비정형텍스트분석

7. GloVe, LDA, Elmo

2019년 10월 12일

데이터사이언스학과 / 이영훈

SEOULTECH

Embedding with GloVe

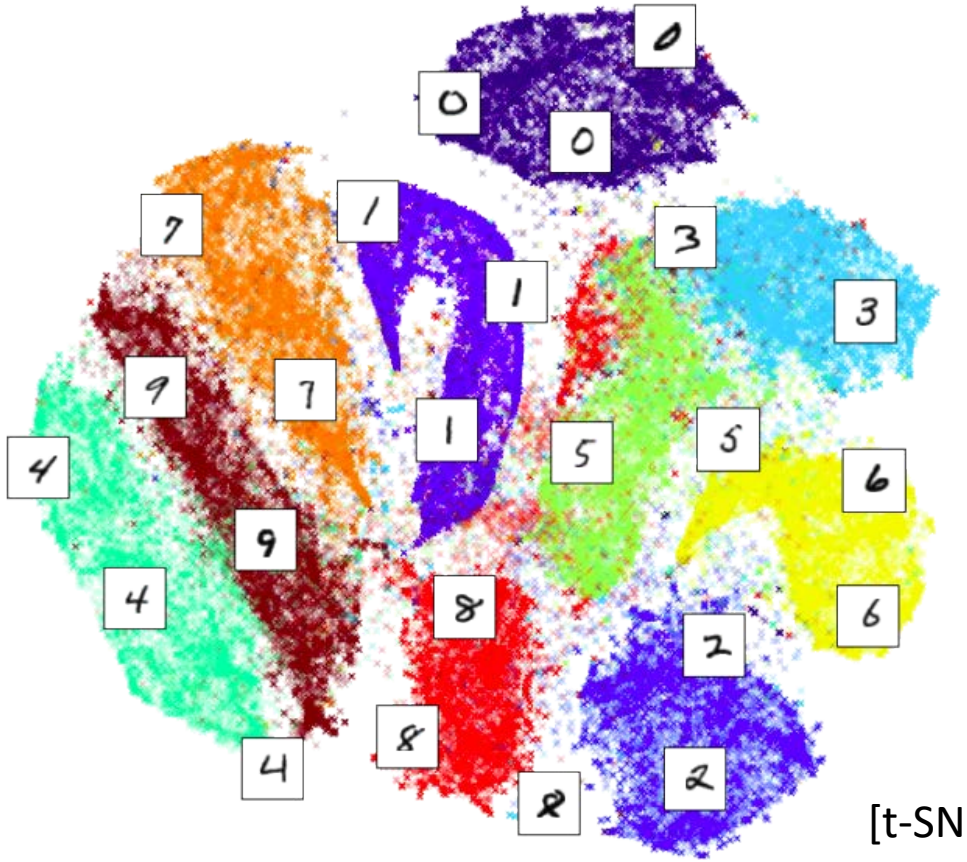
Embedding

- **임베딩은 X라는 공간의 데이터에서 원하는 정보를 잘 저장하며, Y라는 새로운 공간으로 보내는 f: X → Y 함수**
  - (예시) 10,000개 단어로 이뤄진 문서 (1만차원)들의 유사도를 잘 보존하여 2차원으로 보내는 것
  - **어떤 정보를 보존할 것이냐**에 따라서 다양한 임베딩 방법이 존재

- **(벡터) 시각화는 고차원으로 표현되는 객체를 2차원의 저차원 백터로 표현하는 것**
  - 임베딩을 흔히 차원축소라고 부르는 이유

Embedding

- **MDS, LLE, ISOMAP, t-SNE 은 모두 행렬(벡터) 형태로 표현되는 데이터를 다른 차원** (주로 저차원)**의 벡터로 표현하는 임베딩 방법**



[t-SNE 예시]

Embedding

- **Dimensionality Reduction**
  - Number of dimensions = Number of words, phrases, etc.
  - A very important part of text processing
  - Dimensionality reduction of text data is similar to that of structured data

  → Feature subset selection

  → Feature extraction

Embedding

- **Dimensionality Reduction: Feature Subset Selection**
  - Select only the best features for further analysis

    → The most frequent

    → The most informative relative to the all class values

  - Scoring methods for individual feature

Information gain: $\sum_{F=W,\overline{W}} P(F) \sum_{C=pos,neg} P(C|F) \log \frac{P(C|F)}{P(C)}$

Cross-entropy: $P(W) \sum_{C=pos,neg} P(C|W) \log \frac{P(C|W)}{P(C)}$

Mutual information: $\sum_{C=pos,neg} P(C) \log \frac{P(W|C)}{P(W)}$

Weight of evidence: $\sum_{C=pos,neg} P(C)P(W) \left| \log \frac{P(C|W)(1-P(C))}{P(C)(1-P(C|W))} \right|$

Odds ratio: $\log \frac{P(W|pos) \times (1-P(W|neg))}{(1-P(W|pos)) \times P(W|neg)}$

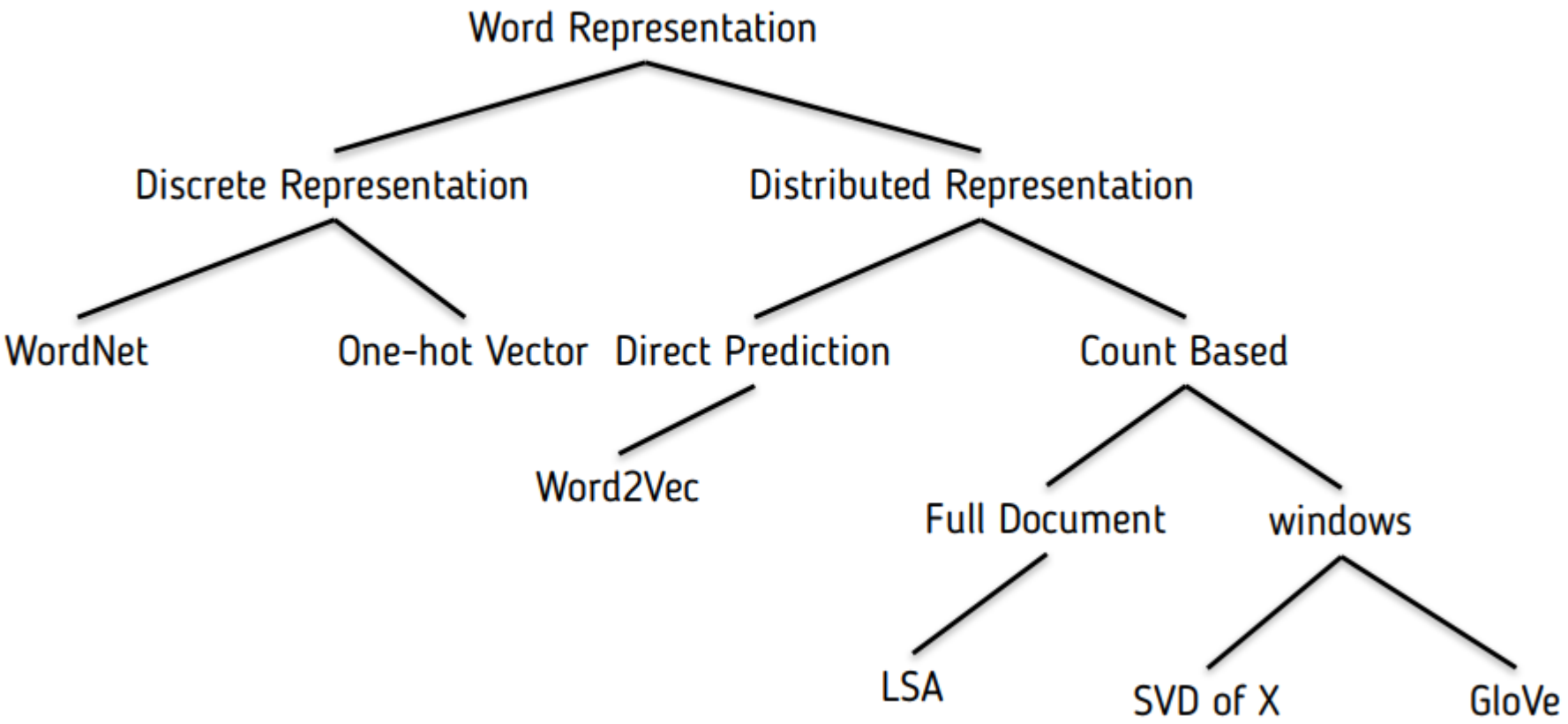
Frequency: $Freq(W)$

Embedding

- **Dimensionality Reduction: (Embedded) Feature Extraction**
  - Principal Component Analysis (PCA)
  - Singular Value Decomposition (SVD) = Latent Semantic Indexing (LSI)
    → Traditional method for matrix factorization
    → Widely used in information retrieval and indexing
  - Latent Dirichlet Allocation (LDA)
    → Widely used in topic modeling

