

The background is a digital-themed abstract image. It features a glowing blue sphere in the upper right, composed of binary code (0s and 1s). A bright light source behind the sphere creates a lens flare effect. The overall color palette is dark blue and black, with white and light blue highlights from the sphere and text.

Text Mining

1. 토픽모델링

- **Topic modeling**
 - Topic Model
 - Generative probabilistic models for the term frequency of documents in a given corpus
 - 여러 방법들(LSI, pLSI, LDA, etc)이 있으며, LDA도 그 중의 하나
 - LSA(Latent SemanticAnalysis,), pLSA
 - LDA(Latent Dirichlet Allocation)
 - NMF(Non Negative Factorization)
 - *LSA와 NMF는 행렬 분해 기반 토픽 모델링*
 - *pLSA와 LDA 확률 기반의 토픽 모델링*

1. 토픽모델링

- **Topic modeling**
 - LSA(Latent Semantic Analysis,): Truncated SVD

Full SVD

$$A = U \Sigma V^T$$

Truncated SVD

$$A' = U_t \Sigma_t V_t^T$$

$$A = U \Sigma V^T$$

$$A_k = U_k \Sigma_k V_k^T$$

1. 토픽모델링

- **Topic modeling**

- LDA

- Latent Dirichlet Allocation

- 배경

- 예를 들어, 문서1과 문서2가 주제는 유사해도 각 문서에 등장하는 단어의 종류나 빈도는 다를 수 있는데, 단순한 키워드 기반의 모델로는 유사도를 계산하거나 주제 분류를 하는 데에는 한계가 발생
 - 많은 텍스트에 기초에 α 와 β 를 찾고, 개별 문서의 θ 를 계산할 수 있으면, 이 θ 를 가지고 유사도 계산이나 분류 작업을 할 수 있음

- 특징

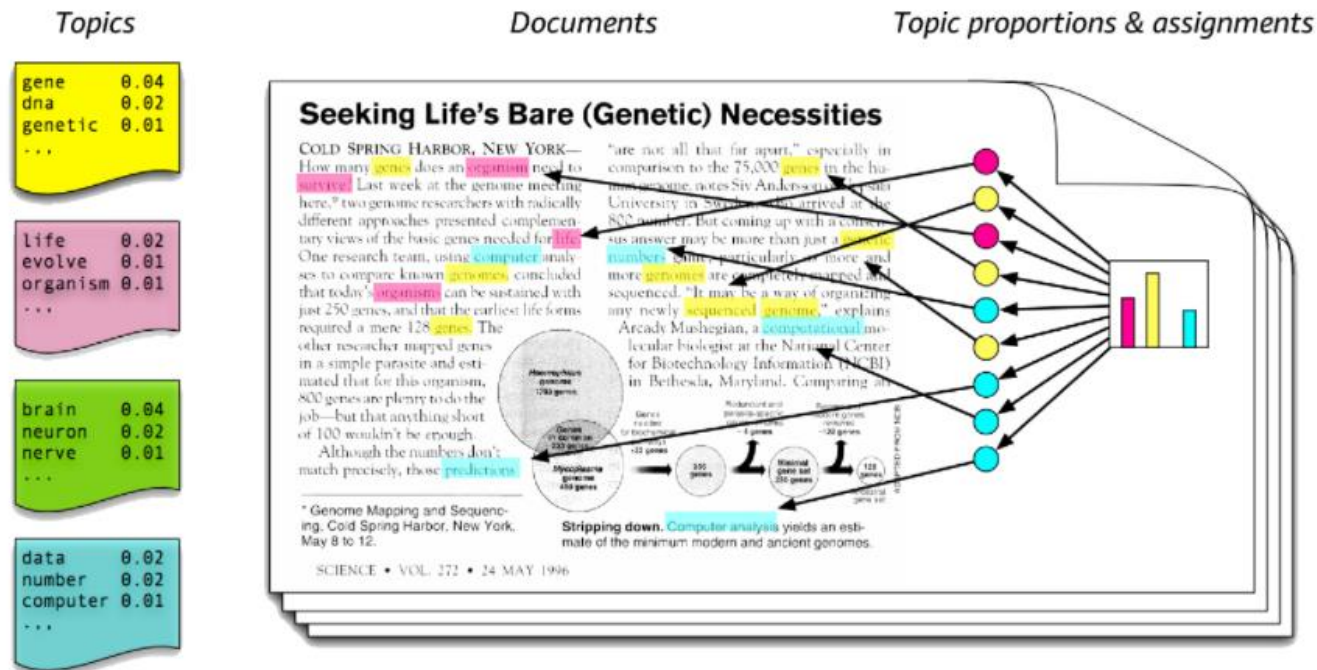
- Bayesian mixture model for discrete data (Topics are assumed to be uncorrelated)
 - Mixture Membership Model (문서는 하나의 토픽에만 속하는 것이 아니라 여러 다른 토픽들의 혼합(Mixture)으로 정의할 수 있음)

