀오다', 'Verb'), ('~', 'Punctuation'), ('ㅎㅎ', 'KoreanParticle')]

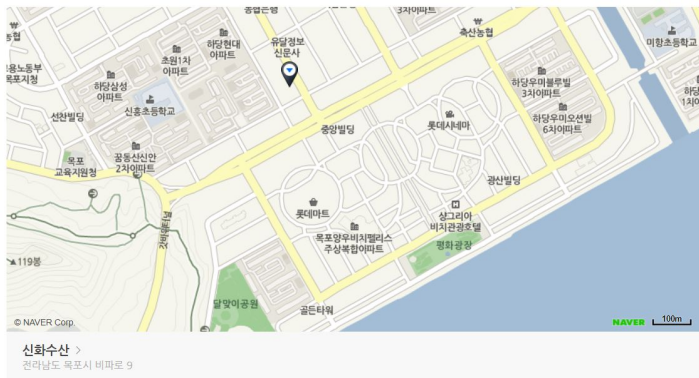
# 데이터 전처리

사진 -> PHOTO 토큰

이모티콘 -> IMO 토큰

지하철역(성대역) -> PLACE 토큰

상호명 -> REST 토큰 (리뷰의 네이버 지도에서 상호명을 유추할 수 있음)



# 각 리뷰를 벡터로 표현

Doc2vec

Tf-idf

Bow



# 분류

Naive Bayes

Random forest

KNN(K- Nearests - Neighbors)

SVM(Support Vector Machine)

기법을 활용해 5-fold-cross-validation

## 분류 결과

	SVM(rbf)	SVM(linear)	SVM(linear)	KNN	NB	RF
정확도	0.81	0.79	0.84	0.78	0.75	0.8
벡터화 방식	D2V	D2V	TF-IDF	TF-IDF	BOW	BOW

기다리다(V), 충주(N), 오빠(N), 백종원(N) 등 단어가 존재할수록 진짜 리뷰일 가능성이 높은 것으로 분석됨

# 결론 및 한계점

리뷰를 구성하는 텍스트정보 만으로 리뷰의 진위 여부를 분류하는 것이 가능

한계:

데이터를 주관적 기준으로 **Labeling** 했다는 점

지원을 받고 솔직하게 작성한 경우, 지원을 받았지만 사비를 지출했다고 거짓말하는 경우 존재할 수 있음

인터넷 은어, 축약어 등은 적절하게 **POS** 태깅 되지 않았을 가능성이 높음