Embedding

- **Word Representation 분류**



Embedding

- **Discrete Representation (Symbolic)**
  - Dictionary 기반 (e.g. WordNet) 혹은 One-hot Vector를 통한 Representation
  - 장점

    → (Dictionary 기반의 경우) 사람이 이해할 수 있는 형태의 Representation

    → (One-hot Vector의 경우) 비교적 간단하게 구축할 수 있다
  - 단점

    → 단어의 관계(e.g. 유사도, 반의어, 문법 등)를 측정할 수 없다

    → 사람이 직접 구축해야한다. 주관적인 판단이 개입될 수 있다

    → 새로운 단어가 나올 경우 일일이 대응해야한다

    → 뉘앙스와 같은 것들을 표현하기 어렵다

Embedding

- **Distributed Representation**
  - 단어의 출현 빈도를 기반으로 계산한 Word Vector

  - 장점

    → 단어의 관계를 측정, 표현 할 수 있다

    → 비지도 학습!

    → 새로운 단어가 나올 경우 Corpus만 제공하면 된다

    → 다른 모델들과 결합해서 추가적인 정보를 제공한다

  - 단점

    → 성능을 측정하기가 쉽지 않다

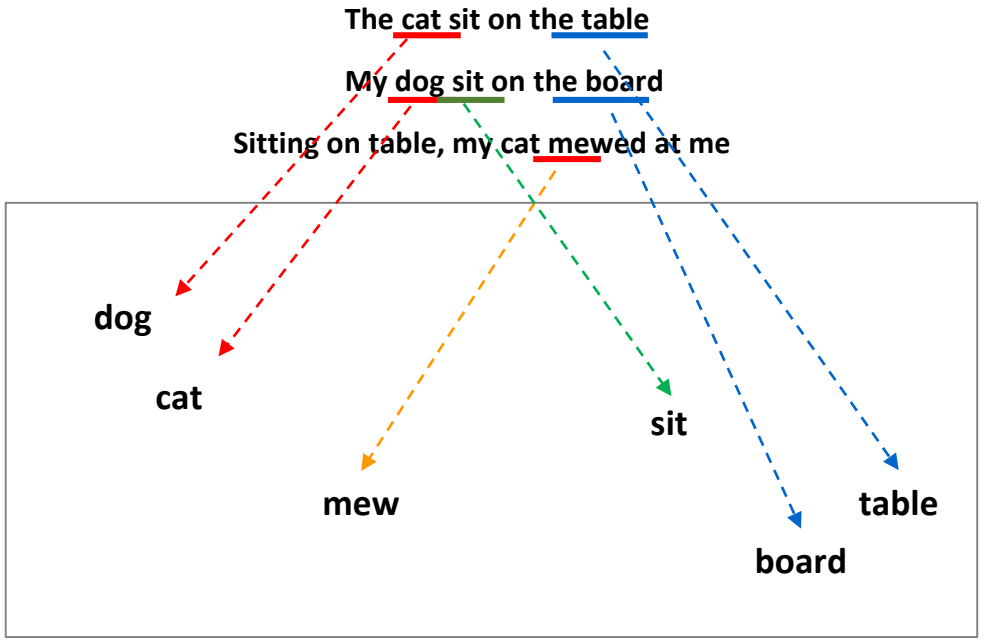
Embedding - Word2Vec

- **Word2Vec, Doc2Vec 은 행렬 형태로 데이터를 가공하지 않은 채 단어나 문서를 벡터로 표현하는 임베딩 방법**
  - 행렬 형태로 가공하지 않는다는 것은 알고리즘을 돌릴 때의 입장이며,
  - 수학적으로 해석하면 매우 큰 sparse vector를 저차원의 dense vector로 표현하는 방법
  - Word2Vec, Doc2Vec을 이해하는 키는 그들이 **"어떤 정보를 보존"**하며 저차원 dense vector를 학습하는지를 파악하는 것

Embedding - Word2Vec

- **Word2Vec은 단어 주변에 등장하는 다른 단어들의 분포의 유사성을 보존하는 벡터 표현 방법**

  ◦ 주위에 등장하는 단어 분포가 유사한 두 단어는 비슷한 벡터를 지님
  'cat' 대신 'dog'을 넣어도 큰 무리가 없음



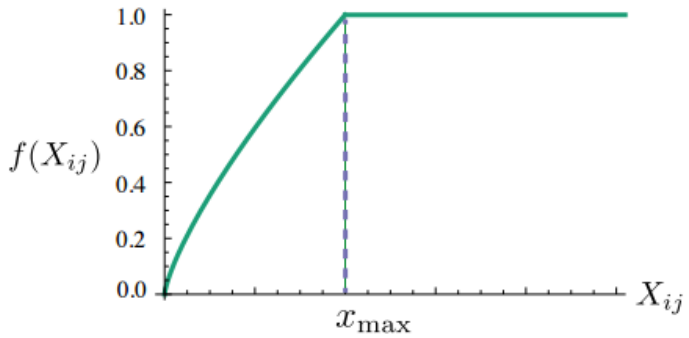
< 단어의 의미 공간 >

Embedding - GloVe

- **Word2Vec은 $P(W_t|W_{[t-2:t+2]})$처럼 단어의 등장 확률을 보존하지만,**
  GloVe는 두 단어 w_i, w_j의 co-occurrence frequency W_{ij}를 보존하는 임베딩

  ◦ 두 단어의 임베딩 벡터의 곱이 log co-occurrence 가 되도록 임베딩

$$J = \sum_{i,j} f(X_{ij}) * \left(w_i^T w_j - log(X_{ij})\right)^2$$



Embedding - GloVe

- **통계 정보를 활용한 Word Vector**
  - Co-occurrence matrix X 를 구축하자

- **Full Document 기반**
  - 단어-문서간 동시 출현을 기반으로 matrix X를 구축한다
  - 일반적인 주제 분류에 적합하다.
  - Latent Semantic Analysis

- **Window 기반**
  - 각 단어의 위치로 단어-단어간 동시 출현을 matrix X를 구축한다
  - 의미와 문법 정보를 모두 캡쳐할 수 있다.