1. 토픽모델링

- Topic modeling

- LDA

1. 개별 문서는 혼합된 여러 개의 주제로 구성
2. 개별 주제는 여러 개의 단어로 구성
3. BoW에 기반한 DTM이나 TF-IDF는 기본적으로 단어의 빈도 수를 이용한 수치화
4. 단어의 의미나 순서를 고려하지 못함



1. 토픽모델링

- Topic modeling

Journal of Machine Learning Research 3 (2003) 993-1022

Submitted 2/02; Published 1/03

Latent Dirichlet Allocation

David M. Blei

*Computer Science Division
University of California
Berkeley, CA 94720, USA*

BLEI@CS.BERKELEY.EDU

Andrew Y. Ng

*Computer Science Department
Stanford University
Stanford, CA 94305, USA*

ANG@CS.STANFORD.EDU

Michael I. Jordan

*Computer Science Division and Department of Statistics
University of California
Berkeley, CA 94720, USA*

JORDAN@CS.BERKELEY.EDU

Editor: John Lafferty

Abstract

We describe *latent Dirichlet allocation* (LDA), a generative probabilistic model for collections of discrete data such as text corpora. LDA is a three-level hierarchical Bayesian model, in which each item of a collection is modeled as a finite mixture over an underlying set of topics. Each topic is, in turn, modeled as an infinite mixture over an underlying set of topic probabilities. In the context of text modeling, the topic probabilities provide an explicit representation of a document. We present efficient approximate inference techniques based on variational methods and an EM algorithm for

Dirichlet distribution

-연속 확률분포

-parameter: k차원의 실수(>0)를 갖는 continuous r.v.

-beta distribution의 multivariate generalization

-벡터 내 모든 element를 더한 값이 1인 경우, 다항 분포의 모수에 사용

-다항분포에 대한 conjugate prior

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

$$B(\alpha) = \frac{\prod_{i=1}^k \Gamma(\alpha_i)}{\Gamma(\sum_{i=1}^k \alpha_i)} \quad (\Gamma \text{ is gamma function})$$

1. 토픽모델링

- **Multinomial & Dirichlet Distribution**

- Multinomial Distribution: 여러 개의 값을 가질 수 있는 독립 확률변수들에 대한 확률분포로, 여러 번의 독립적 시행에서 각각의 값이 특정 횟수가 나타날 확률을 정의

$$p(x_1, x_2, \dots, x_n; n, p_1, \dots, p_k) = \frac{n!}{x_1! x_2! \dots x_k!} p_1^{x_1} p_2^{x_2} \dots p_k^{x_k}$$

- Dirichlet Distribution: k차원의 실수 벡터 중 벡터의 요소가 양수이며 모든 요소를 더한 값이 1인 경우에 대해 확률값이 정의되는 분포

$$f(x_1, \dots, x_k; \alpha_1, \dots, \alpha_k) = \frac{1}{B(\alpha)} \prod_{i=1}^k x_i^{\alpha_i - 1} \quad B(\alpha) = \frac{\prod_{i=1}^k \Gamma(\alpha_i)}{\Gamma(\sum_{i=1}^k \alpha_i)}$$

- Conjugate Prior
 - posterior와 prior가 동일한 분포를 따르면, prior를 likelihood의 conjugate prior

