Chapter 10

Representing and Mining Text

Dealing with Text

- Up to this point, we've ignored data preparation
  - The world does not always present us with data in the feature vector representation
  - We must either engineer the data representation to match the tools (representation engineering), or build new tools to match the data
- This chapter will focus on one particular sort of data: text data
- ex) We've encountered text once before in the example involving clustering news stories about Apple Inc.
  - Deliberately avoided how the news stories were prepared because the focus was on clustering
  - In reality, dealing with text requires dedicated preprocessing steps
- This chapter is devoted to the difficulties and opportunities of dealing with text. TF-IDF

Why Text is Important

- Text is everywhere
  - Many legacy applications still produce or record text: medical records, product inquires, repair records, consumer complaint logs, etc.
- The Internet may be the home of ‘new media’, but much of it is the same form as old media
  - Multimedia data may account for a great deal of traffic volume
  - It still contains a vast amount of text: people communicate with each other on the Internet via text
- In business, understanding customer feedback often requires understanding text
  - In many cases, if we want to listen to the customer, we will actually have to read what they have written: product review, feedback form, opinion pieces, and email messages, etc.

Why Text is Difficult

- Text is often referred to as "unstructured " data
  - Either does not have a pre-defined data model or is not organized in a pre-defined manner.

언어학적인, 사람간의 프로토콜

  - Linguistic structure is intended for human communication and not computers
- Sometimes word order matters, sometimes not
- As data, text is relatively dirty
  - People write ungrammatically, misspell words, abbreviate unpredictably, and punctuate randomly

축약

분야마다 다를수도, 알아듣지 못할수도

  - Synonyms, homographs, abbreviations, etc.

동음이의어

- Context matters
  - It is difficult to evaluate any particular word or phrase without considering the entire context
- For these reasons, text must be preprocessed before data mining

Text Representation

- **Goal**: Take a set of documents –each of which is a relatively free-form sequence of words– and turn it into our familiar feature vector form
- *Each document is one instance*

  document data를 바꿔야 알수 있음
  - *but we don’t know in advance what the features will be*
- Terms
  - *A document is composed of individual tokens or terms*

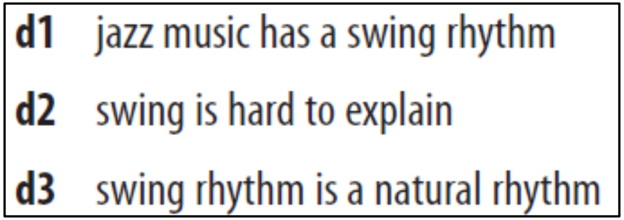
    단어
  - A collection of documents is called a *corpus*

Bag of Words

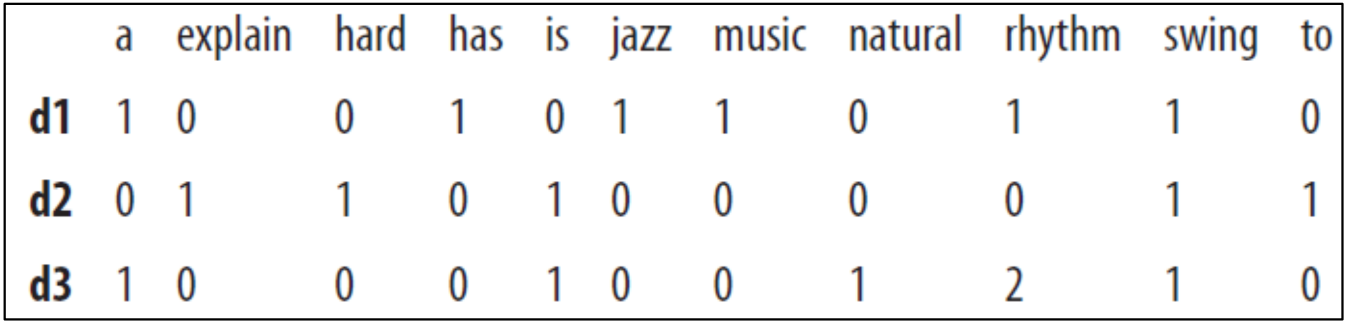
- Treat every document as just a collection of individual words
  - Ignore grammar, word order, sentence structure, and (usually) punctuation
  - Treat every word in a document as a potentially important keyword of the document
- If every word is a feature, what will be the feature's value in a given document? word 자체가 feature이므로 → 0/1
  - Each document is represented by a one (if the token is present in the document) or a zero (the token is not present in the document)
- Straightforward representation
- Inexpensive to generate
- Tends to work well for many tasks

Term Frequency (TF)

- Use the word count (frequency) in the document instead of just a zero or one
  - Differentiates between how many times a word is used



d1	jazz music has a swing rhythm
d2	swing is hard to explain
d3	swing rhythm is a natural rhythm



	a	explain	hard	has	is	jazz	music	natural	rhythm	swing	to
d1	1	0	0	1	0	1	1	0	1	1	0
d2	0	1	1	0	1	0	0	0	0	1	1
d3	1	0	0	0	1	0	0	1	2	1	0

Term Frequency (TF)

- Pre-processing of text

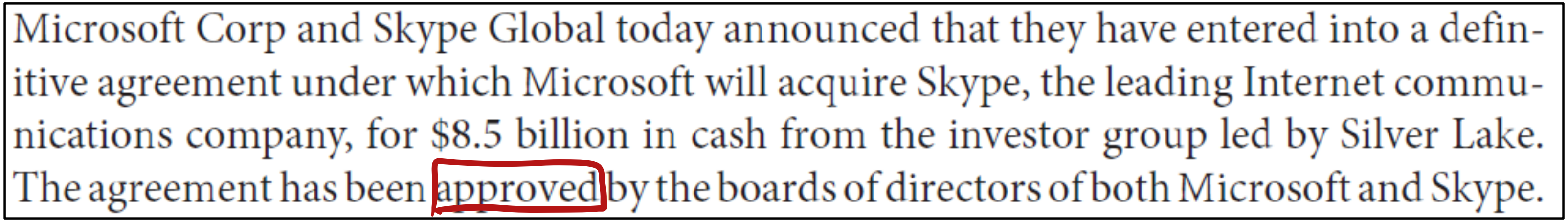
1. The case should be normalized
   - Every term is in lowercase
   - E.g.,) iPhone, iphone, IPHONE → iphone: counted as the same thing
2. Words should be stemmed 줄기. 자잘한거 제거
   - Suffixes are removed: noun plurals, tense in verbs 단/복수, 시제
   - E.g.,) directors → director, announces/announcing/announced→announc ✓

     es,s / ed,d / ... 하나로통일.
3. Stop words should be removed
   - Stop words are words which are filtered out before or after processing of natural language data. A stop-word is a very common in languages
   - Typical stop words in English are *the, a, and, of, on, etc.*

