

## 뉴스 감성분석을 통한 주가 예측



B111122 안정호  
B111278 마상혁  
B211259 경상수

## CONTENTS

# 목차

Part 01

서론

Part 02

기존사례 소개 및 문제점 분석

Part 03

해결 방안



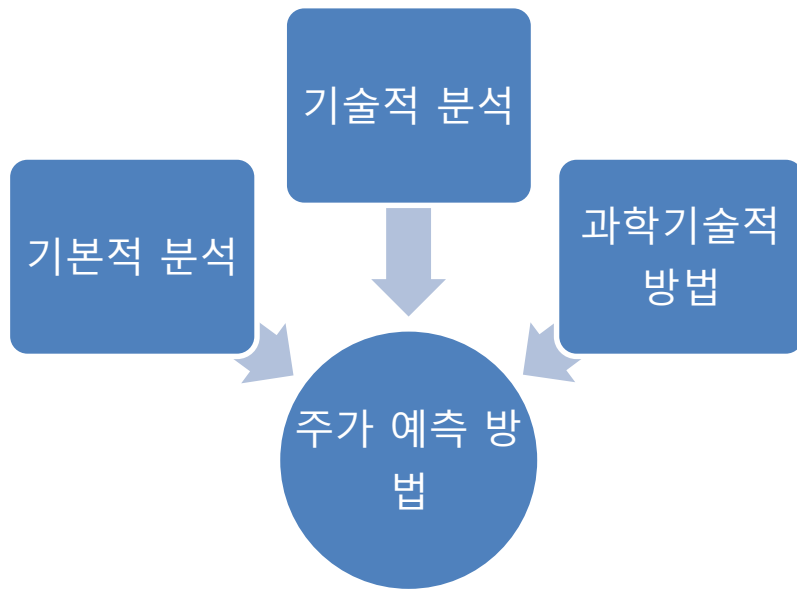
## 주가 예측 현황

- 사람들의 주식시장에 대한 관심이 증가
- 주가를 예측하기 위한 많은 연구가 진행

## 주가 예측 방법

- 기본적 분석: 기업의 과거 성과 기반
- 기술적 분석: 과거주식의 동향 기반
- 과학기술적 방법: 다양한 데이터를 활용

### <주가 예측 연구 방법>



## 연구 목적 및 방법

- 아직 완벽한 주가 예측은 난제로 남아 있는 상태
- 뉴스 데이터를 활용한 감성분석(=오피니언 마이닝)방법의 단점을 보완
- 주가 예측 방법을 제시

### <주가 예측 연구 절차>

폭넓은 주가 예측과  
관련된 기존 사례 분석

기존 사례의 문제 및  
한계점 도출

새로운 예측 방법 제시

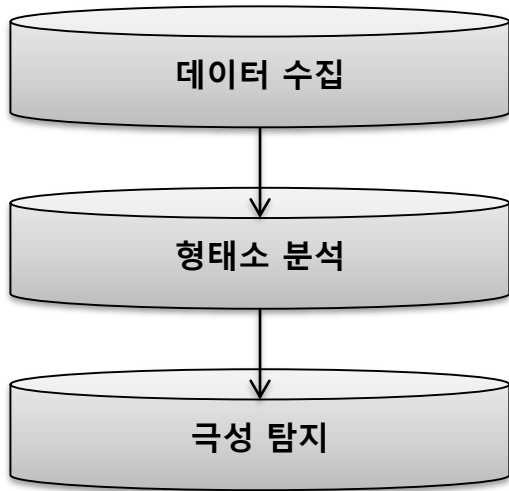
## 감성분석이란?

- > 수집된 데이터를 자연어 처리와 텍스트 분석을 이용해  
텍스트 내에서 주관적인 정보를 확인하고 추출하는 기법

### 1. 데이터 수집(뉴스 파싱)

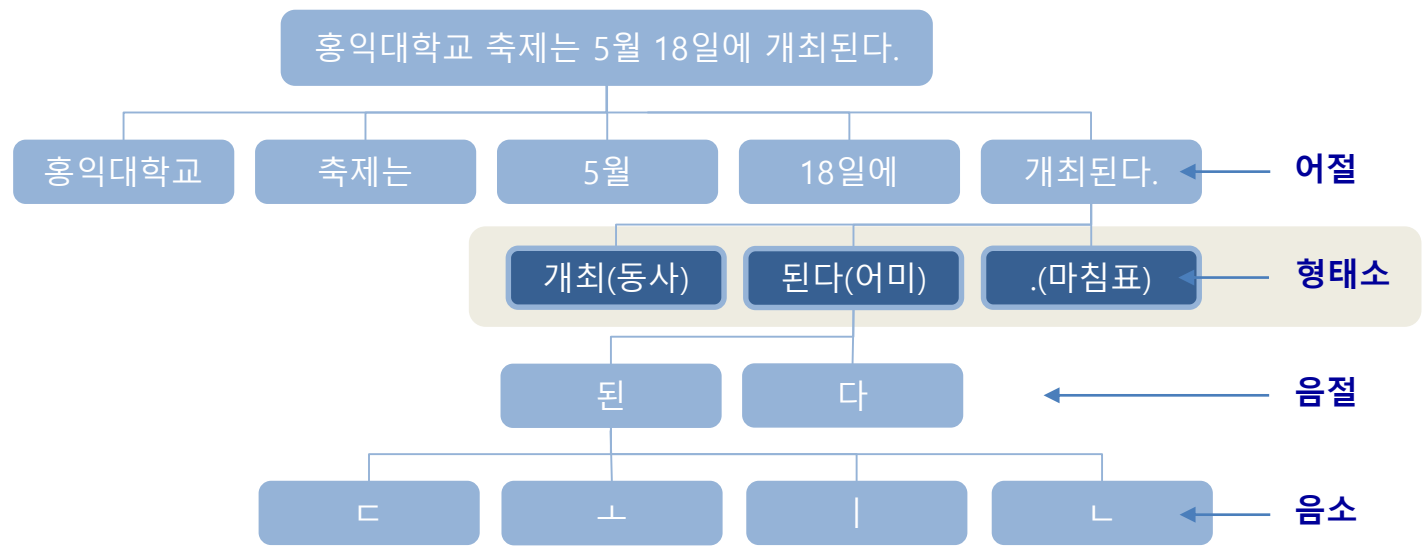
- > 자동으로 시스템에 접속해 데이터를 화면에  
나타낸 후 필요한 자료를 추출하여 가져오는 기술