Likelihood	Conjugate Prior	Posterior
Binomial(N, θ)	Beta(r,s)	Beta(r+n, s+N-n)
Multinomial($\theta_1, \dots, \theta_K$)	Dirichlet($(\alpha_1, \dots, \alpha_K)$)	Dirichlet($\alpha_1 + n_1, \dots, \alpha_K + n_K$)

1. 토픽모델링

- Topic modeling

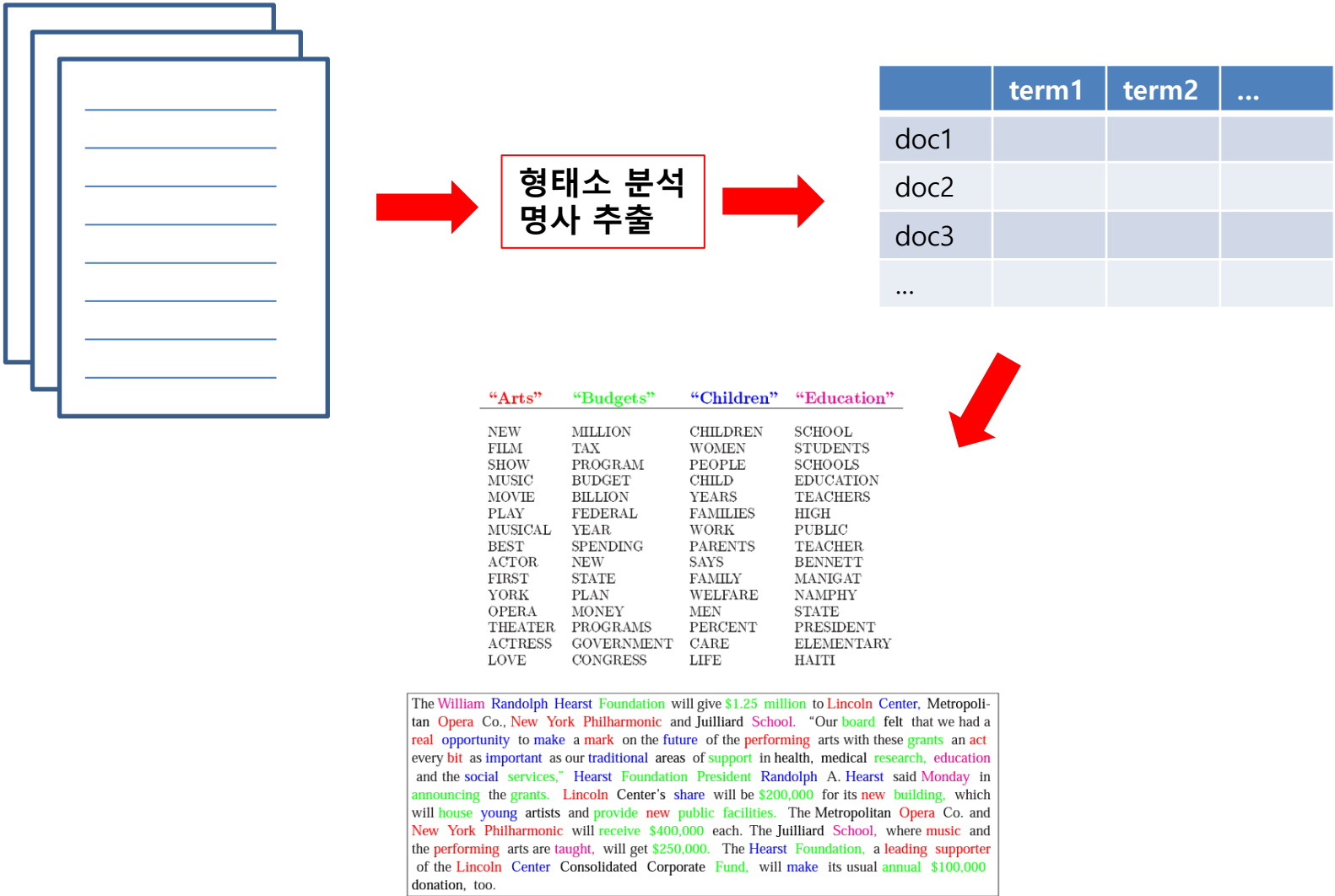
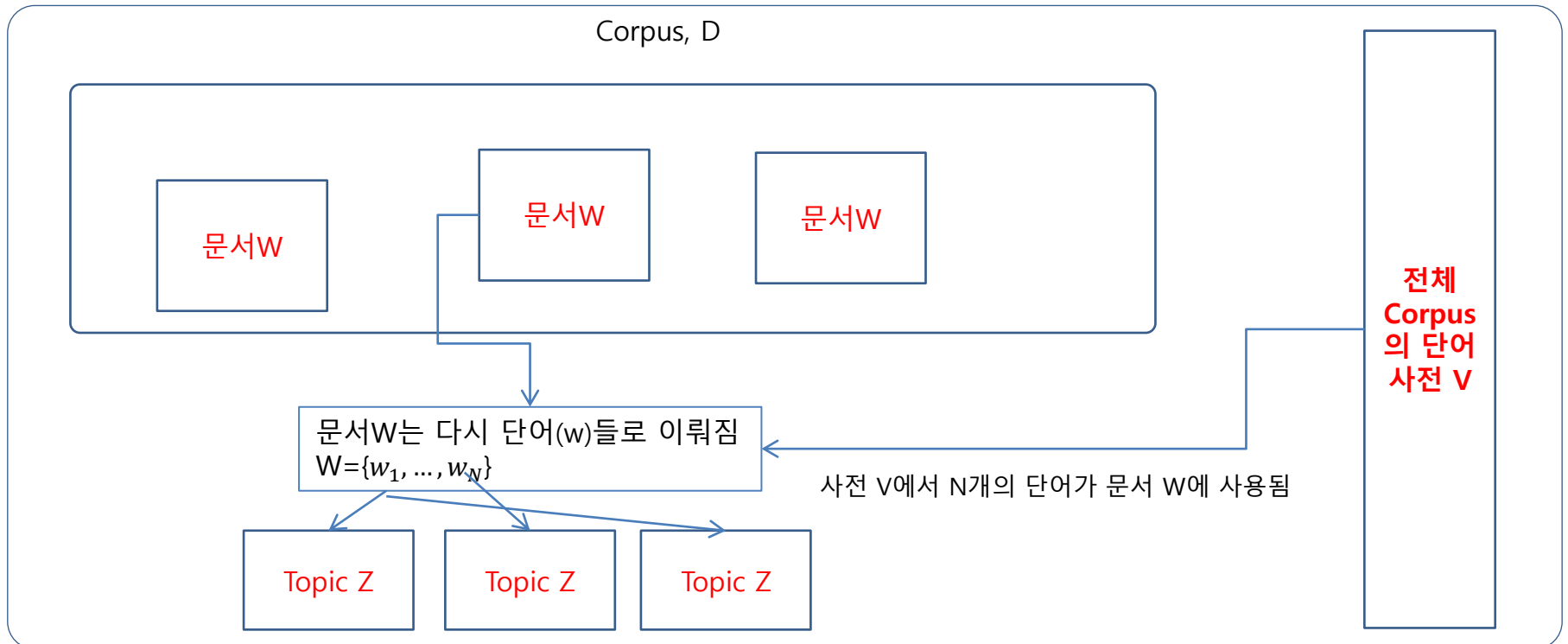