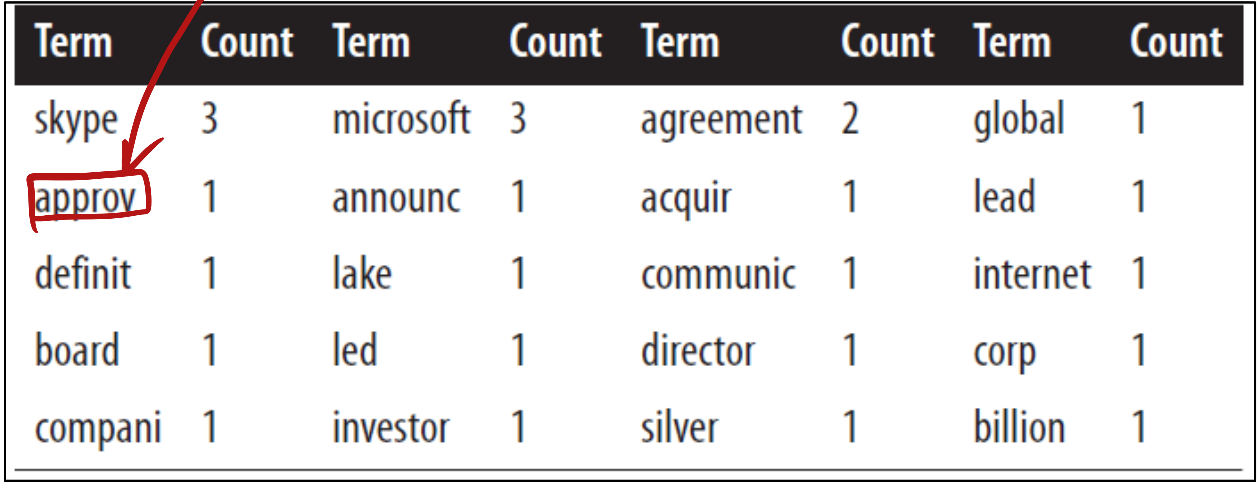
     정관사. 전치사 (not meaningful)

Term Frequency (TF)

- Example: more complex sample document



Microsoft Corp and Skype Global today announced that they have entered into a definitive agreement under which Microsoft will acquire Skype, the leading Internet communications company, for $8.5 billion in cash from the investor group led by Silver Lake. The agreement has been approved by the boards of directors of both Microsoft and Skype.



Term	Count	Term	Count	Term	Count	Term	Count
skype	3	microsoft	3	agreement	2	global	1
approv	1	announc	1	acquir	1	lead	1
definit	1	lake	1	communic	1	internet	1
board	1	led	1	director	1	corp	1
compani	1	investor	1	silver	1	billion	1

Terms after normalization and stemming, ordered by frequency

Normalized Term Frequency

- Documents of various lengths
  - Long documents usually will have more words than shorter ones
  - This does not mean that the longer document is necessarily more important or relevant than the shorter one

- The raw term frequencies are normalized in some way,
  - such as by dividing each by the total number of words in the document
  - or the frequency of the specific term in the corpus

document length

Inverse Document Frequency (IDF)

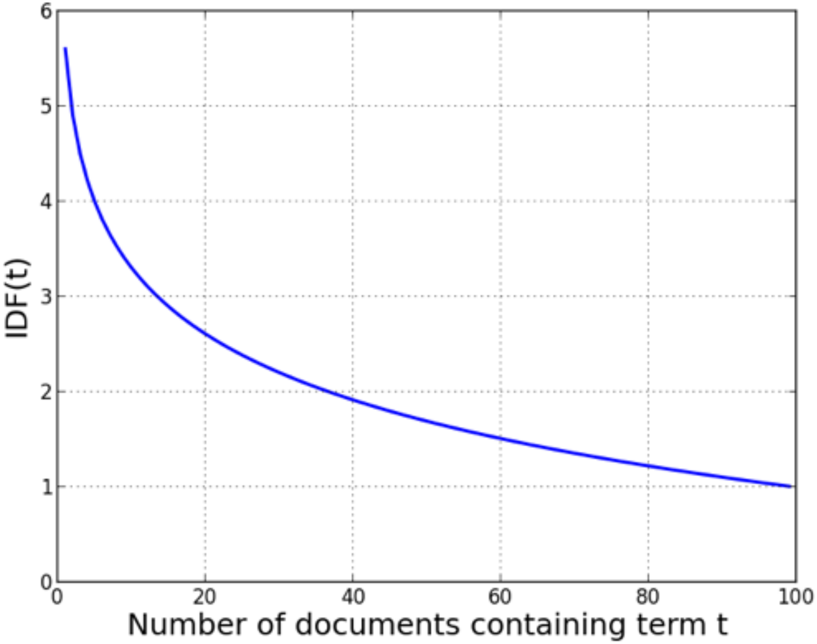
- Term frequency measures how prevalent a term is in a single document
  - We may also care how common it is in the entire corpus we are mining when deciding the weight of a term
- Two opposing considerations
  - A term should not be too rare or too common in your corpus
    - Not useful for clustering or classification
  - Both impose lower or upper limits on term frequency
- Distribution of the term over a corpus
  - The fewer documents in which a term occurs, the more significant it likely is to be to the document → Inverse (작을수록 중요도↑)
  - This sparseness of a term *t* is measured by Inverse Document Frequency (IDF) 희박

Inverse Document Frequency (IDF)

$$\mathrm{IDF}(t) = 1 + \log\left(\frac{\text{Total number of documents}}{\text{Number of documents containing } t}\right)$$