GloVe: Global Vectors for Word Representation (2014) J.Pennington et al. (발표: 곽근봉)

Embedding - GloVe

- **동시등장확률(the words'probabilityof co-occurrence)**

Probability and Ratio	$k = solid$	$k = gas$	$k = water$	$k = fashion$
$P(k\|ice)$	1.9×10^{-4}	6.6×10^{-5}	3.0×10^{-3}	1.7×10^{-5}
$P(k\|steam)$	2.2×10^{-5}	7.8×10^{-4}	2.2×10^{-3}	1.8×10^{-5}
$P(k\|ice)/P(k\|steam)$	8.9	8.5×10^{-2}	1.36	0.96

Embedding - GloVe

- **Objective Function**

$$F(w_i, w_j, \tilde{w}_k) = \frac{P_{ik}}{P_{jk}}$$

① i,j가 각각 주어졌을 때 k가 나올 확률의 차이를 기반으로 i와 j의 의미 차이를 표현해보자

$$F(w_i - w_j, \tilde{w}_k) = \frac{P_{ik}}{P_{jk}}$$

② i,j의 의미 차이이니 (-) 연산으로 모델링을 해보자.

$$F((w_i - w_j)^T \tilde{w}_k) = \frac{P_{ik}}{P_{jk}}$$

③ i,j와 k 간의 관계에 기반하는 상황이니 (i-j)와 K의 관계는 내적으로 모델링을 해보자.

$$F((w_i - w_j)^T \tilde{w}_k) = \frac{F(w_i^T \tilde{w}_k)}{F(w_j^T \tilde{w}_k)}$$

④ I,j가 주어졌을 때 k가 나오는 확률에 기반하여 i,j의 의미 차이를 표현하기 위해 우변을 비율로 모델링하자.

$$F(w_i^T \tilde{w}_k - w_j^T \tilde{w}_k) = \frac{F(w_i^T \tilde{w}_k)}{F(w_j^T \tilde{w}_k)}$$

Embedding - GloVe

- **Objective Function**

$$w_i \longleftrightarrow \tilde{w}_k$$

$$X \longleftrightarrow X^T$$

$$F(X - Y) = \frac{F(X)}{F(Y)}$$

① 임베딩 공간이 바뀌어도 의미 연산이 가능해야 함

② Co-occurrence에 기반하기 때문에 대칭행렬이어야 함

③ 앞의 가정에 맞는 함수가 필요함

Embedding - GloVe

- **Objective Function**

$$exp(w_i^T \tilde{w}_k - w_j^T \tilde{w}_k) = \frac{exp(w_i^T \tilde{w}_k)}{exp(w_j^T \tilde{w}_k)}$$

이 조건을 만족하는 함수는 exp

$$w_i^T \tilde{w}_k = \log P_{ik} = \log X_{ik} - \log X_i$$

i가 주어졌을 때, k의 등장 확률이니 조건부 확률

$$w_i^T \tilde{w}_k = \log X_{ik} - b_i - \tilde{b}_k$$

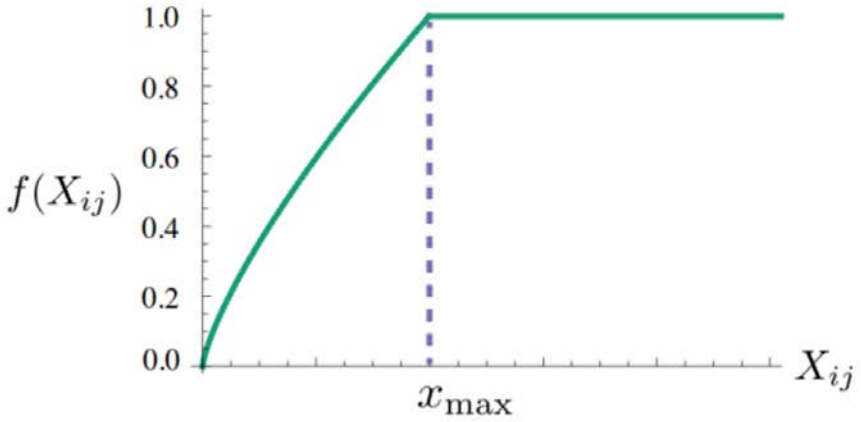
log Xi를 재정의함으로써 앞의 가정을 만족시킴

$$w_i^T \tilde{w}_k + b_i + \tilde{b}_k = \log X_{ik}$$

$$J = \sum_{i,j=1}^{V} (w_i^T \tilde{w}_j + b_i + \tilde{b}_j - \log X_{ij})^2$$

Embedding - GloVe

- **Objective Function**



지나치게 많이 등장하는 단어는 특정 값으로 통일하는 함수

$$J = \sum_{i,j=1}^{V} f\left(X_{ij}\right)\left(w_i^T \tilde{w}_j + b_i + \tilde{b}_j - \log X_{ij}\right)^2$$

$$where \quad f(x) = \begin{cases} \left(\frac{x}{x_{max}}\right)^{\alpha} \\ 1 \quad otherwise \end{cases} \quad if \quad x < x_{max}$$

Embedding - GloVe

- **전체 모델 요약 설명**

"저는 딥러닝을 활용한 자연어 처리에 관심이 많습니다"

"저는 딥러능을 응용한 소프트웨어 개발에 관심이 많습니다"

1. 주어진 Corpus와 Window size를 가지고 co-occurrence matrix X를 만든다

2. Word2Vec과 유사한 방법으로 학습 대상이 되는 단어들을 Window size안에서 고른다

3. 고른 단어와 matrix X 를 기반으로 Objective Function을 사용해서 학습시킨다

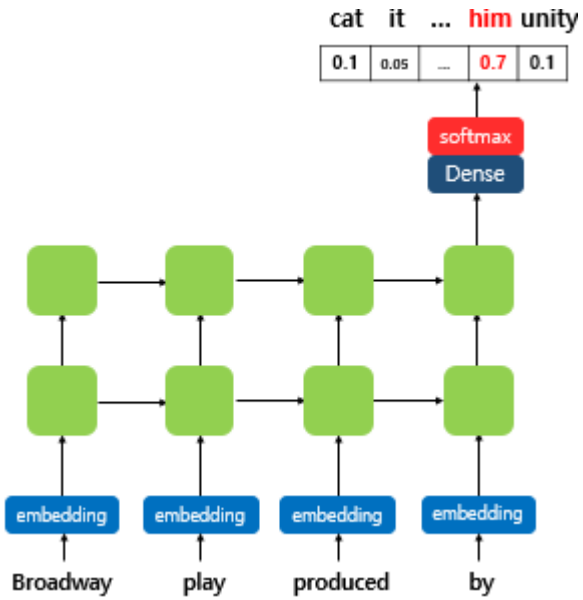
Embedding with Elmo

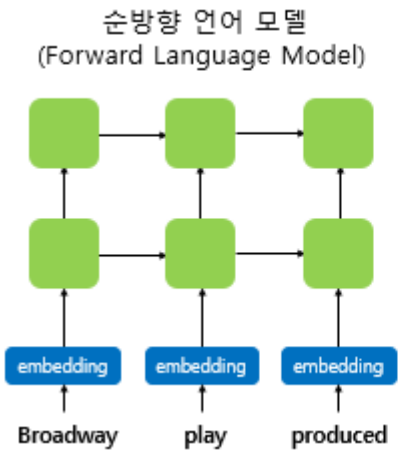
Embedding with Elmo

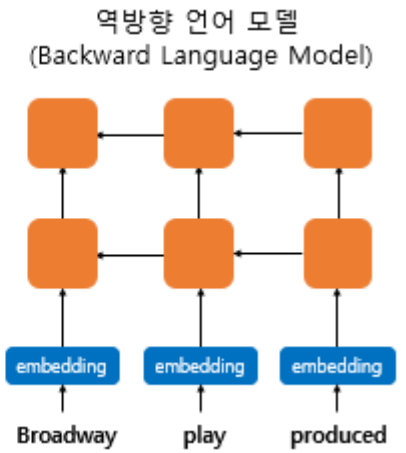
- ELMo
  - ELMo(Embeddings from Language Model)는 2018년에 제안된 새로운 워드 임베딩 방법론
  - ELMo의 가장 큰 특징은 사전 훈련된 언어 모델(Pre-trained language model)을 사용한다는 점
    - Bank Account(은행 계좌)와 River Bank(강둑)에서의 Bank는 전혀 다른 의미를 가지는데,
      Word2Vec이나 GloVe 등으로 표현된 임베딩 벡터들은 이를 제대로 반영하지 못한다는 단점이 있음
    - 예를 들어서 Bank란 단어를 [0.2 0.8 -1.2]라는 임베딩 벡터로 임베딩하였다고 하면,
      Bank는 전혀 다른 의미임에도 불구하고 두 가지 상황 모두에서 [0.2 0.8 -1.2]의 벡터가 사용
  - **문맥을 반영한 워드 임베딩(Contextualized Word Embedding) !!**