Figure 8: An example article from the AP corpus. Each color codes a different factor from which the word is putatively generated.

1. 토픽모델링

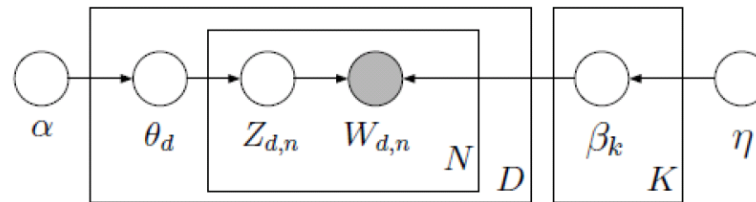
- LDA(Latent Dirichlet Allocation)

- Topics의 수 K 는 사전에 결정되어야 함
- 문서 집합(Corpus)에 대한 Generative Probabilistic Model이며, 1) 문서의 주제 분포와 2) 주제별 단어의 분포를 알고 있다면, 특정 문서가 만들어질 확률을 알 수 있음



1. 토픽모델링

- **LDA Step**
 - Step 1
 - 각 주제($k=1,\dots,K$)에 대해서 Draw Vector of Term Proportion from $\beta_k \sim \text{Dirichlet}_V(\eta)$
 - Step 2
 - 각 문서($d=1,\dots,D$)에 대해서, Draw Topic proportions θ from $\theta_d \sim \text{Dirichlet}_K(\alpha)$
 - Step 3
 - For each of the N words($n=1,\dots,N$) $w_{d,n}$
 - Choose a topic $z_{d,n} \sim \text{Multinomial}_K(1, \theta_d)$
 - Choose a word $w_{d,n}$ from a multinomial probability distribution conditioned on the topic $z_{d,n}$, $\text{Multinomial}_V(1, \beta_{z_{d,n}})$



1. 토픽모델링

- **More on LDA Steps...**

- 위의 과정을 다시 표현하면 아래와 같음

- $w_{d,n} | z_{d,n}, \theta_d, \beta \sim \text{Multinomial}_V(1, \beta_{z_{d,n}})$
- $z_{d,n} | \theta_d \sim \text{Multinomial}_K(1, \theta_d)$
- $\theta_d \sim \text{Dirichlet}_K(\alpha)$
- $(\beta_1, \dots, \beta_K) \sim \prod_{k=1}^K \text{Dirichlet}_V(\eta)$

- α 와 η 가 주어질때, (θ, z, β, w) 의 *Joint Distribution*은 $p(\theta, z, \beta, w; \alpha, \eta)$ 이며 아래와 같이 표시됨
$$p(\theta, z, \beta, w; \alpha, \eta) = [\prod_{k=1}^K p(\beta_k | \eta)] \prod_{d=1}^D [p(\theta_d | \alpha) \prod_{n=1}^N p(z_{d,n} | \theta_d) p(w_{d,n} | z_{d,n}, \theta_d, \beta)]$$

- θ, z, β 는 *parameter*이고 w 는 관측된 데이터여서, (θ, z, β) 의 *Posterior Distribution*은 아래와 같음

$$\frac{(\theta, z, \beta, w) \text{의 } \text{Joint Distribution}}{w \text{의 } \text{Marginal Distribution}} = \frac{p(\theta, z, \beta, w | \alpha, \eta)}{\int \int \sum_z p(\theta, z, \beta, w | \alpha, \eta) d\theta d\beta}$$

- **LDA를 통해 아래의 값을 구하고자 함**

- Topic probability of term $\hat{\beta}_{K,V} = E_{\pi}[\beta_{K,V} | w]$
- Per-Document topic proportion $\hat{\theta}_{d,k} = E_{\pi}[\theta_{d,k} | w]$
- Per-word topic proportion $\hat{z}_{d,n,k} = Pr_{\pi}(z_{d,n} = k | w)$