↳ t를 포함한 document 수가 작을수록 log값이 커진다

- IDF may be thought of as the boost a term gets for being rare



← 1에 수렴

Combining TF and IDF: TF-IDF

- TF-IDF: a very popular representation for text
  - The product of TF and IDF

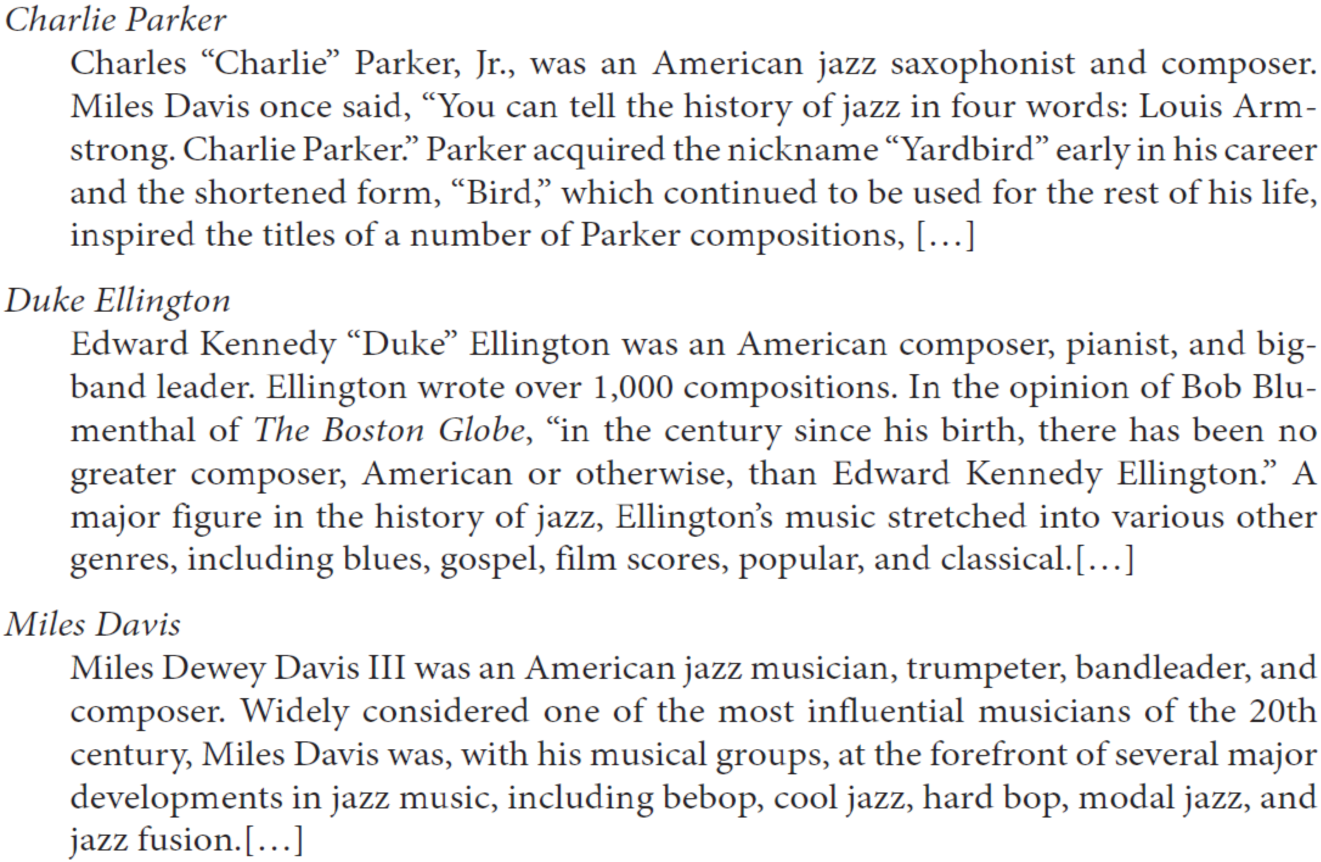
전체 corpus에서

$$\boxed{\mathrm{TF{-}IDF}(t,d) = \mathrm{TF}(t,d) \times \mathrm{IDF}(t)}$$

  - Note: TF-IDF value is specific to a single document (d), whereas IDF depends on the entire corpus.
  - Each document becomes a feature vector
  - The corpus is the set of these feature vectors
- Feature selection is often employed 너무 많아서
  - There are many potential terms with text representation
  - Various ways: imposing min/max thresholds of term counts, measure information gain to rank the terms by importance

Example: Jazz Musicians

- 15 prominent jazz musicians and excerpts of their biographies from Wikipedia



Charlie Parker

Charles “Charlie” Parker, Jr., was an American jazz saxophonist and composer. Miles Davis once said, “You can tell the history of jazz in four words: Louis Armstrong. Charlie Parker.” Parker acquired the nickname “Yardbird” early in his career and the shortened form, “Bird,” which continued to be used for the rest of his life, inspired the titles of a number of Parker compositions, [...]

Duke Ellington

Edward Kennedy “Duke” Ellington was an American composer, pianist, and big-band leader. Ellington wrote over 1,000 compositions. In the opinion of Bob Blumenthal of *The Boston Globe*, “in the century since his birth, there has been no greater composer, American or otherwise, than Edward Kennedy Ellington.” A major figure in the history of jazz, Ellington’s music stretched into various other genres, including blues, gospel, film scores, popular, and classical.[...]

Miles Davis

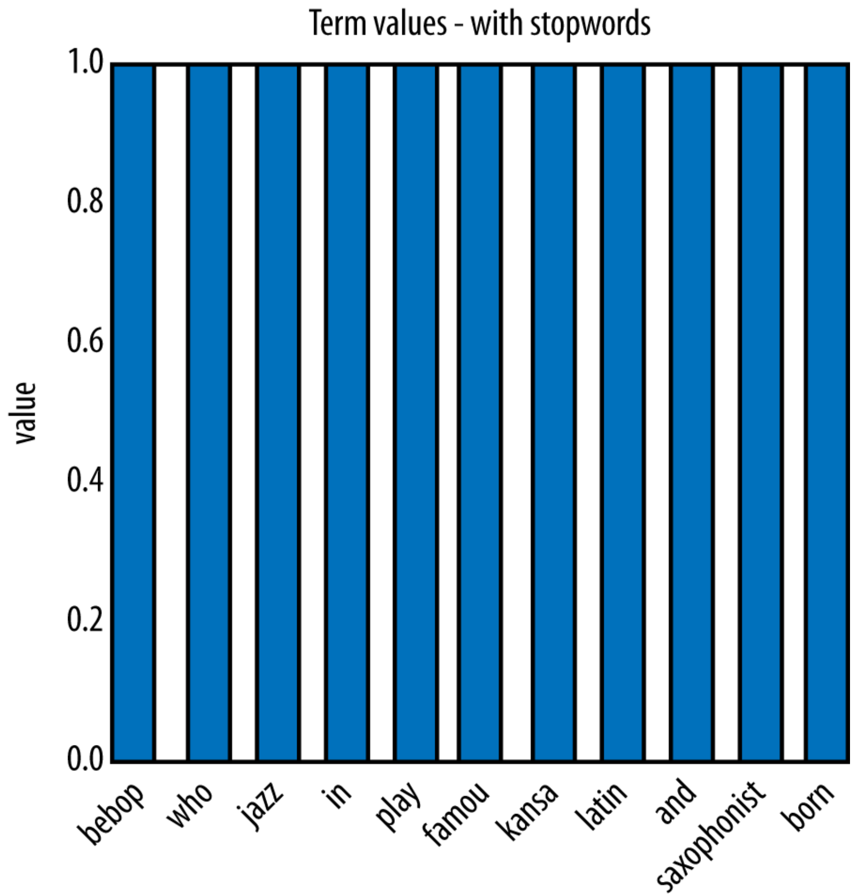
Miles Dewey Davis III was an American jazz musician, trumpeter, bandleader, and composer. Widely considered one of the most influential musicians of the 20th century, Miles Davis was, with his musical groups, at the forefront of several major developments in jazz music, including bebop, cool jazz, hard bop, modal jazz, and jazz fusion.[...]