Embedding with Elmo

- ELMo의 사전 훈련
  - 기본적으로 RNN 언어 모델링에서 출발



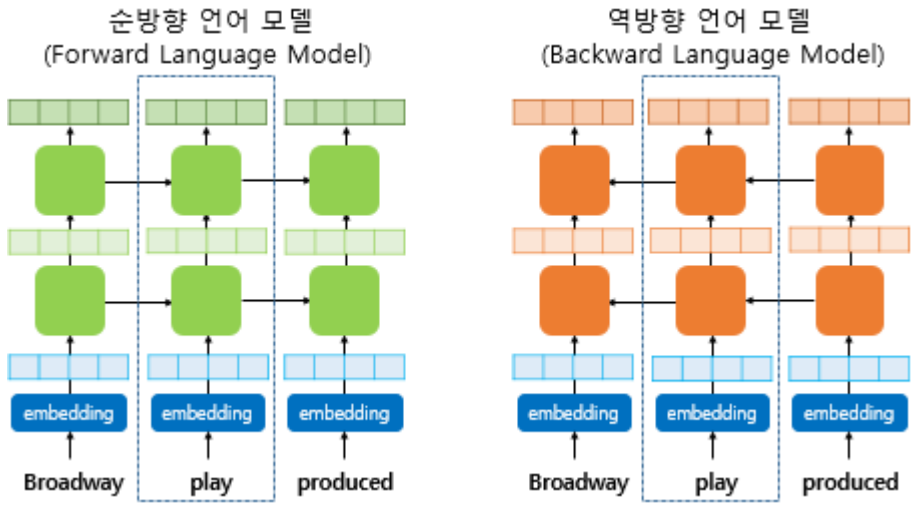




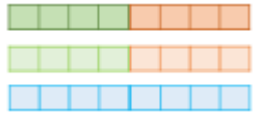
  - LMo는 양쪽 방향의 언어 모델을 둘 다 활용한다고하여 이 언어 모델을 biLM(Bidirectional Language Model)

Embedding with Elmo

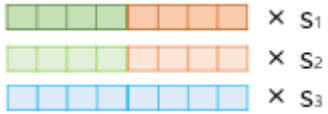
- ELMo의 개별 task 적용
  - 사전 훈련된 biLM에 기반하여 개별 task를 위한 학습 진행.



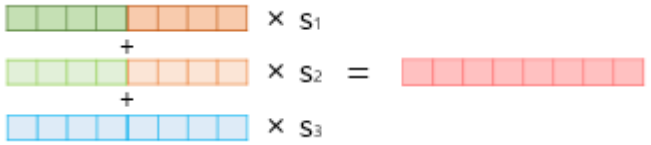
- 1) 각 층의 출력 값을 연결한다.



2) 각 층의 출력값 별로 가중치를 준다.



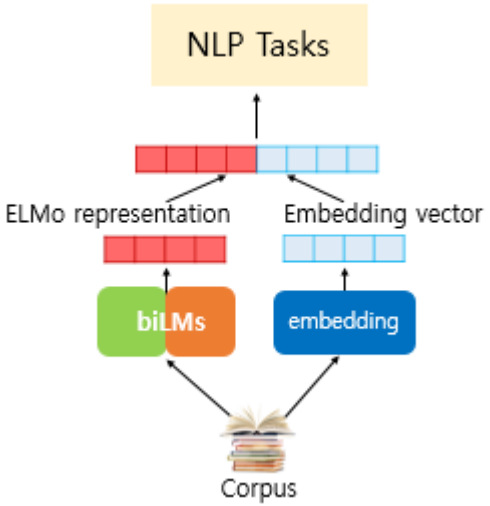
- 3) 각 층의 출력 값을 모두 더한다.



4) 벡터의 크기를 결정하는 스칼라 매개변수를 곱한다.

Embedding with Elmo

- ELMo의 개별 task 적용
  - 완성된 ELMo representation은 기존의 임베딩 벡터와 함께 사용

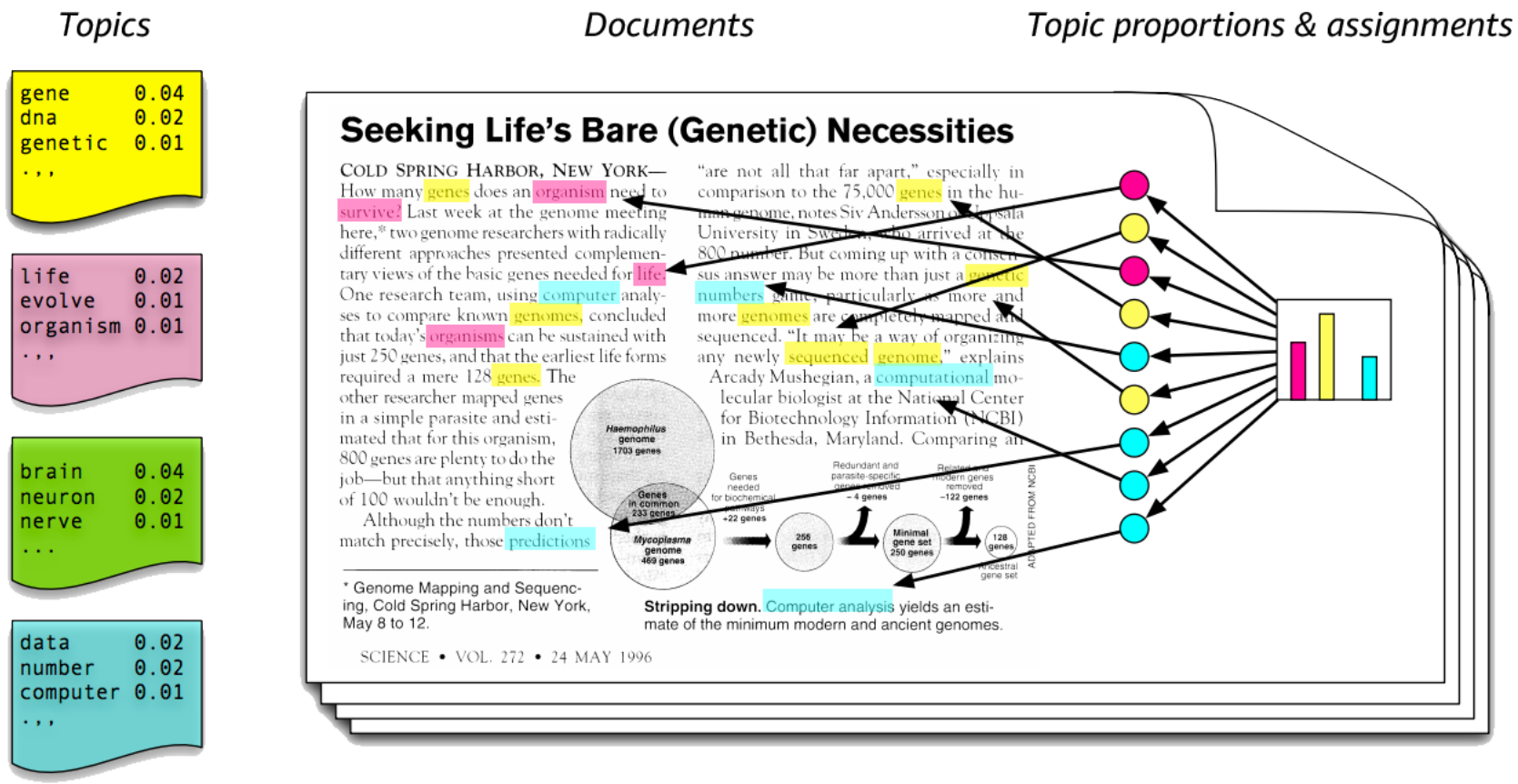


  - 이 때, ELMo 표현을 만드는데 사용되는 사전 훈련된 언어 모델의 가중치는 고정.
  - 그리고 대신 위에서 사용한 s1, s2, s3와 γ는 훈련 과정에서 학습.