<감성 분석의 3단계>



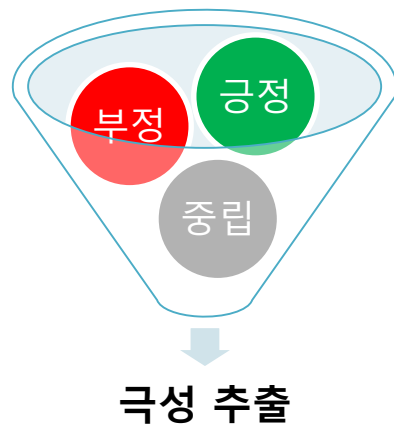
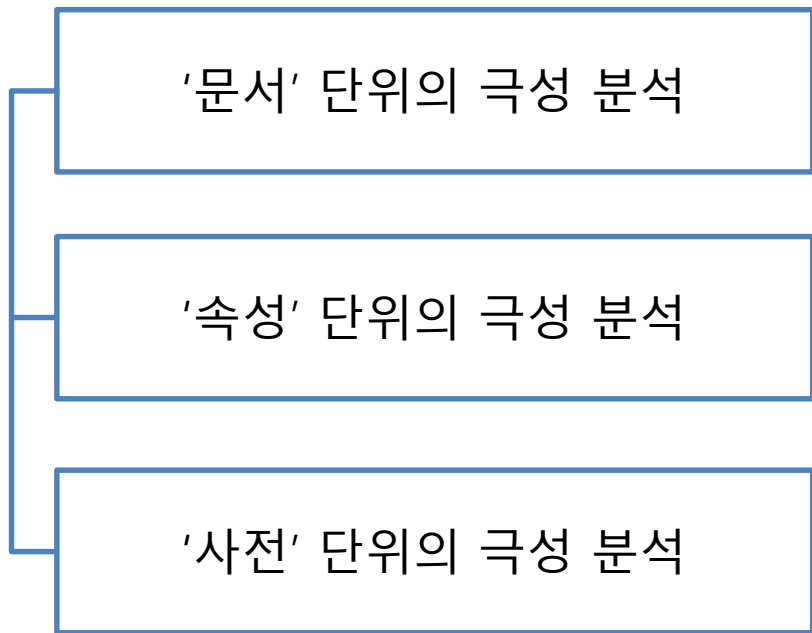
## 2. 형태소 분석

-> 텍스트로부터 작성자의 감정이나 의견을 추출하기 위해선 텍스트를 형태소단위로 분리하여  
각 형태소별 극성을 파악한 후 전체 텍스트의 극성을 분류하는 방식



### 3. 극성 탐지

<극성 분석 방법>



-> 감성분석의 기본 작업은 텍스트의 극성을 긍정, 부정, 중립 등으로 분류하는 것이다.

## 1&2. 데이터 수집 (뉴스 스크래핑&파싱)

-> 수집데이터: 네이버 증권 뉴스 (13년 1월~15년 4월)

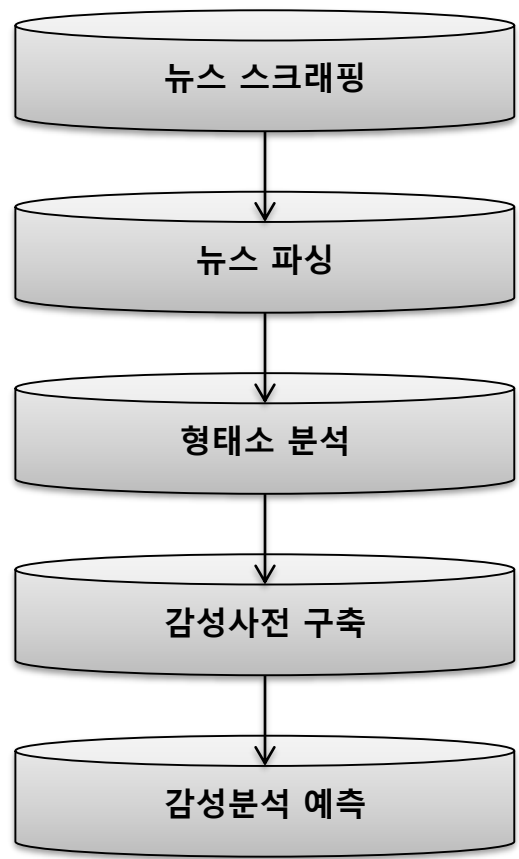
## 3.형태소 분석

-> 수집된 온라인 뉴스에서 '명사' 만을 활용.