Example: Jazz Musicians

- Even with this fairly small corpus of 15 documents, the corpus and its vocabulary are too large to show here.
  - Nearly 2,000 features after stemming and stop-word removal
- So, let’s consider the sample phrase “Famous jazz saxophonist born in Kansas who played bebop and latin”
- Frist, basic stemming is applied. 복수형처럼
  - Stemming methods are not perfect: can produce terms like *kansa* and *famou* from “Kansas” and “famous”
  - As long as it is consistent among all documents, stemming perfection usually is not important

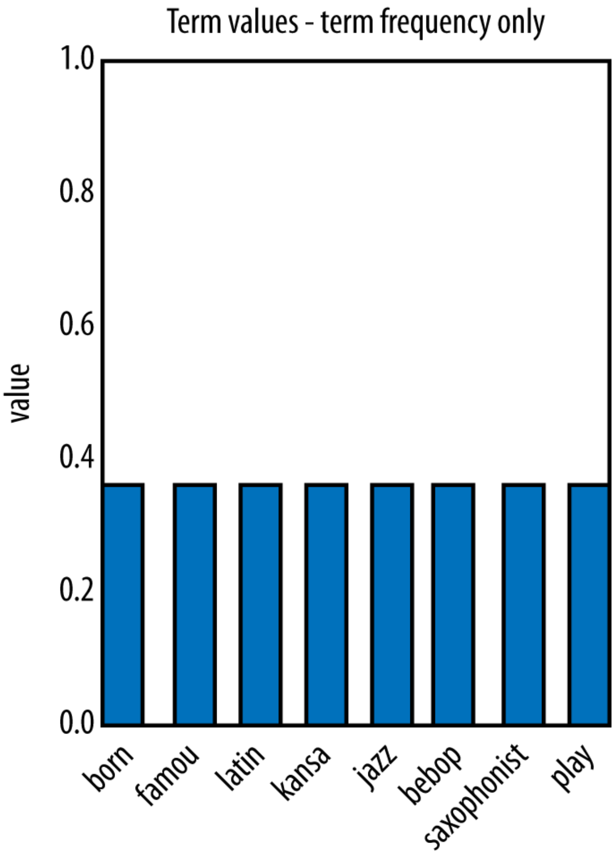
Example: Jazz Musicians

- Stemming



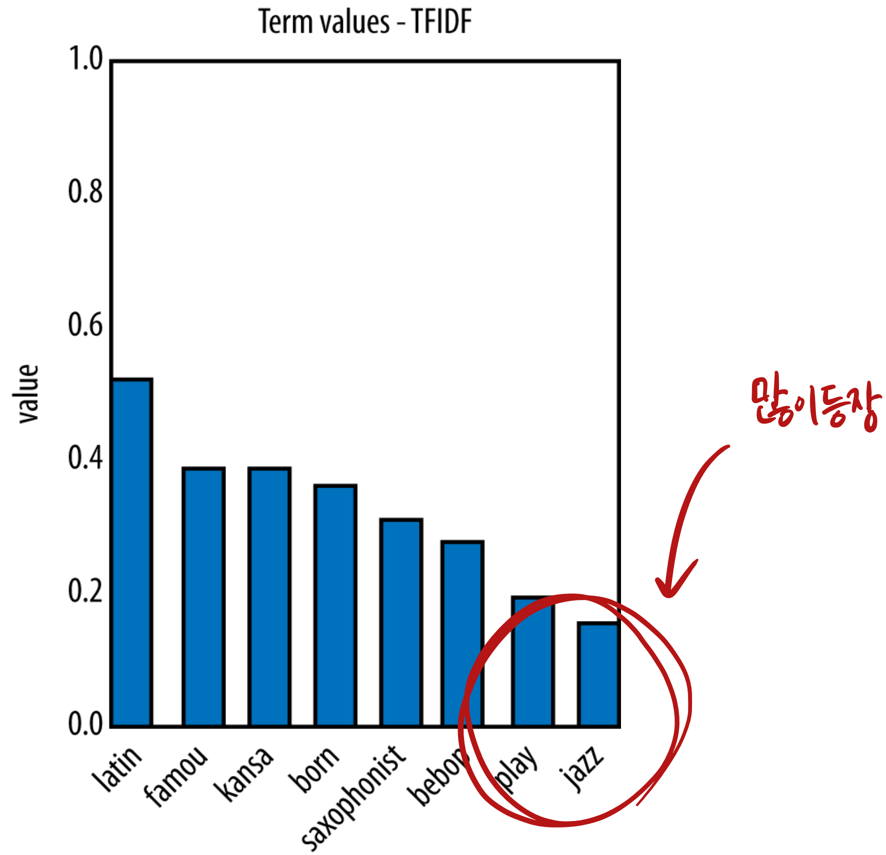
Example: Jazz Musicians

- Stopwords (`in` and `and`) removal and term frequency normalization
  - Normalized with respect to term frequency or document length



Example: Jazz Musicians

- Final TFIDF representation ➔ this is the feature vector of this sample doc
    - Multiplied by each term's TF value by its IDF value



Example: Jazz Musicians

- Let's use this TF-IDF representation for implementing a simple search engine.

  그 구절에 해당하는 feature vector를 만드는것.

- Assume a user typed our sample phrase for a search query: "Famous jazz saxophonist born in Kansas who played bebop and latin." → 여러 웹페이지 (document)와 distance를 구해서 유사도가 높은 순대로 display

- How might it work?
  - First, translate the query to tis TF-IDF representation
  - Second, compute the similarity of our query term to each musician's biography and choose the closest one!
    - We will use cosine similarity because it is often used in text classification to measure the similarity of two documents

가장 많이 사용

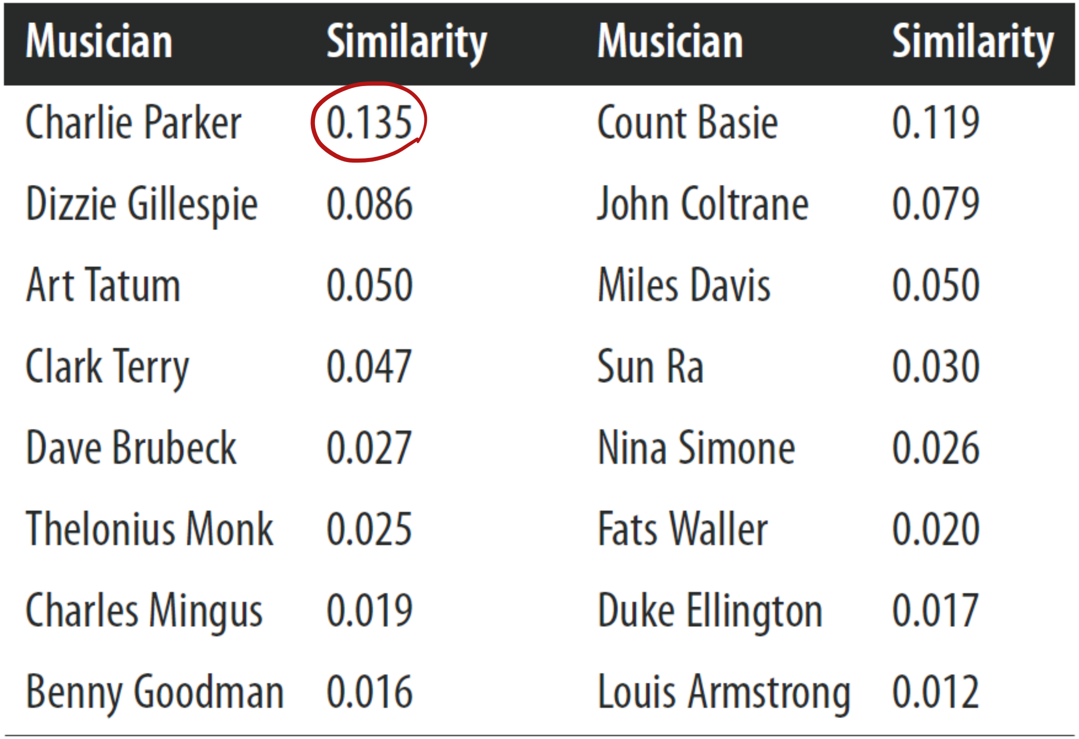
내적

$$\text{Cosine similarity} = \frac{X \circ Y}{\|X\|_2 \ \|Y\|_2}$$

norm (길이)