LDA

LDA

- **LDA (Latent Dirichlet Allocation)**
  - 주어진 문서에 대하여 각 문서에 어떤 주제들이 존재하는지에 대한 확률 모형
  - LDA는 토픽별 단어의 분포, 문서별 토픽의 분포를 모두 추정

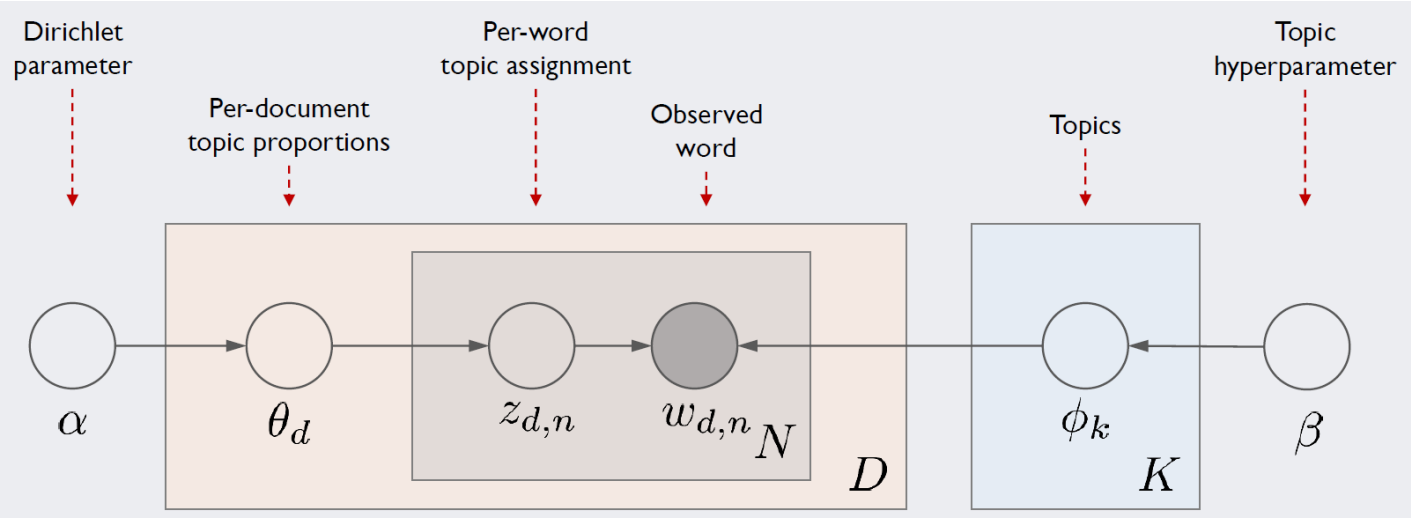


LDA

- **LDA (Latent Dirichlet Allocation)**
  - 노란색 토픽엔 gene이라는 단어가 등장할 확률이 0.04, dna는 0.02, genetic은 0.01
  - 주어진 문서를 보면 파란색, 빨간색 토픽에 해당하는 단어보다는 노란색 토픽에 해당하는 단어들이 많음
  - 우측에 있는 'Topic proportions & assignments'가 LDA의 핵심 프로세스
  - 문서 집합에서 얻은 토픽 분포로부터 토픽을 뽑음. 이후 해당 토픽에 해당하는 단어들을 뽑는다.
    이것이 LDA가 가정하는 문서 생성 과정
  - 실제 문서에 보이는 단어들이 드러난 정보지만 어떤 토픽에서 뽑힌 단어인지 알 수가 없음.
    따라서 이렇게 이면에 존재하는 정보 (잠재된 정보)를 추론하는 것이 LDA.

LDA

- **모델 아키텍처**
  - D 는 말뭉치 전체 문서 개수, K는 전체 토픽 수(하이퍼 파라미터), N은 d번째 문서의 단어 수를 의미
  - 네모칸은 해당 횟수만큼 반복하라는 의미이며 동그라미는 변수를 가리킴
  - 우리가 관찰 가능한 변수는 d번째 문서에 등장한 n번째 단어 wd,n이 유일함
  - 이 정보만을 가지고 하이퍼 파라미터(사용자 지정) α,β를 제외한 모든 잠재 변수를 추정해야 함



(1) Draw each per-corpus topic distributions $\phi_k \sim Dir(\beta)$ for $k \in \{1, 2, \ldots K\}$

(2) For each document, Draw per-document topic proportions $\theta_d \sim Dir(\alpha)$

(3) For each document and each word, Draw per-word topic assignment $z_{d,n} \sim Multi(\theta_d)$

(4) For each document and each word, Draw observed word $w_{d,n} \sim Multi(\phi_{z_{d,n},n})$

LDA

- **LDA 모델의 변수**
  - φk 는 k번째 토픽에 해당하는 벡터, 말뭉치 전체의 단어 개수만큼의 길이를 가짐
  - φ1 은 아래 표에서 첫번째 열. 마찬가지로 φ2는 두번째, φ3은 세번째 열벡터
  - φk 는 하이퍼 파라미터 β에 영향을 받음 (디리클레 분포)

Terms	Topic 1	Topic 2	Topic 3
Baseball	0.000	0.000	0.200
Basketball	0.000	0.000	0.267
Boxing	0.000	0.000	0.133
Money	0.231	0.313	0.400
Interest	0.000	0.312	0.000
Rate	0.000	0.312	0.000
Democrat	0.269	0.000	0.000
Republican	0.115	0.000	0.000
Cocus	0.192	0.000	0.000
President	0.192	0.063	0.000

LDA

- **LDA 모델의 변수**
  - θd 는 d번째 문서가 가진 토픽 비중을 나타내는 벡터φ1 은 아래 표에서 첫번째 열.
  - θ1 은 아래 표에서 첫번째 행벡터, θ5는 다섯번 째 행벡터
  - θd 역시 하이퍼 파라미터 α에 영향을 받음

Docs	Topic 1	Topic 2	Topic 3
Doc 1	0.400	0.000	0.600
Doc 2	0.000	0.600	0.400
Doc 3	0.375	0.625	0.000
Doc 4	0.000	0.375	0.625
Doc 5	0.500	0.000	0.500
Doc 6	0.500	0.500	0.000

LDA

- **LDA 모델의 변수**
  - $Z_{d,n}$ 은 d번째 문서 n번째 단어가 어떤 토픽에 해당하는지 할당해주는 역할을 함
  - 세번째 문서의 첫번째 단어는 Topic1과 2가 뽑힐 확률이 각각 0.375, 0.625이므로 Topic2일 가능성이 높음

  - $W_{d,n}$ 은 문서에 등장하는 단어를 할당해주는 역할을 하고, φ_k와 $z_{d,n}$에 동시에 영향을 받음.
  - 직전 예시에서 $Z_{3,1}$은 실제로 Topic2에 할당됐다고 하면
    $W_{3,1}$은 Topic2의 단어 분포 가운데 Money가 0.313으로 가장 높기 때문에 Money가 될 가능성이 높음

LDA

- **LDA의 inference**
  - Wd,n 를 가지고 잠재변수를 역으로 추정하는 inference 과정
  - 다시 말해 LDA는 토픽의 단어분포와 문서의 토픽분포의 결합으로 문서 내 단어들이 생성된다고 가정
  - Inference는 실제 관찰가능한 문서 내 단어를 가지고 우리가 알고 싶은 토픽의 단어분포, 문서의 토픽분포를 추정하는 과정

Given a dataset $\mathcal{D} = \{x_1, ..., x_n\}$:

Bayes Rule:

$$P(\theta|\mathcal{D}) = \frac{P(D|\theta)P(\theta)}{P(\mathcal{D})}$$

$P(\mathcal{D}\|\theta)$	Likelihood function of θ
$P(\theta)$	Prior probability of θ
$P(\theta\|\mathcal{D})$	Posterior distribution over θ

