

Lecture 2 preview

Word2Vec review

- word2vec의 벡터공간에 벡터의 덧셈과 뺄셈을 적용할 수 있음
→ ex) king - man + woman = queen
- 2d projection에 의해 누락된 정보들이 존재하기에 overtrust해서는 안된다

Optimization :

1. Gradient Descent

- cost function을 최소화하기 위한 알고리즘
- 아이디어 : negative gradient의 방향으로 조금씩 세타값을 이동시킨다
- 문제점 : J는 모든 window에 대한 함수이기 때문에, 처리할 단어의 수가 많아지면 계산량이 많아져서 속도가 매우 느리다

2. Stochastic Gradient Descent

- 랜덤한 샘플을 하나 뽑아 그라디언트를 계산하여 세타값을 업데이트한다
- 오차가 크지만, 학습속도가 Gradient Descent Algorithm에 비해 빠르다
- **mini-batch**
 - 전체 학습 데이터를 32나 64의 크기로 나누어 계산을 수행
 - 그라디언트의 오차가 작아진다
 - parallelization에 유용하다(32나 64 크기의 미니배치를 이용할 경우)

Word2Vec : More details

1. Skip-grams : center word에 대해 모든 context word를 예측한다
2. Continuous Bag of Words : context word를 이용하여 center word를 예측한다

GloVe Model

window vs. full document co-occurrence matrix

full document : 한 문서를 기준으로 단어가 등장하는 횟수로 벡터를 구성

- Latent Semantic Analysis - 잠재적 의미분석

window : 한 문장을 기준으로 윈도우에 단어가 등장하는 횟수로 벡터를 구성

- 벡터의 유사성 → 단어의 유사성 유추 가능
- 문제점 :
 - increase in size with vocab.
 - high dimensional - requires a lot of storage
 - sparsity issues
- solution : low dimensional vectors

How to evaluate word vectors? : word embedding evaluation

- intrinsic : word embedding만 평가
 - evaluation on a specific subtask
 - fast to compute
 - helps to understand that system
 - unclear if really helpful unless correlation to real task is established
- extrinsic : word embedding 활용하여 시스템의 성능을 평가
 - evaluation on a real task
 - can take a long time to compute accuracy
 - unclear if the subsystem is the problem or its interaction or other subsystems

출현 빈도 수 높은 단어들은 다양한 의미를 갖는다

오래된 단어들은 다양한 의미를 갖는다

→ cluster the contexts