Example: Jazz Musicians

- AS you can see, the closest matching document is Charlie Parker
  - Who was, in fact, a saxophonist born in Kansas and played the bebop style of jazz



Musician	Similarity	Musician	Similarity
Charlie Parker	0.135	Count Basie	0.119
Dizzie Gillespie	0.086	John Coltrane	0.079
Art Tatum	0.050	Miles Davis	0.050
Clark Terry	0.047	Sun Ra	0.030
Dave Brubeck	0.027	Nina Simone	0.026
Thelonius Monk	0.025	Fats Waller	0.020
Charles Mingus	0.019	Duke Ellington	0.017
Benny Goodman	0.016	Louis Armstrong	0.012

Beyond “Bag of Words”

- The basic bag of words approach is relatively simple but performs surprisingly well on a variety of tasks
  - Usually the first choice of data scientists for a new text mining problem

- Still, there are applications for which more sophisticated techniques must be brought to bear

ex) 단어의 순서

- N-gram Sequences
- Named Entity Extraction 개체명추출
- Topic Models

N-gram Sequences

- Bag of words representation treats every words as a term, discarding word order entirely
- In some cases, **word order is important** and you want to preserve some information about it in the representation
  - A next step up in complexity is to include sequences of adjacent words as terms
- Example: “The quick brown fox jumps”
  - It would be transformed into the set of its constituent words {*quick, brown, fox, jumps*} plus the tokens *quick_brown, brown_fox,* and *fox_jumps*

pair형태의 sequence정보 (복잡도↑)

  - This general representation tactic is called ***n-grams***
  - Adjacent pairs are commonly called *bi-grams*

N-gram Sequences

- “Bag of n-grams up to three” :
  - Simply means it represents each document using as features its individual words, adjacent word pairs, and adjacent word triples

    1개 2개 3개
- Advantage
  - Easy to generate: they requires no linguistic knowledge or complex parsing algorithm 해부.분석, ..

    파싱
- Disadvantages
  - Greatly increase the size of the feature set: far more word pairs than individual words

Named Entity Extraction

개체명추출

document에서의 word, term

- We want still more sophistication in phrase extraction
  - We want to be able to recognize common named entities in document
  - E.g.,) Silicon Valley, Minnesota Twins, The Lord of the Rings, etc.
  - Their component words mean one thing (may not be significant), but in sequence they name unique entities with interesting identities
  - The basic bag of words or even n-grams representation may not capture this because they are based on segmenting text on white space and punctuation

- Named entity extractors are knowledge intensive
  - They have to be trained on a large corpus, or coded by hand

단순한작업X

"name"이기 때문에 기계적 분류X

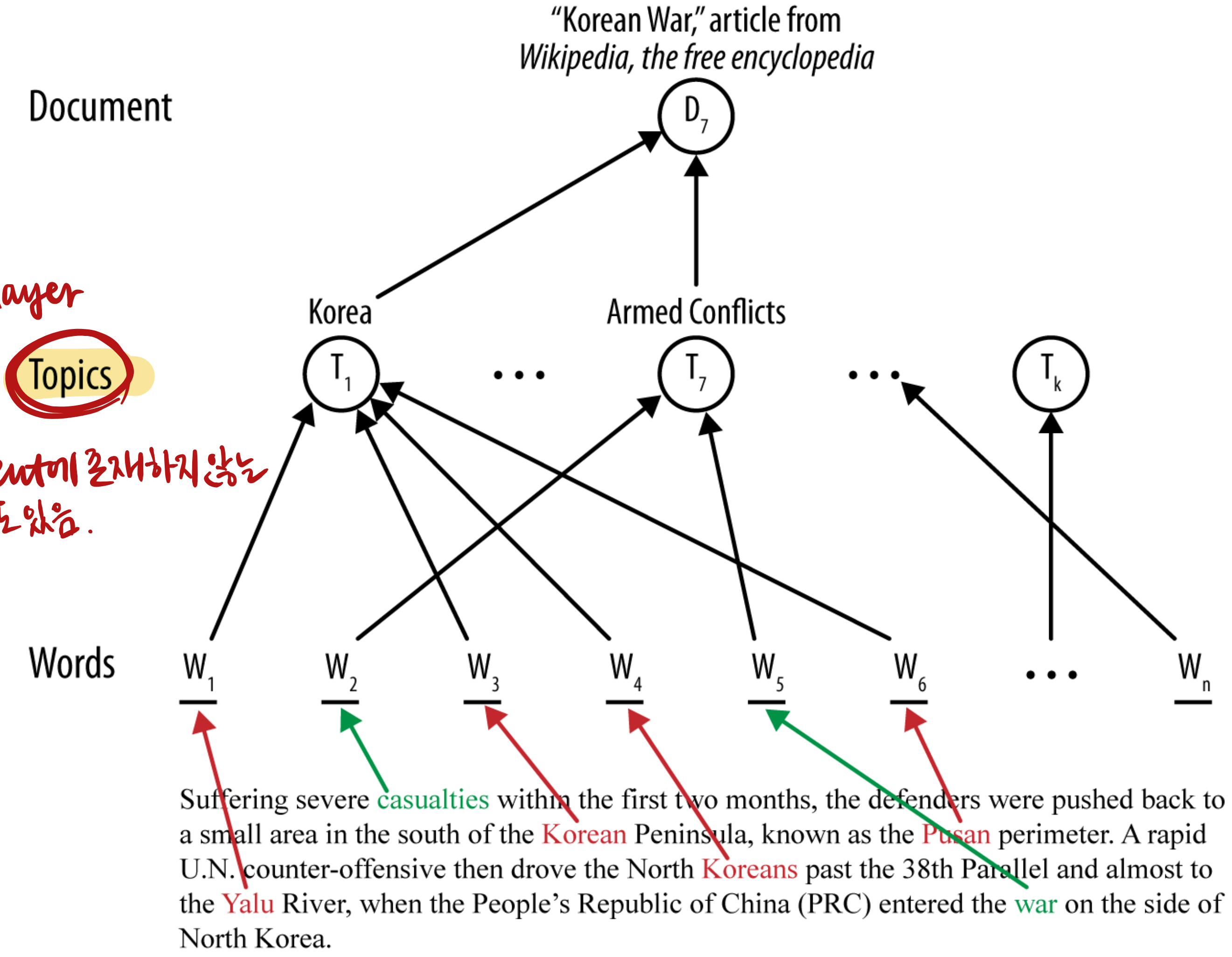
① 사람이따로

② dictionary 작업

Topic Models

- So far we've dealt with models created directly from words appearing from a document
  - Relatively efficient, but not always optimal
- Due to the complexity of language and documents, we sometimes want an additional layer (called topic layer) between document and the model
- Main idea: model the set of topics in a corpus separately
  - Each document constitutes a sequence of words and the words map to one or more topics
  - The topics also are learned from the data
  - We can think of the topic layer as being a clustering of words
- Advantage
  - In a search engine, a query can use terms that do not exactly match the specific words of a document; if they map to the correct topics, the document will still be considered relevant to the search

Topic Models



Suffering severe casualties within the first two months, the defenders were pushed back to a small area in the south of the Korean Peninsula, known as the Pusan perimeter. A rapid U.N. counter-offensive then drove the North Koreans past the 38th Parallel and almost to the Yalu River, when the People's Republic of China (PRC) entered the war on the side of North Korea.