Computing posterior distribution is known as the **inference** problem.
But:

$$P(\mathcal{D}) = \int P(\mathcal{D}, \theta) d\theta$$

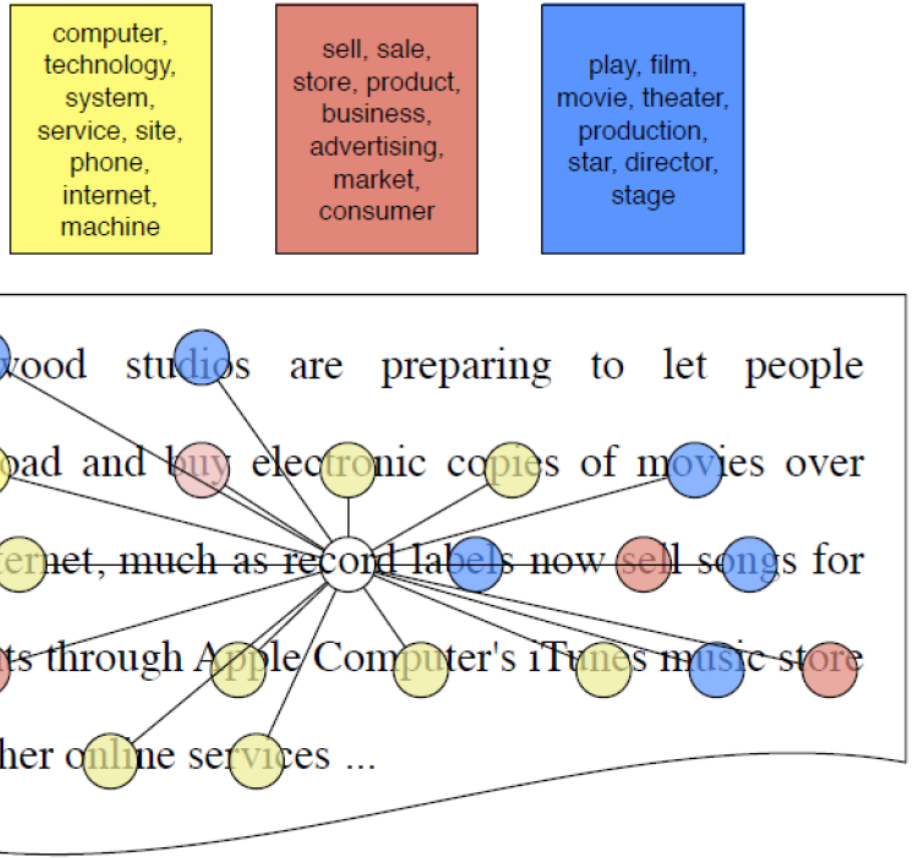
This integral can be very high-dimensional and difficult to compute.

LDA

- **LDA의 inference**
  - 따라서 우리는 사후확률(posterior) p(z,φ,θ|w)를 최대로 만드는 z,φ,θ를 찾아야 함
  - p(w) 는 잠재변수 z,φ,θ의 모든 경우의 수를 고려한 각 단어(w)의 등장 확률을 가리키기 때문에 p(w) 를 단번에 계산하는 것이 어려움
  - 따라서 p(z,φ,θ|w)을 깁스 샘플링을 사용해서 구하게 됨

$$p(z_i = j | z_{-i}, w)$$

(http://i.imgur.com/T9DG4PH.gif)



LDA

- **LDA의 inference**
  - 몇 가지 수식 정리 과정을 거치면 d번째 문서 i번째 단어의 토픽 Zd,i가 j번째에 할당될 확률은 다음과 같이 쓸 수 있음

$$p(z_{d,i}=j|z_{-i},w) = \frac{n_{d,k}+\alpha_j}{\sum_{i=1}^{K}(n_{d,i}+\alpha_i)} \times \frac{v_{k,w_{d,n}}+\beta_{w_{d,n}}}{\sum_{j=1}^{V}(v_{k,j}+\beta_j)} = AB$$

Doc1의 i번째 단어의 토픽 $z_{1,i}$	3	2	1	3	1
Doc1의 n번째 단어 $w_{1,n}$	Etruscan	trade	price	temple	market

구분	Topic1	Topic2	Topic3
Etruscan	1	0	35
market	50	0	1
price	42	1	0
temple	0	0	20
trade	10	8	1
…	…	…	…

LDA

- **LDA의 inference**
  - 첫번째 문서의 두번째 단어의 토픽 ($Z_{1,2}$) 정보를 지운 상태에서, 나머지 단어들의 토픽 할당 정보를 활용해 $P(Z_{1,2})$를 계산

$z_{1,i}$	3	?	1	3	1
$w_{d,n}$	Etruscan	trade	price	temple	market

  - $Z_{1,2}$를 지우고 나니 첫번째 문서엔 1번/3번 토픽이 각각 절반씩 있음
    α는 사용자가 지정하는 하이퍼파라메터로 깁스 샘플링 과정에서 변하는 값이 아니므로
    A의 크기는 n1,1과 n1,3에 영향을 받아 2번 토픽이 될 확률은 0이고, 1번/3번은 절반 정도로 같음

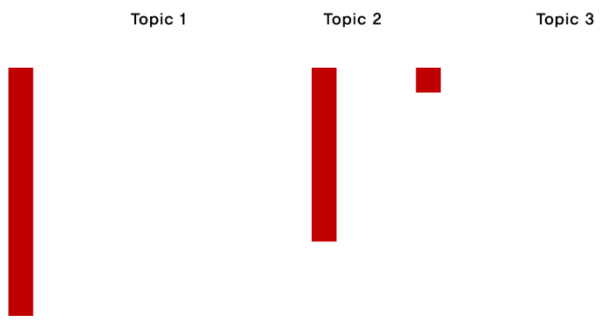
LDA

- **LDA의 inference**
  - 전체 문서 모음을 대상으로 단어들의 토픽 할당 정보를 조사

구분	Topic1	Topic2	Topic3
Etruscan	1	0	35
market	50	0	1
price	42	1	0
temple	0	0	20
trade	**10**	**8-1**	**1**
…	…	…	…

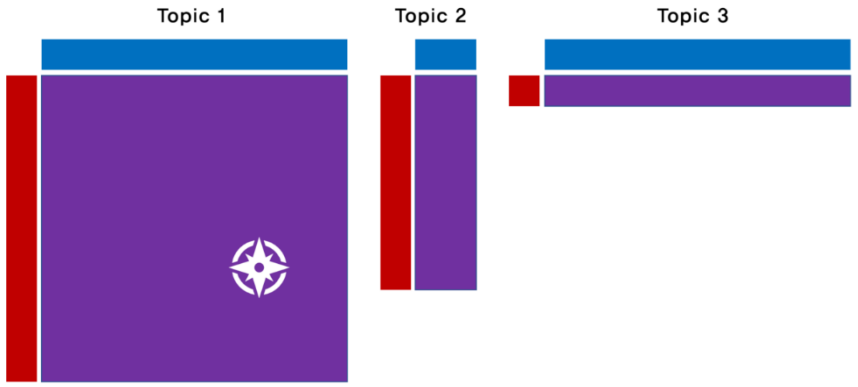
  - B에 적용된 β 역시 샘플링 과정에서 바뀌는 과정이 아니므로 B의 크기는 vk에 가장 많은 영향을 받음

$$v_{1,trade} = 10, v_{2,trade} = 7, v_{3,trade} = 1$$



LDA

- **LDA의 inference**
  - A 와 B를 각각 직사각형의 높이와 너비로 둔다면, p(z1,2)는 아래와 같이 직사각형의 넓이로 이해할 수 있음
  - Z1,2 는 Topic1에 할당될 가능성이 제일 크지만 확률적인 방식으로 토픽을 할당하기 때문
    Topic3에 할당될 가능성도 Topic1에 비해선 작지만 아주 없는 것은 아님



$z_{1,i}$	3	1	1	3	1
$w_{d,n}$	Etruscan	trade	price	temple	market

구분	Topic1	Topic2	Topic3
Etruscan	1	0	35
market	50	0	1
price	42	1	0
temple	0	0	20
trade	**10+1**	7	1
…	…	…	…