-> 빈번하게 발생하는 불필요한 어휘와 의미를 알 수  
없는 단음절 체언 및 용언을 제거

-> 최종적으로 추출된 명사의 감성 점수화를 통해  
기업별 감성사전을 구축

<뉴스 데이터를 활용한 감성분석 단계>





## 4. 감성사전 구축

<각 어휘의 감성점수 계산방법>

금정적 영향을 갖는 뉴스에서  
발생한 i의 출현 빈도

(1)

$$TermScore(i_p) = \frac{Num(i \in PosDocs)}{TotalNum(i)}$$

어휘 i의 순 감성 점수

(3)

$$TermScore(i) = TermScore(i_p) - TermScore(i_n)$$

부정적 영향을 갖는 뉴스에서  
발생한 i의 출현 빈도

(2)

$$TermScore(i_n) = \frac{Num(i \in NegDocs)}{TotalNum(i)}$$

뉴스 전체에 나온 i의 출현빈도

Termscore(i) > 0 : 긍정 속성단어

Termscore(i) < 0 : 부정 속성단어

## 5. 감성분석 예측

<개별 기업의 주가 예측식>

t시점의 기업 j에 대한 오피니언 점수

↓

어휘 i의 극성 점수

↓

$$ComScore(j_t) = \frac{\sum_{i=1}^n Num(i_t) \times TermScore(i)}{\sum_{i=1}^n Num(i_t)}$$

↑

t 시점에 발생한 모든 뉴스에서의 어휘 i의 출현 빈도

-> 뉴스의 COMSCORE 를 구한 뒤

Comscore > 0 : 상승

(뉴스 기간: 전일 거래종료일 ~ 다음 거래일 시작 시간)

Comscore < 0 : 하락

실제 다음 거래일의 주가 등락과 일치하는지 확인

## 6. 기존사례 문제점 분석

확률론 적인 방법이 아닌, 단순 빈도수 의 합

불용어 수준의 단어를 처리하지 못함



주가 예측이  
평균 56%로  
낮은 정확도 관측

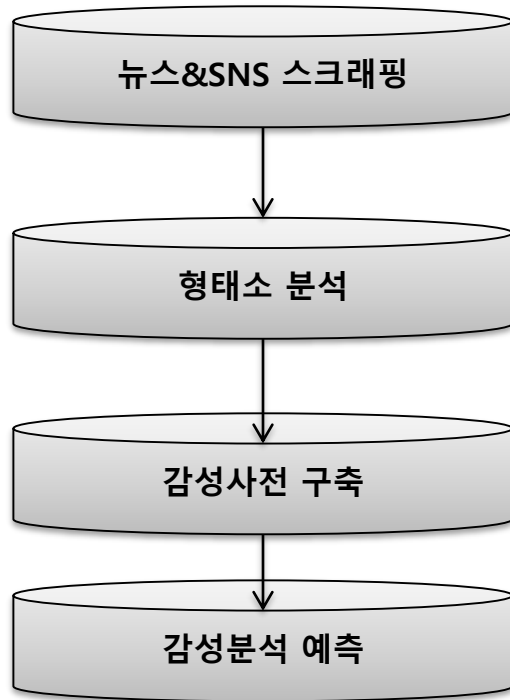
## 1. 데이터 수집 (뉴스 스크래핑&파싱)

-> 수집데이터: 다음 증권 뉴스 (13년 1월~13년 12월)  
SNS (Twitter) 추출

## 2. 형태소 분석

-> 수집된 온라인 뉴스에서 '명사' 만을 활용.

-> 최종적으로 추출된 명사의 감성 점수화를 통해  
기업(7종목)감성사전을 구축



### 3. 감성사전 구축

#### (1) 단어의 P(i) <긍정지수> 구하기

$$word(i, j) = \begin{cases} 1 & \text{\{기사 } j\text{에 단어 } i\text{가 포함된 경우}\}} \\ 0 & \text{(그 외의 경우)} \end{cases}$$

$$NSP(j) = \begin{cases} 1 & \begin{cases} \text{기사 } j\text{가 게재된 후 익일} \\ \text{주가가 상승한 경우} \end{cases} \\ 0 & \text{(그 외의 경우)} \end{cases}$$

↑  
익일 추가

$$positive(i) = \sum_{j=1}^n \{word(i, j) \times NSP(j)\}$$

↑  
Positive(i)는 긍정 값

감성사전은 '단어'와 '긍정' 2가지를 주축으로 구축

### 3. 감성사전 구축

#### (1) 단어의 P(i) <긍정지수> 구하기

$$frequency(i) = \sum_{j=1}^n word(i, j)$$



학습된 뉴스에서 출현 횟수의 합

$$P(i) = \frac{\sum_{j=1}^n \{word(i, j) \times NSP(j)\}}{frequency(i)}$$



긍정지수는 긍정 값을 빈도수로 나눔

### 4. 감성분석 예측

#### (1) 텍스트의 PT(i) <긍정지수> 구하기

$$match(i, j) = \begin{cases} 1 & \left\{ \begin{array}{l} \text{텍스트 } i\text{에 포함된 명사 } j\text{가} \\ \text{감성사전에 존재 할 경우} \end{array} \right. \\ 0 & (\text{그 외의 경우}) \end{cases}$$

$$PT(i) = \frac{\sum_{j=1}^n \{match(i, j) \times P(j)\}}{\sum_{j=1}^n match(i, j)}$$