새로운 layer

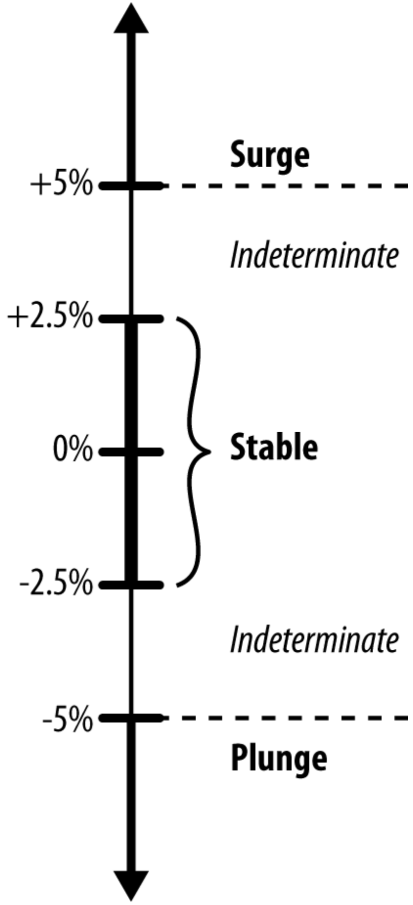
document에 존재하지 않는 단어일수도 있음.

Example: Mining News Stories to Predict Stock Price Movement

- **The Task**: predict the stock market based on the stories that appear on the news wires 주식예측
- Some of the problems and simplifying assumptions
  - – 1. It is difficult to predict the effect of news far in advance. Therefore, we'll try to predict what effect a news story will have on stock price the *same day*.
  - – 2. It is difficult to predict exactly what the stock price will be. Instead, we will be satisfied with the *direction* of movement: **change** and **no change**.
  - – 3. It is difficult to predict small changes in stock price, so instead we'll predict *relatively large* changes.
  - – 4. It is difficult to associate a specific piece of news with a stock price change. We will assume that only news stories mentioning a specific stock will affect that stock's price.

Example: Mining News Stories to Predict Stock Price Movement

- Two-class problem
  - Change: merge surge and plunge into a single class 급등 급락
    - Positive class
  - Stable (no change): stock price between -2.5% and 2.5%
    - Negative class
  - For the zones between 2.5% to 5% and -2.5% to -5%, we will refuse to label them

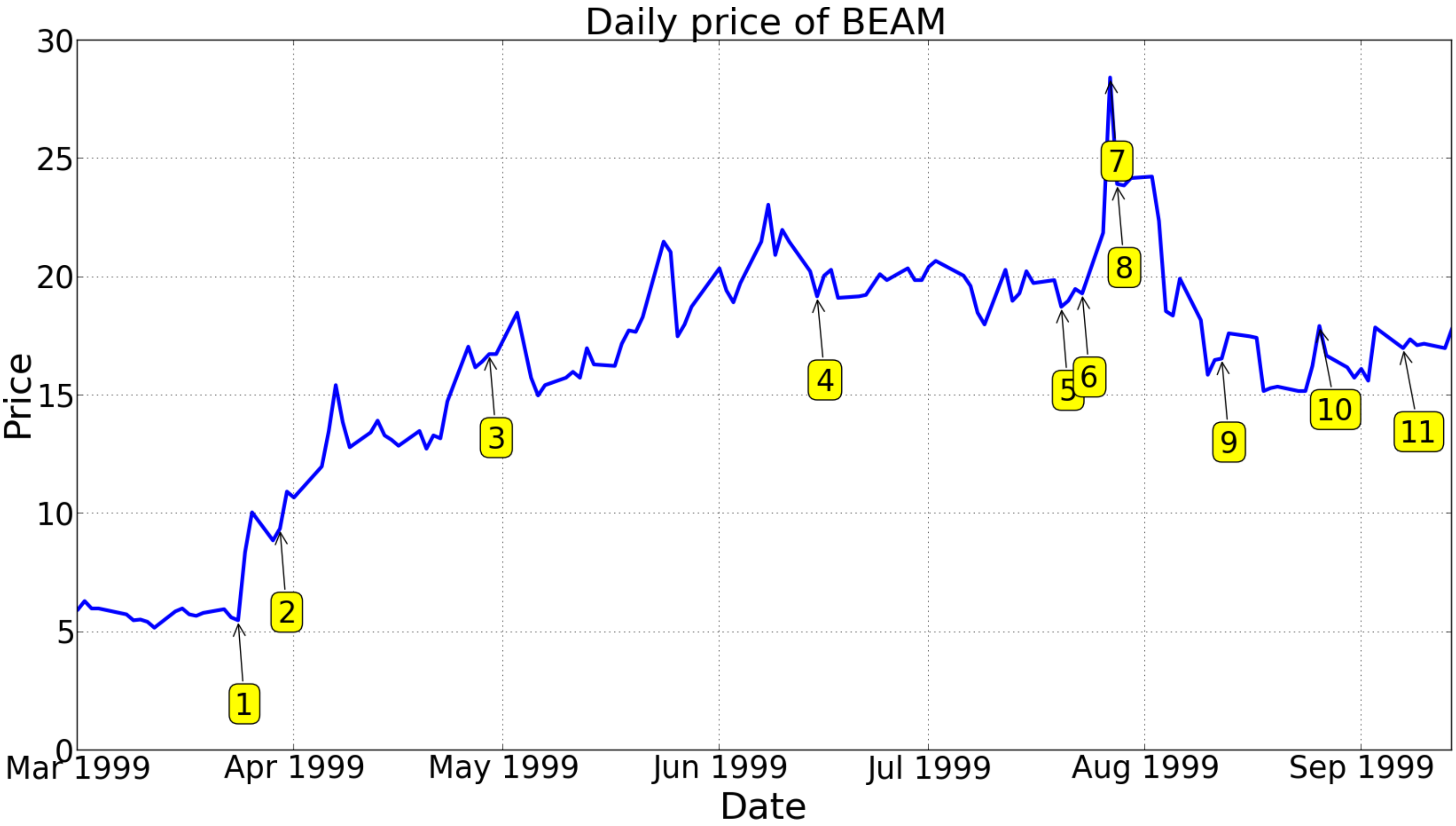


Example: Mining News Stories to Predict Stock Price Movement

- **The Data**: historical data from 1999 for stocks listed on the New York Stock Exchange and NASDAQ 뉴욕증권거래소
  - Open and close stock prices on the major stock exchanges : training data set을 위한 정답(실제값)
  - A large compendium of financial news stories throughout the year 요약본
    - Nearly 36000 stories altogether
- A sample news story