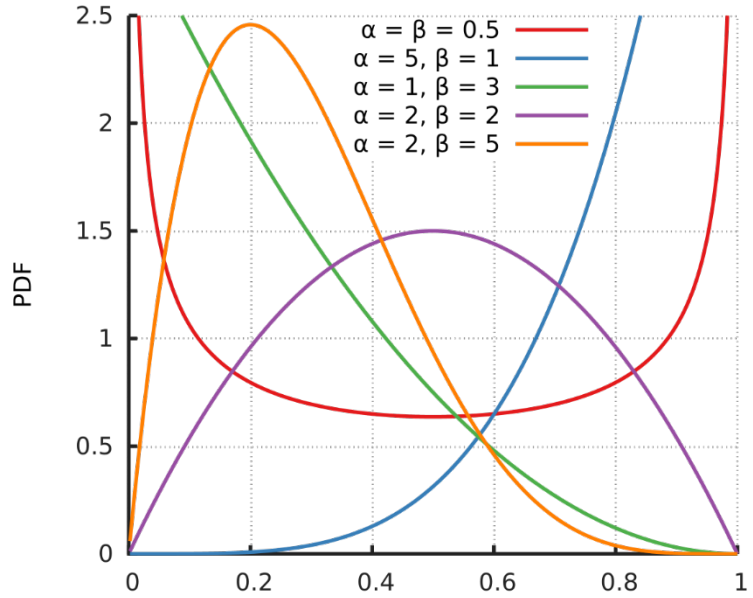
LDA

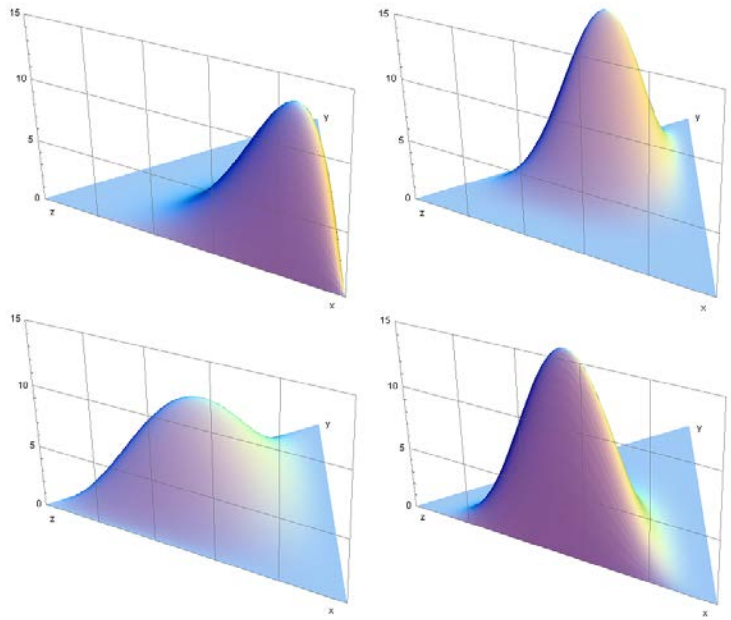
- **디리클레 분포**

$$X \sim Dir(\alpha) \sim Dir(\alpha_1,\ldots,\alpha_k)$$

$$P(X=x) = f(x_1,\ldots,x_k;\alpha_1,\ldots,\alpha_k) = \frac{1}{B(\alpha)}\prod_{i=1}^{k} x_i^{\alpha_i - 1},$$

$$where\ B(\alpha) = \frac{\prod_{i=1}^{k}\Gamma(\alpha_i)}{\Gamma(\sum_{i=1}^{k}\alpha_i)},\ \sum x_i = 1$$





- 디리클레분포는 베타분포의 확장
- 베타분포가 [0, 1]에서의 단항확률변수에 대해 모델링 한 것이라면,
  디리클레분포는 합이 1인 k차원의 다항확률변수에 대해 모델링한 것

LDA

- **디리클레 분포**

1. Choose $N \sim \text{Poisson}(\xi)$.
2. Choose $\theta \sim \text{Dir}(\alpha)$.
3. For each of the N words w_n:
   (a) Choose a topic $z_n \sim \text{Multinomial}(\theta)$.
   (b) Choose a word w_n from $p(w_n \mid z_n, \beta)$, a multinomial probability conditioned on the topic z_n.

  - 우리가 결국 하고 싶은 일은 문서 내용(위 식의 data)으로부터 θ를 추정하는 것.
  - 위에서 Posterior라고 된 부분이 어떤 분포를 따르는지 알고 있으면 이 작업이 그나마 좀 쉬워지고,
  - 수식에서 Posterior와 Prior가 동일한 분포를 따르면, Prior를 Likelihood의 Conjugate Prior라고 한다.
  - Likelihood라고 된 부분은 θ에 대한 다항분포인데 다항분포의 Conjugate Prior가 디리클레 분포다.

$$\underbrace{p(\theta|\text{data})}_{\text{posterior}} \propto \underbrace{\ell(\text{data}|\theta)}_{\text{likelihood}} \underbrace{\tilde{p}(\theta)}_{\text{prior}}$$

토픽 모델링 - LDA

- LDA 결과
  - 각 토픽에 속한 단어를 보고 토픽의 특성을 파악
  - 각 문서가 어떤 토픽을 담고 있는지 파악 가능

Topic 247	Topic 5	Topic 43	Topic 56
"Arts"	"Budgets"	"Children"	"Education"
NEW	MILLION	CHILDREN	SCHOOL
FILM	TAX	WOMEN	STUDENTS
SHOW	PROGRAM	PEOPLE	SCHOOLS
MUSIC	BUDGET	CHILD	EDUCATION
MOVIE	BILLION	YEARS	TEACHERS
PLAY	FEDERAL	FAMILIES	HIGH
MUSICAL	YEAR	WORK	PUBLIC
BEST	SPENDING	PARENTS	TEACHER
ACTOR	NEW	SAYS	BENNETT
FIRST	STATE	FAMILY	MANIGAT
YORK	PLAN	WELFARE	NAMPHY
OPERA	MONEY	MEN	STATE
THEATER	PROGRAMS	PERCENT	PRESIDENT
ACTRESS	GOVERNMENT	CARE	ELEMENTARY
LOVE	CONGRESS	LIFE	HAITI

The William Randolph Hearst Foundation will give $1.25 million to Lincoln Center, Metropolitan Opera Co., New York Philharmonic and Juilliard School. "Our board felt that we had a real opportunity to make a mark on the future of the performing arts with these grants an act every bit as important as our traditional areas of support in health, medical research, education and the social services," Hearst Foundation President Randolph A. Hearst said Monday in announcing the grants. Lincoln Center's share will be $200,000 for its new building, which will house young artists and provide new public facilities. The Metropolitan Opera Co. and New York Philharmonic will receive $400,000 each. The Juilliard School, where music and the performing arts are taught, will get $250,000. The Hearst Foundation, a leading supporter of the Lincoln Center Consolidated Corporate Fund, will make its usual annual $100,000 donation, too.