같은 개념으로 텍스트의 긍정지수 계산

## 4. 감성분석 예측

### (2) 일별 긍정 지수 구하기

일별 긍정 지수


$$DP(i) = \frac{\sum_{j=1}^n PT(j)}{n}$$

$n = \text{number of text in } i$

$DP(i) > 0.5 \Rightarrow \text{상승}$

$DP(i) < 0.5 \Rightarrow \text{하락}$

사례1과 차이점

- 상승할 때, 기사만을 고려해 상승 예측에 주력
- 반복되는 단어의 횟수를 모두 1로 처리

<사례1> 과 <사례2>의 비교

공통점	차이점
<div>1. 감성사전 구축</div> <div>2. 매일 opinion 스코어로 주가 등락예측</div>	<div>1. &lt;사례1&gt;은 기사 하나하나에서 중복 횟수를 counting 해주었으나, &lt;사례2&gt;는 단일기사에 여러번 등장해도 1로 처리</div> <div>2. &lt;사례1&gt;은 긍정, 부정 기사 둘 다 고려, &lt;사례2&gt;는 긍정 속성 기사만 고려해 좀 더 상승에 집중</div>



## 1&2. 데이터 수집(뉴스 스크래핑 & 파싱)

-> 수집데이터 : 1999/11/14 and 2000/02/11 (나스닥 시가총액상위 12위)

## 3. 형태소 분석

-> 명시되지 않음

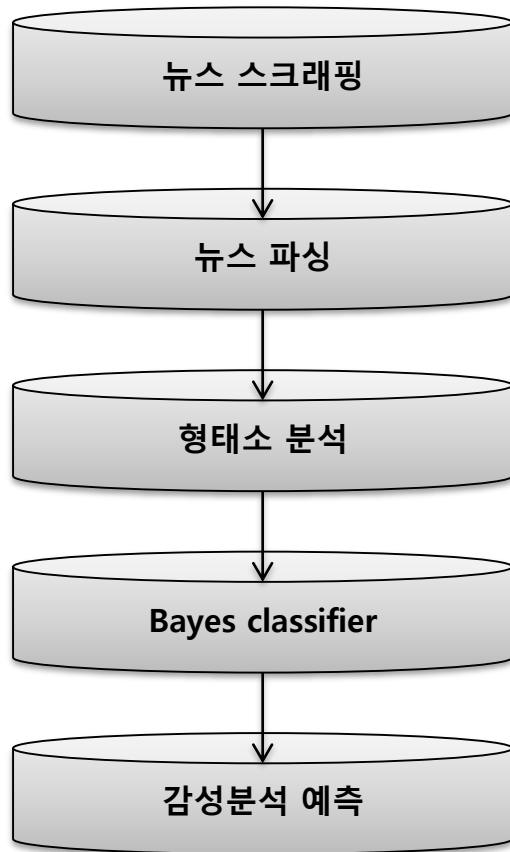
## 4. Bayes classifier

-> 다음 슬라이드부터 자세히 설명

## 5. 감성분석 예측

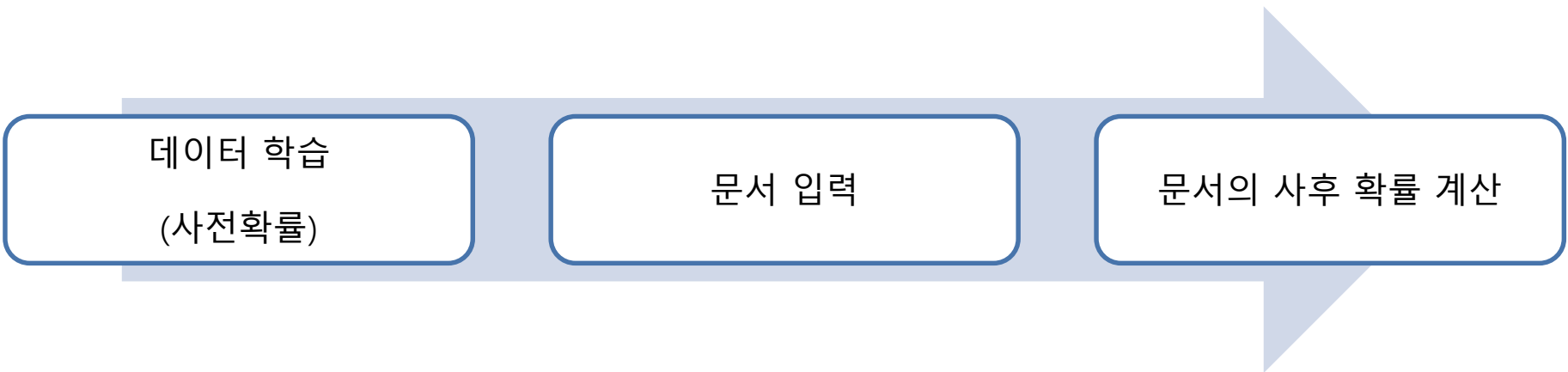
->  $P = P(\text{positive} | \text{document})$  개수 / number of document

$P > 0.5$  다음날 상승    $P < 0.5$  다음날 하락



## 4.1 Naive Bayesian Classification [10]

- > 이 분류는 분류를 위해서 베이즈 룰(Bayes' Rule)을 기본적으로 사용한다.
- > 분류에 필요한 파라미터를 추정하기 위한 트레이닝 데이터의 양이 적어도 사용 가능하다.
- > 많은 복잡한 실제 상황에서 잘 작동한다.