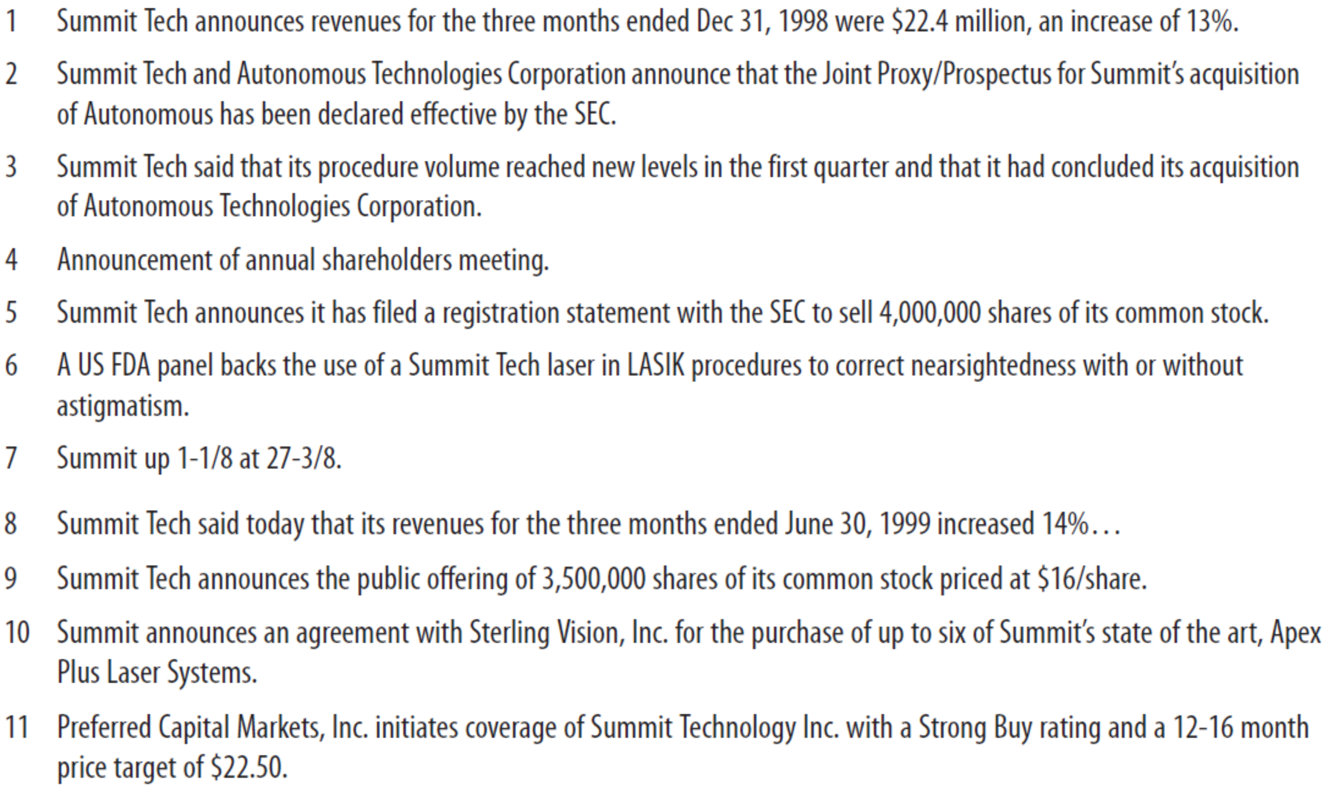
1999-03-30 14:45:00 과식과석기계?
WALTHAM, Mass.--(BUSINESS WIRE)--March 30, 1999--Summit Technology, Inc. (NASDAQ:BEAM) and Autonomous Technologies Corporation (NASDAQ:ATCI) announced today that the Joint Proxy/Prospectus for Summit's acquisition of Autonomous has been declared effective by the Securities and Exchange Commission. Copies of the document have been mailed to stockholders of both companies. "We are pleased that these proxy materials have been declared effective and look forward to the shareholder meetings scheduled for April 29," said Robert Palmisano, Summit's Chief Executive Officer.

Example: Mining News Stories to Predict Stock Price Movement



Example: Mining News Stories to Predict Stock Price Movement

- Annotation with news story summaries



1 Summit Tech announces revenues for the three months ended Dec 31, 1998 were $22.4 million, an increase of 13%.

2 Summit Tech and Autonomous Technologies Corporation announce that the Joint Proxy/Prospectus for Summit's acquisition of Autonomous has been declared effective by the SEC.

3 Summit Tech said that its procedure volume reached new levels in the first quarter and that it had concluded its acquisition of Autonomous Technologies Corporation.

4 Announcement of annual shareholders meeting.

5 Summit Tech announces it has filed a registration statement with the SEC to sell 4,000,000 shares of its common stock.

6 A US FDA panel backs the use of a Summit Tech laser in LASIK procedures to correct nearsightedness with or without astigmatism.

7 Summit up 1-1/8 at 27-3/8.

8 Summit Tech said today that its revenues for the three months ended June 30, 1999 increased 14%. . .

9 Summit Tech announces the public offering of 3,500,000 shares of its common stock priced at $16/share.

10 Summit announces an agreement with Sterling Vision, Inc. for the purchase of up to six of Summit's state of the art, Apex Plus Laser Systems.

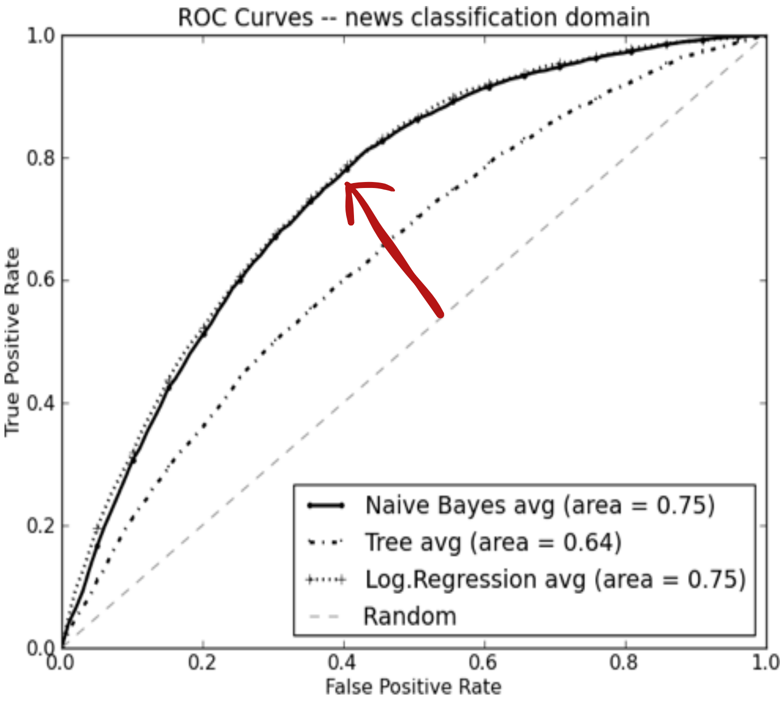
11 Preferred Capital Markets, Inc. initiates coverage of Summit Technology Inc. with a Strong Buy rating and a 12-16 month price target of $22.50.