LDA 예시

- LDA 활용 정도
  - http://www.dbpia.co.kr/search/topSearch?startCount=0&collection=ALL&range=A&searchField=ALL&sort=RANK&query=LDA&srchOption=*&includeAr=false#none

- 사용자 리뷰를 통한 소셜커머스와 오픈마켓의 이용경험 비교분석

〈Table 2〉 The result of LDA topic modeling for social commerce

Topic1 (0.126)	Topic2 (0.072)	Topic3 (0.12)	Topic4 (0.078)	Topic5 (0.119)	Topic6 (0.09)	Topic7 (0.147)	Topic8 (0.111)	Topic9 (0.059)	Topic10 (0.074)
안되	상품	편해	최고	빠르	쿠폰	느림	가격	굳굳	편리
로그인	알리	쿠폰	편해	배송	이벤트	튕김	배송	조음	편해
쿠폰	감사	이용	로켓배송	로켓배송	가입	오류	빠르	입금	조아요
장바구니	사용	최고	쿠폰	최고	할인	상품	편해	빠르고	최고
구매	아깝	편리	조아용	친절	링크	안되	최고	다른	굿굿
접속	그렇게	쇼핑	이벤트	편해	조음	로딩	상품	로켓배송짱	좋아용
오류	바뀜	물건	진짜	빠른배송	편리	검색	이용	좋아유	굿굿굿
안되요	설정	상품	필요	만족	회원가입	장바구니	로켓배송	좋네여	좋아
못하	느림	이벤트	쇼핑	가격	물건	앱이	편리	쇼핑몰	쇼핑
카트	최악	구매	가격	편리	이용	문제	제품	아주좋아요	좋아요

LDA 예시

- Word2Vec과 LDA기법을 이용한 게임 리뷰 분석

토픽 모델링 및 Word2Vec distance matrix 생성

- LDA 를 이용해 전체 리뷰 데이터를 총 12개의 토픽으로 정의
- 총 12 개의 토픽 정의

- LDA 토픽 정의

	Topic	Word 1	Word 2	Word 3	Word 4	Word 5	Word 6	Word 7	Word 8
1	그래픽	play	graphics	thing	player	first	fun	best	Combat
2	스토리	story	pc	multiplayer	system	players	people	controls	Simple
3	캐릭터	characters	old	real	only	world	team	campaign	Sound
4	1인칭 게임	weapons	fps	few	best	fantastic	space	dead	als
5	레벨 디자인	little	story	maps	original	Weapon	sure	part	shooters
6	레벨 디자인2	level	main	own	big	excellent	score	Feels	fan
7	완성도 (버그,콘텐츠 등)	fun	bugs	content	mode	enjoyable	thing	huge	duty
8	그래픽2	graphics	something	engine	voice	title	overall	work	combat
9	시리즈	short	play	series	character	end	perfect	cool	repetitive
10	인공지능 (캐릭터)	ai	story	people	few	money	different	review	system
11	게임플레이	fun	such	fist	shooter	levels	enemy	players	single
12	게임플레이2	play	story	enemies	weapons	fps	pc	best	experience

LDA 예시

- 초기 시청자 반응과 드라마 평균 시청률 사이의 관계: 토픽 모델링 측면에서

표 1. 주제의 이름, 이름을 결정하기 위해 반영한 단어 목록 및 상위 20개 단어 목록

주제 번호	주제 이름	반영한 단어 목록	상위 20개 단어 목록
주제 1	남녀 배우 연기의 조화	케미, 좋다, 여주, 연기, 캐스팅, 아쉽다, 공감, 배우, 남성, 캐릭	케미, 밉다, 노래, 좋다, 지금, 얼굴, 여주, 연기, 캐스팅, 아쉽다, 내일, 공감, 배우, 작다, 남성, 모든, 수목, 티비, 캐릭, 월화
주제 2	비속어와 자극적인 댓글	가슴, 존잘, 유치, 반대, 존나	가슴, 그렇다, 어떻다, 많다, 댓글, 웃기, 드라마, 처음, 요즘, 존잘, 유치, 반대, 저렇다, 보고, 부럽다, 진짜, 한국, 궁금하다, 존나, 무섭다
주제 3	대사, 연기 등 외적 단어로 의견을 표명	사람, 대사, 연기, 드라마, 현실, 생각, 이유	사람, 정말, 저런, 대사, 보고, 다시, 연기, 하나, 드라마, 누가, 많다, 사극, 시청률, 좋다, 힘들다, 현실, 생각, 조금, 그냥, 이유
주제 4	연기와 내용의 재미에 대한 평가	재미있다, 괜찮다, 소름, 매력, 진짜, 아직, 심각하다	재밌다, 연기, 드라마, 괜찮다, 캐릭터, 이제, 소름, 아직, 설정, 시작, 매력, 진짜, 이미지, 배우, 어제, 제목, 감독, 돋다, 나중, 심각하다
주제 5	외모, 표정, 눈빛 등에 대한 찬사	잘생기다, 멋지다, 눈빛, 표정, 설레다, 좋아하다, 아주	잘생기다, 멋지다, 눈빛, 진짜, 표정, 연기, 설레다, 시간, 좋아하다, 이번, 질투, 키스, 화신, 아주, 드라마, 오빠, 아이, 바로, 얼굴, 좋다
주제 6	남자, 여자 등 성 차이에서의 접근	남자, 여자, 남녀, 배우, 실제	남자, 여자, 역시, 생각, 드라마, 연기, 똑같다, 거리, 배우, 성희롱, 자꾸, 영화, 예쁘다, 남녀, 실제, 연기자, 가수, 얘기, 보고, 오랜만
주제 7	방송에 대한 응원과 기대감	여기, 오늘, 파이팅, 응원, 기대, 기대하다	기대, 여기, 오늘, 파이팅, 별로, 우리, 응원, 기대하다, 아프다, 심장, 연기, 이중성, 이후, 다음주, 저기, 모두, 셔츠, FA, 드라마, 동안
주제 8	연기에 대한 약한 평가	연기, 어색하다, 솔직하다, 이쁘다, 제발, 이상하다, 자연스럽다, 그냥, 좋다, 수준	연기, 배우, 어색하다, 정도, 솔직하다, 원래, 드라마, 이쁘다, 연출, 제발, 마지막, 그렇다, 자체, 이상하다, 자연스럽다, 생각, 그냥, 좋다, 수준, 항상
주제 9	연기에 대한 강한 평가	진짜, 연기, 대박, 사랑, 발연기, 좋다, 최고, 이쁘다, 나쁘다, 예쁘다, 뻔하다, 상처, 안습	진짜, 연기, 대박, 장면, 영상, 사랑, 배우, 발연기, 좋다, 몸매, 최고, 이쁘다, 나이, 나쁘다, 예쁘다, 뻔하다, 상처, 보고, 주연, 연습
주제 10	선정성	미치다, 진짜, 진심, 야하다, 섹시하다, 좋다	미치다, 진짜, 그냥, 역할, 진심, 드라마, 원작, 몰입, 연기, 연기력, 다르다, 작가, 혼자, 꿀잼, 야하다, 소리, 좋다, 섹시하다, 방송, 캐릭터
주제 11	여배우의 외모에 대한 칭찬	FA(여자배우), 이쁘다, 재미, 얼굴, 예쁘다, 쩐다, 미모, 외모, 스타일	FA, 연기, 이쁘다, 뭔가, 드라마, 재미, 얼굴, 스토리, 예쁘다, 심하다, 쩐다, 미모, 문제, 프로, 외모, 그렇다, 스타일, 결혼, 해수
주제 12	소재에 대한 얘기	유방암, 엄마, 언니, 주인공	완전, 느낌, 때문, 좋다, 엄마, 유방암, 목소리, 악플, 주인공, 심쿵, 제일, 연기, 분위기, 여성, 제대로, 다른, 이상, 말투, 눈물, 언니
주제 13	소재에 대한 비판	불편하다, 성추행, 계속, 상황, 오늘	계속, 본방, 불편하다, 머리, 성추행, 사수, 모습, 액션, 상황, 얼마나, 발음, 다음, 안보, 인정, 해도, 오늘, 좋다, 그거, 가지, 대단하다
주제 14	남자배우의 외모에 대한 칭찬	MA(남자배우), 너무, 귀엽다, 좋다, 멋있다, 이쁘다, 목소리, 재미있다, 매력, 발성	MA, 너무, 귀엽다, 좋다, 연기, 멋있다, 이쁘다, 목소리, 커플, 재미있다, 캐릭터, 로코, 예쁘다, 매력, 사극, 발성, 분량, 안타깝다, 사진, 주군
주제 15	내용에 대한 비판	싫다, 보기, 무슨, 오글거리다, 아이돌, 난리, 짜증, 장난, 웃음	싫다, 보기, 무슨, 오글거리다, 이해, 내용, 드라마, 아이돌, 부분, 자기, 난리, 한번, 그렇다, 웃음, 장난, 그냥, 언제, 슬프다, 짜증, 때문

Q&A

감사합니다.