데이터 학습  
(사전확률)

문서 입력

문서의 사후 확률 계산

### 4.1.1 베이즈 정리

- 두 확률 변수의 사전확률과 사후확률 사이의 관계를 나타내는 정리
- 어떤 사건 B가 일어났을 때 사건 A가 일어날 확률 (사후확률)

즉,  $P(A)$ ,  $P(B)$ ,  $P(B|A)$  를 알면  $P(A|B)$ 를 구할 수 있다.

$$P(A|B) = \frac{P(A \cap B)}{P(B)} \quad \quad \quad P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

\*  $P(A)$  = A의 사전확률,  $P(B)$  = B의 사전확률,

$P(B|A)$  = A가 주어졌을 때 B의 조건부확률,  $P(A|B)$  = 사후확률

## 4.1.2 Naive Bayesian Classification

$$P(c|d) = \frac{P(d|c)p(c)}{P(d)}$$

d : 입력 문서

c : 분류할 부류(Class)로 긍정, 부정으로 나뉨

If  $P(\text{긍정}|\text{문서}) > P(\text{부정}|\text{문서})$ , 문서가 긍정부류에 속함

If  $P(\text{긍정}|\text{문서}) < P(\text{부정}|\text{문서})$ , 문서가 부정부류에 속함

$$P(\text{긍정}|\text{문서}) = \frac{P(\text{문서}|\text{긍정})P(\text{긍정})}{P(\text{문서})} \quad P(\text{부정}|\text{문서}) = \frac{P(\text{문서}|\text{부정})P(\text{부정})}{P(\text{문서})}$$

동일하므로 긍정, 부정  
대소비교에 지장없음

## 4.1.2 Naive Bayesian Classification

$w$  = vector of words =  $(w_1, w_2, \dots, w_n)$

문서에 속한 단어들의 벡터(모음)  
ex) 긍정 or 부정에 상당한 영향을 줄 수 있는 단어벡터

$$\begin{aligned} P(\text{문서}|\text{부정}) &= P(w|\text{부정}) \\ &= P(w_1, w_2, \dots, w_n|\text{부정}) \end{aligned}$$

각각의 단어들이 서로 독립이므로 분리가 가능하다  
(베이즈 정리의 기본가정)

$$\begin{aligned} P(w|\text{부정}) &= P(w_1|\text{부정}) P(w_2|\text{부정}) \dots P(w_n|\text{부정}) \\ &= \prod_{i=1}^n P(w_i|\text{부정}) \end{aligned}$$

$$\therefore P(\text{부정}|\text{문서}) \propto P(\text{문서}|\text{부정})P(\text{부정})$$

$$\begin{aligned} &\propto P(w|\text{부정}) P(\text{부정}) \\ &\propto \prod_{i=1}^n P(w_i|\text{부정}) P(\text{부정}) \end{aligned}$$

$$\rightarrow P(w_1|\text{부정}) P(w_2|\text{부정}) \dots P(w_n|\text{부정})P(\text{부정})$$

## 4.1.2 Naive Bayesian Classification

### # 문제점

만약 학습 문서에 없던 새로운 단어가 나왔을 때 확률이 0이 되는 문제가 발생

### # Laplace Smoothing

학습문서에 없던 새로운 단어가 나오더라도 해당 빈도에 +1을 해줌으로써 확률이 0이 되는 것을 방지

$$\hat{P}(x|c) = \frac{\text{count}(x,c)+1}{\sum_{x \in V} (\text{count}(x,c)+1)} = \frac{\text{count}(x,c)+1}{(\sum_{x \in V} \text{count}(x,c)) + |V|}$$

↑  
|V|는 전체 단어의 수가 아니라  
유일한 단어의 개수

### 4.1.3 Naive Bayesian Classification

- 영화 평점을 추론하는 베이지안 분류기 예시 [8]
- 단어 별 사전확률을 구하기위해 데이터를 입력해서 학습시킴

D1. (7/10 점) ---->POS (뜻 : 영화의 평이 7점이고 5점 이상이니 영화에 긍정적인 평이다.)

" The Lobster " which is a **great** movie ever seen, is more **better** than a satire on the dating game. It digs **better**, **great** at the status of our most tender emotions.  
The sence has the **greatest** scence and **good** acting.

D3. (3/10 점) ---->NEG (뜻 : 영화의 평이 3점이고 5점 이하 이니 영화에 부정적인 평이다.)

"The Lobster" is a **good** piece of satire, but largely fails in an poor attempt to build its **poor** wit into a more conventional romance. And general flat affect, mirrored in **poor** visuals and **poor** performance, eventually stopped being funny.

입력된 위의 예시 데이터 2 문서의  
형용사 부분 추출 해서 표로 정리

	"good"	"great"	"poor"	"class"
d1.	3	3	0	pos
d3.	1	0	3	neg

### 4.1.3 Naive Bayesian Classification

	"good"	"great"	"poor"	"class"
d1.	3	3	0	pos
d2.	0	2	1	pos
d3.	1	0	3	neg
d4.	1	2	5	neg
d5.	0	0	2	neg

-> 현재 "class" 는 {positive, negative} 2개가 있으며 학습 문서는 5개가 있다. 앞 예시 데이터 2개와 추가로 3개 더 입력

영화 평점 분류기 입력 예시 데이터

Review: A good, good plot and great characters, but poor acting

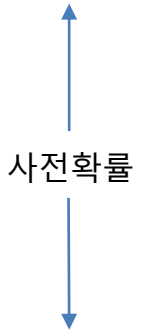
이 리뷰가 입력 됐을 때, 이 리뷰의 극성을 탐지해보자.