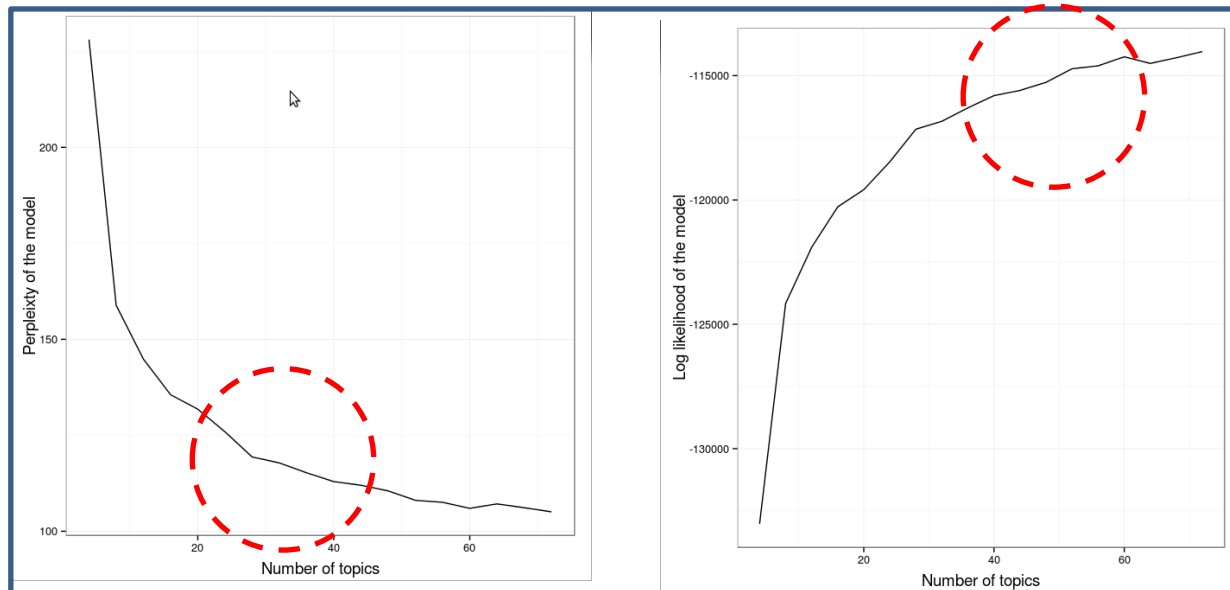
1. 토픽모델링

- **Parameter Estimation**

- 데이터의 log-likelihood 를 최대화하는 Parameter 추정
- 이번 분석에서는 Variational Expectation Maximization Algorithm을 사용

$$l(\alpha, \beta) = \log(p(w|\alpha, \beta)) = \log \int \left\{ \sum_z \left[\prod_{i=1}^N p(w_i|z_i, \beta) p(z_i|\theta) \right] \right\} p(\theta|\alpha) d\theta$$