Example: Mining News Stories to Predict Stock Price Movement

- **Data Preprocessing**: the basic steps in bag of words were applied to reduce each story to a TFIDF representation
  - Each word was case-normalized and stemmed, and stopwords were removed
  - Created n-grams up to two. That is, every individual term and pair of adjacent terms were used to represent each story
  - Each story is tagged with a label (change or no change) based on the associated stock price movement
    - Results in about 16000 usable tagged stories: 75% no change, 25% change (13% surge and 12% plunge)

Example: Mining News Stories to Predict Stock Price Movement

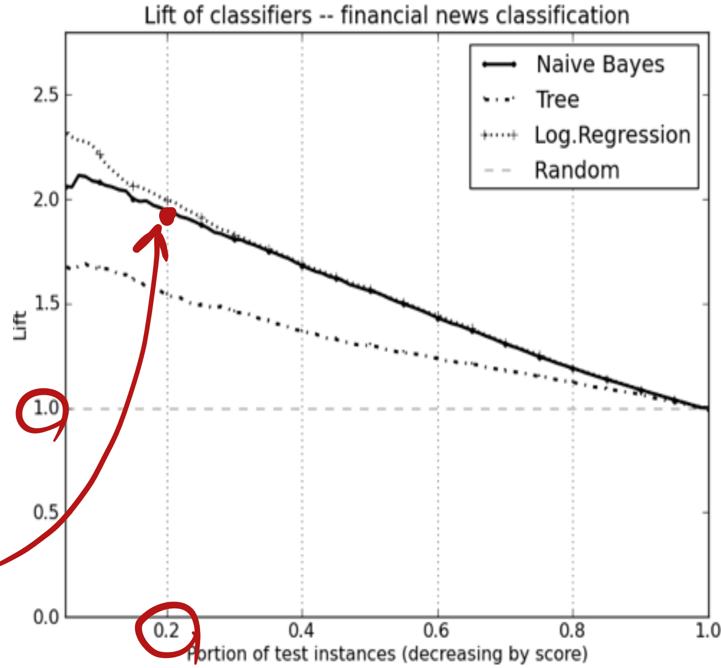
- ROC curve: to look at predictability
  - These curves are averaged from 10-fold cross validation
- Results
  - 1. there is predictive signal in the news stories: significant bowing out (all substantially above 0.5)
  - 2. NB and LR perform better than Tree



Mining News Stories to Predict Stock Price Movement

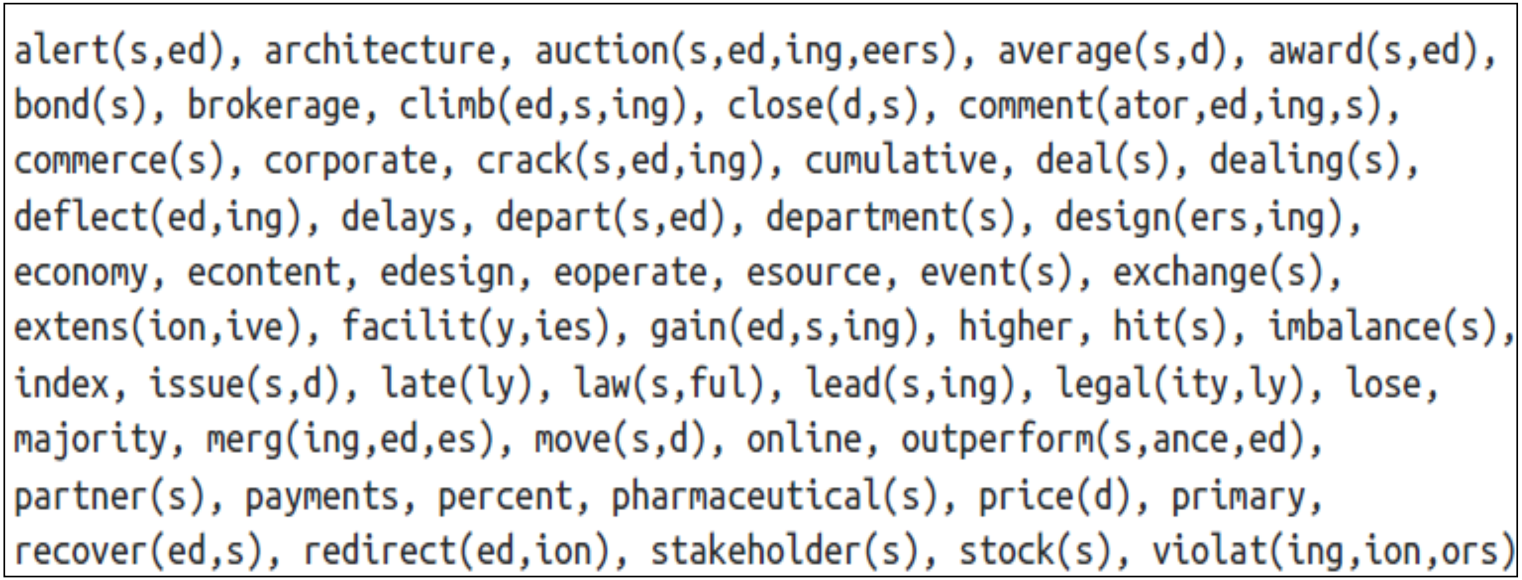
- Lift curves 얼마만큼 정확한지
  - Shows the lift in precision we would get if we used the model to score and order the news stories

- Results
  - Consider the point at x=0.2, where the lifts of LR and NB are around 2.0
  - This means if you were to score all the news stories and take the top 20%, you'd have twice precision of finding a positive story than random guessing



Mining News Stories to Predict Stock Price Movement

- A list of terms with high information gain
  - Many of these suggest significant announcements of good or bad news for company or its stock price



alert(s,ed), architecture, auction(s,ed,ing,eers), average(s,d), award(s,ed), bond(s), brokerage, climb(ed,s,ing), close(d,s), comment(ator,ed,ing,s), commerce(s), corporate, crack(s,ed,ing), cumulative, deal(s), dealing(s), deflect(ed,ing), delays, depart(s,ed), department(s), design(ers,ing), economy, econtent, edesign, eoperate, esource, event(s), exchange(s), extens(ion,ive), facilit(y,ies), gain(ed,s,ing), higher, hit(s), imbalance(s), index, issue(s,d), late(ly), law(s,ful), lead(s,ing), legal(ity,ly), lose, majority, merg(ing,ed,es), move(s,d), online, outperform(s,ance,ed), partner(s), payments, percent, pharmaceutical(s), price(d), primary, recover(ed,s), redirect(ed,ion), stakeholder(s), stock(s), violat(ing,ion,ors)

Summary

- Difficulties of dealing with text
  - Our problems do not always present us with data in a neat feature vector representation
  - Real-world problems often require some form of data representation engineering

- A common way to turn text into a feature vector
  - Break each document into individual words (i.e., bag of words representation), and assign values to each term using TFIDF formula
  - Relatively simple, inexpensive and versatile, and requires little domain knowledge
  - In spite of its simplicity, it performs surprisingly well on a various tasks