### # 나이브 베이즈 분류기 적용

$P(\text{pos}|\text{doc}) \leftarrow$  문서가 긍정일 확률(사후확률)  
=  $P(\text{doc}|\text{pos}) P(\text{pos})$   
=  $P(\text{good}|\text{pos}) P(\text{good}|\text{pos})P(\text{great}|\text{pos})P(\text{poor}|\text{pos})P(\text{pos})$   
$$= \frac{3+1}{9+3} \cdot \frac{3+1}{9+3} \cdot \frac{5+1}{9+3} \cdot \frac{1+1}{9+3} \cdot \frac{2}{5}$$
$$= \frac{4}{12} \cdot \frac{4}{12} \cdot \frac{6}{12} \cdot \frac{2}{12} \cdot \frac{2}{5}$$
$$= 0.0037$$

$P(\text{neg}|\text{doc}) \leftarrow$  문서가 부정일 확률(사후확률)  
=  $P(\text{doc}|\text{neg}) P(\text{neg})$   
=  $P(\text{good}|\text{neg})P(\text{good}|\text{neg})P(\text{great}|\text{neg}) P(\text{poor}|\text{neg})P(\text{neg})$   
$$= \frac{2+1}{14+3} \cdot \frac{2+1}{14+3} \cdot \frac{2+1}{14+3} \cdot \frac{10+1}{14+3} \cdot \frac{3}{5}$$
$$= \frac{3}{17} \cdot \frac{3}{17} \cdot \frac{3}{17} \cdot \frac{11}{17} \cdot \frac{3}{5}$$
$$= 0.0021$$

$P(\text{neg}|\text{d}) < P(\text{pos}|\text{d})$  이므로 해당 리뷰는 positive 부류로 분류된다.





## 5. 기존사례 문제점 분석

- > 단어별로 문서의 감성에 영향을 끼치는 정도가 다를 수 있다.
- > 베이지안 분류기 학습 시 많은 데이터 양 필요
- > 단어별 독립을 가정.

02. 기존사례 비교분석표

논문	수집범위	예측범위	추출단위	분류알고리즘	텍스트처리방법	주식시장	소스	언어
사례1	13년1월~15년4월 상위100개	다음날 증가	명사	감성사전	KoNLPy	KOSPI	네이버 뉴스	파이썬
사례2	13년1월~12월 (7개 종목)	다음날 증가	명사	감성사전	꼬꼬마 형태소 분석기	KOSPI	SNS, 다음뉴스	자바
사례3	99년11월~00년2월	다음날 증가	명사	Bayesian Classifier	명시 안됨	NASDAQ	TIMES, CNN	명시 안됨
수행 과제	14년6월~15년6월 상위30개	당일 증가	명사	Bayesian Classifier & 감성사전구축	KoNLPy	KOSPI	네이버 금융뉴스	파이썬

## 기존 방식 문제점

### 감성사전 이용방식 문제점

- > 여러 기사에서 단순히 많이 등장 하는(ex: 기대,고려,생각..) 불용어가 높은 점수를 가질 수 있다.
- > 뉴스에서 단어끼리 서로 의미와 등장에 영향을 주기에 베이지안 처럼 곱하기 연산을 해야 하나 더하기 연산으로 서로 관계를 가지지 않고 독립적인 존재로 생각함.

### 단순 베이지안 분류기 문제점

- > 감성 분류에서 빈도수보다는 그 해당 단어 자체가 있느냐 없느냐가 더 중요 할 수 도 있다.
- 즉, 단어의 가중치를 생각하지 못함.

### 감성사전 이용방식 문제점 해결<sup>[11]</sup>

Com score 조금 변형한 TF-IDF 이용.

#### 1) TF

한 문서 내에서 등장하는 단어의 빈도를 나타내는데 단어와 문서 간의 중요도를 나타내기 위한 것이다.

문서 내에서 많이 출현 할수록(TF가 높을수록) 상대적으로 더 중요하다는 의미이다.

$$TF_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}}$$

*n<sub>i,j</sub>* ← 문서 *d<sub>j</sub>*에서 단어 *t<sub>i</sub>*가 나오는 횟수  
*n<sub>k,j</sub>* ← 문서 *d<sub>j</sub>*에서 나오는 모든 단어횟수

문서 *d<sub>j</sub>*에서 단어 *t<sub>i</sub>*의 중요도.

#### 2) IDF

- DF란?

DF(Document frequency)는 문서 빈도, 자주 등장하는 단어가 몇 개의 문서에 등장 하는지를 나타낸다.

(DF가 높다? 전체문서에서 많은 횟수로 등장하는 단어로, 불용어 수준이라 생각할 수 있다.)

- IDF란?

IDF는 DF의 역수이며 로그를 취해 준다. 단어 간의 거리를 일정하게 유지하기 위해서 로그를 취해 주는데 자연로그나 상용로그 중 선택 하면 된다.

$$IDF(t,D) = \log \left( \frac{\text{전체 문서의 갯수}}{\text{단어 } t \text{가 포함된 문서의 수}} \right)$$

## TF-IDF 요약

$$TF = \frac{\text{문서 내 단어의 개수}}{\text{문서 내 모든 단어의 수}}$$

$$IDF = \log\left(\frac{\text{문서 전체 갯수}}{\text{단어를 포함한 문서의 수}}\right)$$

$$TF-IDF(t,d,D) = TF(t,d) \times IDF(t,D)$$

-> 특정 문서 내에서 단어 빈도가 높고(TF 높고), 전체 문서에서 그 단어가 포함된 문서가 적다면(IDF 높으면) 이 값은 높아진다.

-> 따라서 이 값을 이용하면 불용어를 걸러 낼 수 있으며 단어별 가중치가 된다.