Model Selection: Choose K



1. 토픽모델링

- Topic modeling

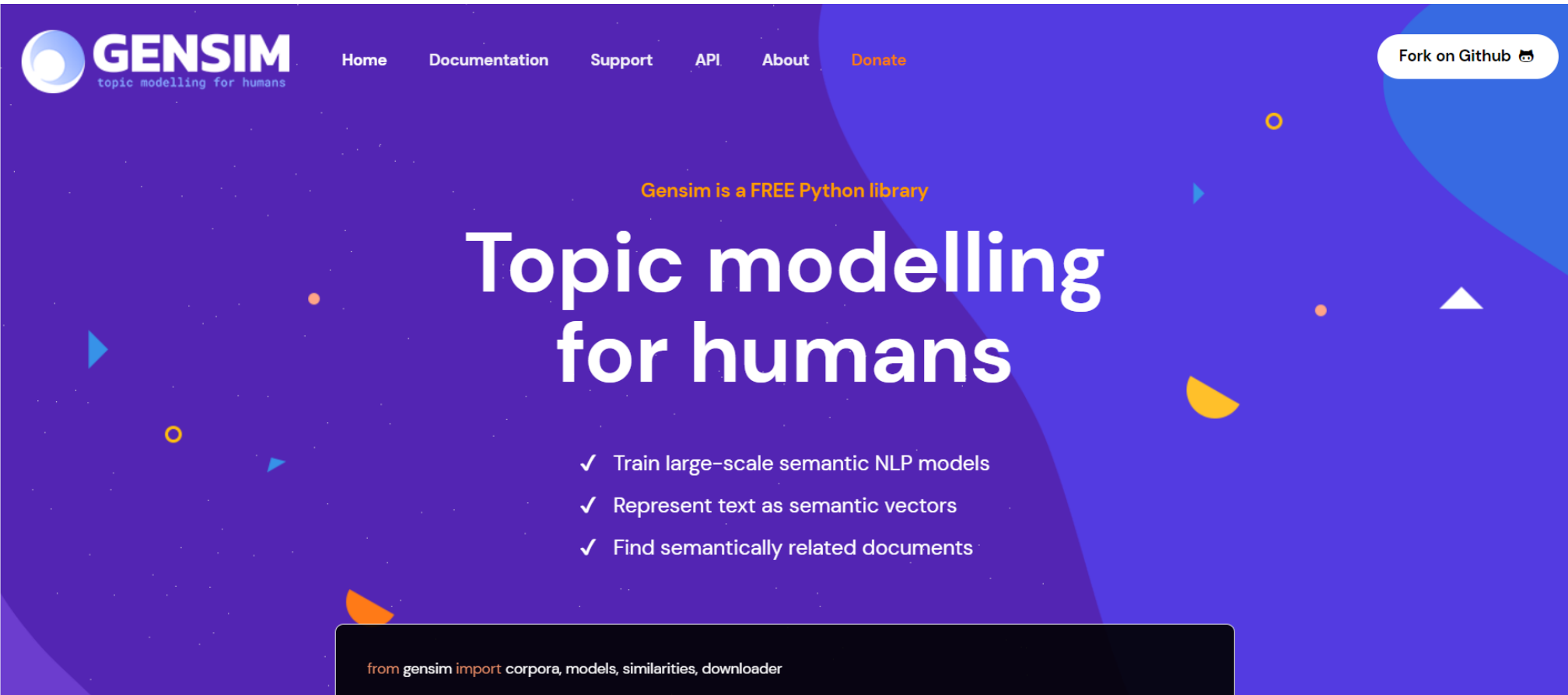
- LDA

- ① 단순 Count 기반 Document-Term 행렬을 생성 : 주어진 단어들의 빈도수에 기반하므로 Tf-idf 방법이 아닌 Count에 기반
- ② 토픽의 개수를 사전에 설정
- ③ 각 단어들을 임의의 토픽으로 최초 할당한 후 문서별 토픽 분포와 토픽별 단어 분포가 결정
- ④ 특정 단어를 하나 추출하고 추출한 해당 단어를 제외하고 문서의 토픽 분포와 토픽별 단어 분포를 다시 계산(Gibbs Sampling)
- ⑤ 추출된 단어는 새롭게 토픽 할당 분포를 계산
- ⑥ 다른 단어를 추출하고 4번 단계를 다시 수행, 계속해서 다른 단어를 추출하고 모든 단어들이 재계산되도록 반복
- ⑦ 지정된 반복 횟수(하이퍼파라미터로 지정)만큼 4~6번 단계를 수행하면서 모든 단어들의 토픽 할당 분포가 변경되지 않고 수렴할 때까지 수행

2. Gensim 라이브러리

➤ Gensim

- 텍스트 분석에 널리 사용되는 라이브러리

The image shows the homepage of the Gensim library. The background is a vibrant purple with abstract geometric shapes in blue, orange, and white. In the top left corner is the Gensim logo, which consists of a blue circle with a white dot inside, followed by the text "GENSIM" in bold white letters and "topic modelling for humans" in smaller white letters below it. To the right of the logo is a navigation bar with links: "Home", "Documentation", "Support", "API", "About", and "Donate" (in orange). In the top right corner, there is a white button with the text "Fork on Github" and a GitHub logo. The main content area features the text "Gensim is a FREE Python library" in orange, followed by "Topic modelling for humans" in large white letters. Below this, there is a list of three features, each preceded by a white checkmark: "Train large-scale semantic NLP models", "Represent text as semantic vectors", and "Find semantically related documents". At the bottom, there is a dark blue box containing the Python code snippet:

```
from gensim import corpora, models, similarities, downloader
```



2. Gensim 라이브러리

➤ Gensim

- 특징
 - Practicality: 실제 현실 텍스트 분석에 잘 활용될 수 있도록 디자인
 - Memory independence : 데이터 스트리밍을 이용하여, 대용량/웹스케일 Corpora를 처리, 한번에 대량의 Corpora를 메모리에 적재하지 않아서 가능
 - Performance : Vector Space algorithm을 C 등으로 구현하여 성능 향상
- 지원 모형
 - LDA
 - Word2Vec
 - Doc2Vec
 - Ensemble LDA
 - fastText model

Q&A