Topic Model: Latent Dirichlet Allocation

Jae Jin Yoo

Department of Statistics, Sunkyunkwan University

2019

INDEX

TOPIC MODEL

Topic Model

- Task of identifying Topics
  that best describe a set of documents.
- Topics will only emerge during the topic modeling process.
- One method of Topic Modeling $\Rightarrow$ LDA

TOPIC MODEL

Latent Dirichlet Allocation

- Each topic represents a set of words.
- Goal of LDA :
  To map all the Documents to Topics.
  To capture words in each documents by Topics.
- So the IDEA behind LDA is that...

Each document can be described by a distribution of topics

&

Each topic can be described by a distribution of words.

Introduction

Assumption

- Set of 1000 words, 1000 document.
  Document have Average 500 words each.
- How can you understand what category each document belongs to?
- Connect each document to each word by a thread based on their appearance in the document.

Introduction

Connecting Thread

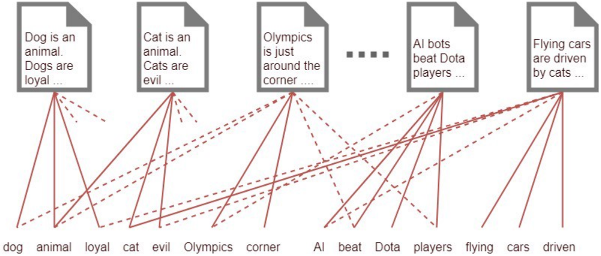

Figure 1: Model

- Categorization seems pretty fine.
- But too much computation. 500×1000 thread are needed. Too expensive, so how to avoid?

INTRODUCTION

Introduce Latent Layer

- Assume 10 topics/themes throughout documents. This topic can not be observed, since latent.
- So connect words to the topics depending on how well that word fall in that topic.
- Then, connect the topics to the documents based on what topics each documents touch upon.

Introduction

Connecting Thread

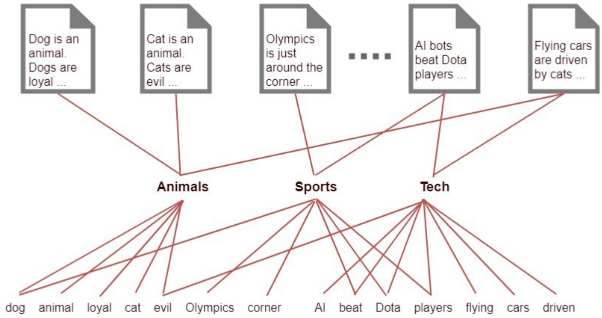

Figure 2: Model

- Assume 5 topics each relating to 500 words.
  Then number of threads needed is $1000 \times 5 + 10 \times 500 =$ 10,000.

Introduction

Connecting Thread

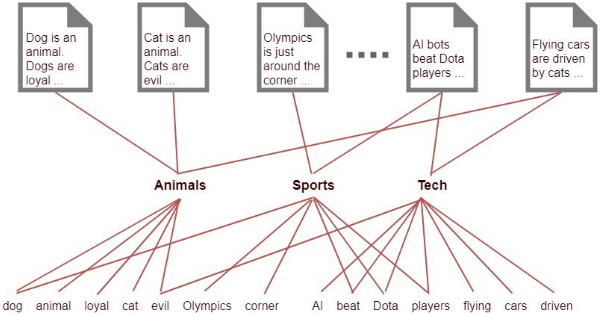

Figure 3: Model

- Assume 5 topics each relating to 500 words.
  Then number of threads needed is $1000 \times 5 + 10 \times 500 =$ 10,000.

QUESTION WRITING

Assumption

- Introduction은 컴퓨팅 관점이 더 강함.
- 각 단어는 한 topic에만 exhaustive하게 들어가나?
- 일단은 아닌 듯

INTRODUCTION

How LDA imagine document generation process

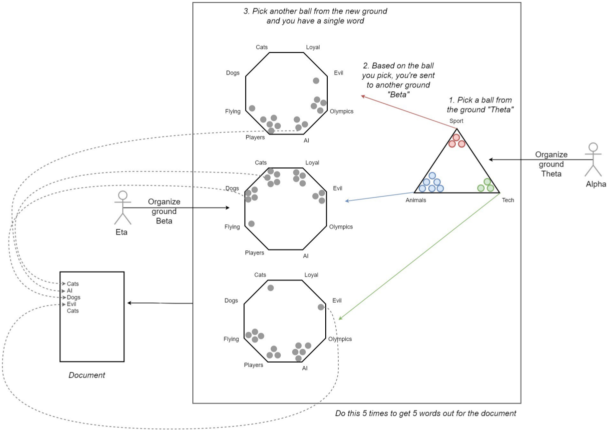

Figure 4: LDA model

Introduction

Simplification

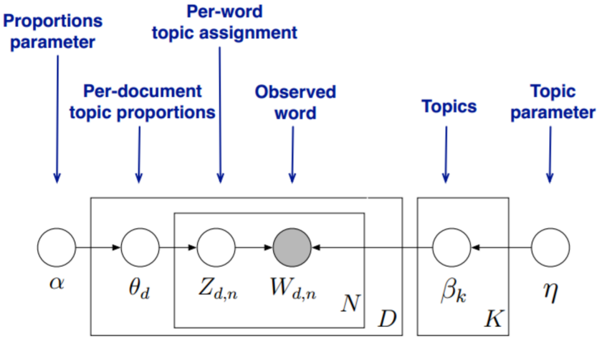

Figure 5: LDA model

PROBLEM

k : number of topic
W : word given topic

$$\boldsymbol{W} = \begin{bmatrix} w_{11} & \cdots & w_{1n} \\ \vdots & \ddots & \vdots \\ w_{k1} & \cdots & w_{kn} \end{bmatrix} \tag{1}$$

Z : topic given document

$$\boldsymbol{Z} = \begin{bmatrix} z_{11} & \cdots & z_{1k} \\ \vdots & \ddots & \vdots \\ z_{m1} & \cdots & z_{mk} \end{bmatrix} \tag{2}$$

PROBLEM

$$\begin{aligned} p(z_i|w_{dn}) &\propto p(z_i, w_{dn}) \\ &= p(w|z_i) \times p(z_{i|d}), i = 1, \ldots, k \end{aligned} \tag{3}$$

- 코드 상 topic을 고정
- 단어 하나와 해당 단어 보유하는 도큐먼트수로 임베딩이 생성됨

$$p(w|z_i) \times p(z_i|d) \text{ where i is fixed} \tag{4}$$

REFERENCE

- Blei, David M., and Johh D. Laerty. "Topic models" *Text Mining*. Chapman and Hall/CRC, 2009. 101-124.
- 김인영. "다의어 분석을 위한 군집화 방법." Master Thesis (2018)

Reference Figure

- **Figure 1** : ”word2vec Parameter Learning Explained.”
- **Figure 2** : ”Distributed representations of words and phrases and their compositionality.”
- **Figure 3** : ”Efficient Non-parametric Estimation of Multiple Embeddings perWord in Vector Space.”
- **Figure 4** : Python Result of Simulation
- **Figure 5** : R Result of Simulation