예시) Binarized Naive Bayesian Classification

	"good"	"great"	"poor"	"class"
d1.	1	1	0	pos
d2.	0	1	1	pos
d3.	1	0	1	neg
d4.	1	1	1	neg
d5.	0	0	1	neg

-> 현재 "class" 는 {positive, negative} 2개가 있으며  
학습 문서는 5개가 있다.

분류기 입력 예시 데이터

Review: A good, good plot and great characters, but poor acting

# 나이브 베이즈 분류기 적용

$$\begin{aligned} P(\text{pos}|\text{doc}) &= P(\text{doc}|\text{pos}) P(\text{pos}) \\ &= P(\text{good}|\text{pos}) P(\text{good}|\text{pos})P(\text{great}|\text{pos})P(\text{poor}|\text{pos})P(\text{pos}) \\ &= \frac{1+1}{4+3} \cdot \frac{1+1}{4+3} \cdot \frac{2+1}{4+3} \cdot \frac{1+1}{4+3} \cdot \frac{2}{5} \\ &= \frac{2}{7} \cdot \frac{2}{7} \cdot \frac{3}{7} \cdot \frac{2}{7} \cdot \frac{2}{5} \\ &= 0.0040 \end{aligned}$$

$$\begin{aligned} P(\text{neg}|\text{doc}) &= P(\text{doc}|\text{neg}) P(\text{neg}) \\ &= P(\text{good}|\text{neg})P(\text{good}|\text{neg})P(\text{great}|\text{neg}) P(\text{poor}|\text{neg})P(\text{neg}) \\ &= \frac{2+1}{6+3} \cdot \frac{2+1}{6+3} \cdot \frac{1+1}{6+3} \cdot \frac{3+1}{6+3} \cdot \frac{3}{5} \\ &= \frac{3}{9} \cdot \frac{3}{9} \cdot \frac{2}{9} \cdot \frac{4}{9} \cdot \frac{3}{5} \\ &= 0.0066 \end{aligned}$$

$P(\text{neg}|\text{d}) > P(\text{pos}|\text{d})$  이므로  
해당 리뷰는 negative 부류로 분류된다.

예시) Binarized Naive Bayesian Classification

	"good"	"great"	"poor"	"class"
d1.	1	1	0	pos
d2.	0	1	1	pos
d3.	1	0	1	neg
d4.	1	1	1	neg
d5.	0	0	1	neg

-> 감정 분류에서는 빈도수 보다는 그 해당 단어 자체가 있느냐 없느냐가 더 중요할 수 있다.

-> 해당 단어가 출현하기만 하면 1로 간주하고 그 출현 빈도에 앞선 TF-IDF 가중치 값을 곱한다.

긍정 문서

	"good"	"great"	"poor"
D1 TF-IDF	4.1768	2.277762	1.10648
D2 TF-IDF	4.38974	0.29504	2.05
TF-IDF(Avg)	4.28327	1.53633	1.57824

부정 문서

	"good"	"great"	"poor"
D3 TF-IDF	4.69054	0.02145	1.48033
D4 TF-IDF	2.40144	1.43837	2.73898
D5 TF-IDF	4.5889	3.88289	2.81341
TF-IDF(Avg)	3.89363	2.66063	2.77619

예시) Binarized Naive Bayesian Classification  
TF-IDF 가중치 적용

	"good"	"great"	"poor"	"class"
d1.	4.2833	1.5363	0	pos
d2.	0	1.5363	1.5782	pos
d3.	3.8936	0	2.7762	neg
d4.	3.8936	2.6606	2.7762	neg
d5.	0	0	2.7762	neg

-> 현재 "class" 는 {positive, negative} 2개가 있으며

학습 문서는 5개가 있다.

분류기 입력 예시 데이터

Review: A good, good plot and great characters, but poor acting

# 나이브 베이즈 분류기 적용

$$\begin{aligned} P(\text{pos}|\text{doc}) &= P(\text{doc}|\text{pos}) P(\text{pos}) \\ &= P(\text{good}|\text{pos}) P(\text{good}|\text{pos}) P(\text{great}|\text{pos}) P(\text{poor}|\text{pos}) P(\text{pos}) \\ &= \frac{4.2833+1}{8.9341+3} \cdot \frac{4.2833+1}{8.9341+3} \cdot \frac{3.0726+1}{8.9341+3} \cdot \frac{1.5782+1}{8.9341+3} \cdot \frac{2}{5} \\ &= \frac{5.2833}{11.9341} \cdot \frac{5.2833}{11.9341} \cdot \frac{4.0726}{11.9341} \cdot \frac{2.5782}{11.9341} \cdot \frac{2}{5} \\ &= 0.00578 \end{aligned}$$

$$\begin{aligned} P(\text{neg}|\text{doc}) &= P(\text{doc}|\text{neg}) P(\text{neg}) \\ &= P(\text{good}|\text{neg}) P(\text{good}|\text{neg}) P(\text{great}|\text{neg}) P(\text{poor}|\text{neg}) P(\text{neg}) \\ &= \frac{7.7872+1}{18.7764+3} \cdot \frac{7.7872+1}{18.7764+3} \cdot \frac{2.6606+1}{18.7764+3} \cdot \frac{8.3286+1}{18.7764+3} \cdot \frac{3}{5} \\ &= \frac{8.7872}{21.7764} \cdot \frac{8.7872}{21.7764} \cdot \frac{3.6606}{21.7764} \cdot \frac{9.3286}{21.7764} \cdot \frac{3}{5} \\ &= 0.00704 \end{aligned}$$

$P(\text{neg}|\text{d}) > P(\text{pos}|\text{d})$  이므로  
해당 리뷰는 negative 부류로 분류된다.



