

Dimension Reduction

Feature Selection

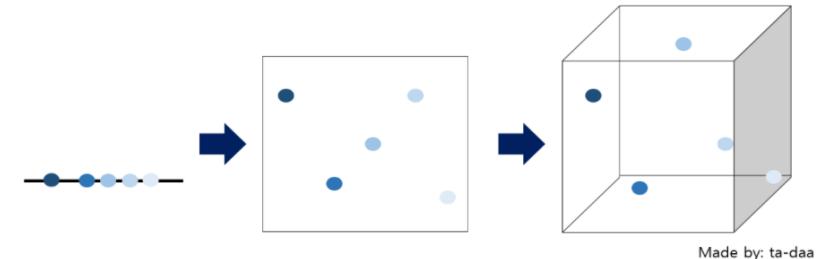
Feature Extraction ; LSA



Curse of Dimensionality

■ Sparse vs Dense

- Context을 생각하지 않은 “차” …> symbol, categorical value, count
→ one hot encoding, corpus에 나온 모든 단어 대비 특정 단어를 표현해야 되니 희귀할 수 밖에
- Context을 생각 했을 때 “차” …> tea, car 등 → Dense가 유리
- 문제는 **HOW do you make dense vector?**
- feature가 많을 수록 좋다???
→ 아니다. 상황에 따른 적절한 feature 수
→ 차원이 높아 질수록 sparseness 증가
- 좋은 feature 유지하거나 새로 만들어서 적은 차원으로 표현하면 modeling fitting시 유리



■ 대부분의 데이터의 차원은? (in NLP)

- 대부분 우리가 접하는 문서는 매우 많은 words로 구성
: 통계적인 측면에서 최소한 sample 수 > feature 수
- 그 중에 일부만 중요한 정보를 담고 있다(???)

Dimensionality Reduction

■ 필요성

- 계산 효율성(computationally efficient)
- Text Mining의 질적 향상

■ Feature Selection

- **부분 집합**을 선택 하는 것
- Filter : 기준을 만들고 그 기준에 만족하는 Feature를 선택
- Wrapper : Learning Algorithm을 사용, 학습을 통해서 Feature 선택, Feedback Loop

■ Feature Extraction

- 특징을 잘 보존 하여 새로운 Feature를 만드는 것
- Max Variance : PCA
- Max Distance : MDS(다차원 척도법)
- Reveal Latent Structure : LSA

■ 특징

- **비지도 학습이며, 특정 알고리즘과 무관**

Singular Value Decomposition(SVD), LSA(잠재적 의미 분석)

■ Full SVD

- $A = U * \Sigma * V^T$
- U : $m * m$ 직교행렬
- Σ : $m * n$ 직사각 대각행렬
- V^T : $n * n$ 직교 행렬

$$A_{m \times n} = U_{m \times m} \times \Sigma_{m \times n} \times V^T_{n \times n}$$

The diagram illustrates the Full SVD decomposition. On the left, a grey rectangular matrix labeled A is shown with dimensions $m \times n$. To its right is an equals sign. Next is a blue rectangular matrix labeled U with dimensions $m \times m$. To its right is a green rectangular matrix labeled Σ with dimensions $m \times n$, containing several green zeros and a single green square at the top-left. To the right of Σ is another green rectangular matrix labeled V^T with dimensions $n \times n$.

■ Truncated SVD

- 원래 행렬을 그대로 복원은 불가
- NLP 분야에서는 때로는 Noise 제거로 사용
- U : m 은 유지, n 을 축소
- Σ : **hyper-parameter**
- V^T : m 을 축소, n 은 유지

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

The diagram illustrates Truncated SVD. On the left, a white rectangular matrix labeled A' is shown. To its right is an equals sign. Next is a white rectangular matrix labeled U_t with a dashed border. To its right is a white rectangular matrix labeled Σ_t with a dashed border, containing two blue boxes labeled σ_1 and σ_t . To the right of Σ_t is another white rectangular matrix labeled V_t^T with a dashed border.

■ 실습

- 간단한 SVD 개념 이해
- Topic Modeling