## 1&2. 데이터 수집(뉴스 스크래핑 & 파싱)

-> 수집데이터 : 2014년 6월 ~ 2015년6월 시가총액 상위 30위

## 3. 형태소 분석

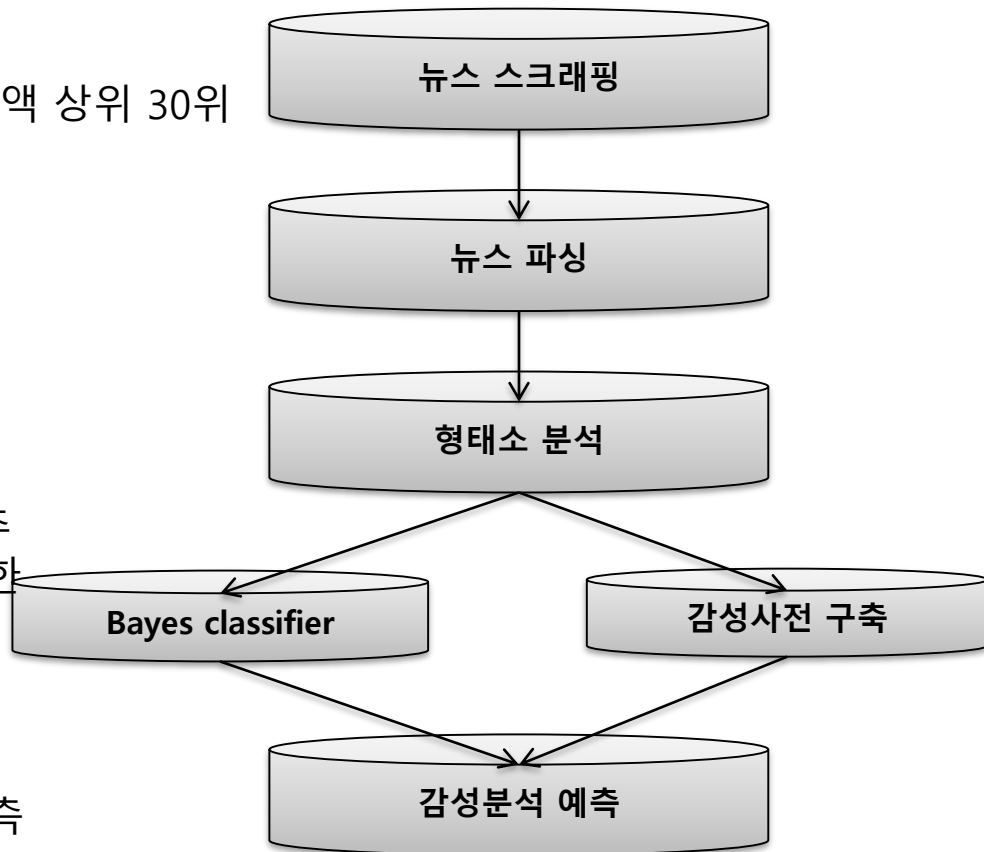
-> KoNLPy(파이썬) 명사만 추출

## 4&5. Bayes classifier & 감성사전 구축

-> Com score 조금 변형한 TF-IDF 이용  
-> 문서에 해당 단어가 출현하기만 하면 1로 간주하고 그 출현 빈도에 앞선 TF-IDF 가중치 값을 곱한다.

## 6. 감성분석 예측

-> 문서의 긍정, 부정확률 비교하여 감성분석 예측



## 검증에 대한 방안

- > 실험항목은 종목에 대한 해당 뉴스가 발행된 시간을 기준으로 주가의 방향성, 정확도, 예측도를 고려하여 다음날 종가(거래 시간 외에 발행된 뉴스라면 다음 개장시의 시초가)가 이전 가격보다 상승 혹은 하락하였는지에 대한 여부를 통계적으로 분석하여 예측하는 실험을 할 계획

# 참고문헌

- [1] <정지선, 김동성, 김종우>, (2015.12), [온라인 언급이 기업 성과에 미치는 영향 분석 : 뉴스 감성분석을 통한 기업별 주가 예측]
- [2] <G .Gidofalvi>, (2003), [Using News Articles to Predict Stock Price Movements], University of San Diego
- [3] <김동영, 박제원, 최재현>, (2014.9), [SNS와 뉴스기사의 감성분석과 기계학습을 이용한 주가예측 모형 비교 연구]
- [4] <김유신, 김남규, 정승렬>, (2012.6), [뉴스와 주가: 빅데이터 감성분석을 통한 지능형 투자의사결정모형]
- [5] <문하늘, 김종우>, (2013), [인터넷 뉴스를 활용한 개별 주식 수익률 예측 모델 연구: 오피니언 마이닝 기법의 활용]
- [6] <감미아, 송민>, (2012.9), 텍스트 마이닝을 활용한 신문사에 따른 내용 및 논조 차이점 분석
- [7] <http://unlimitedpower.tistory.com/entry/NLP-Naive-Bayesian-Classification>
- [8] <http://operatingsystems.tistory.com/entry/Data-Mining-Naive-Bayes-Classification>
- [9] <http://ko.wikipedia.org>
- [10] <Hao Wang\*, Dogan Can\*\*, Abe Kazemzadeh\*\*, François Bar\* and Shrikanth Narayanan\*\* >, (2012), [A System for Real-time Twitter Sentiment Analysis of 2012 U.S. Presidential Election Cycle]
- [11] <이성직, 김한준> Keyword Extraction from News Corpus using Modified TF-IDF

**감 사 합 니 다**