Lecture2 - Neural Classifiers

01. Intro
    Review : Main idea of word2vec
02. Word2Vec2
    word2vec parameters
    Optimization : Stochastic Gradient Descent
    Word2Vec algorithm family : more details
03. GLOVE
    Toward GloVe : Cound based vs. direct prediction
    word vector 평가 방법
04. Word senses and word sense ambiguity
    Word sense ambiguity
    Solution1) **Improving Word Representations Via Global Context And Multiple Word Prototypes (Huang et al. 2012)**
    Solution2) **Linear Algebraic Structure of Word Senses, with Applications to Polysemy**

01. Intro

Review : Main idea of word2vec

- random word vector로 시작
- iterate through each word in the whole corpus
- 단어에 대한 단어 벡터, 문맥 벡터 사이의 내적 관점에서 아래 확률 분포 식을 수행

$$: P(o|c) = \frac{\exp(u_o^T v_c)}{\sum_{w \in V} \exp(u_w^T v_c)}$$

- Learning

- 주변 단어 예측을 더욱 잘하도록 벡터 업데이트
- 이를 통해 벡터는 고차원에서 유사성이 높은 단어들과 더 가까워지고 의미있는 방향을 갖게 됨

02. Word2Vec2

word2vec2는 아래와 같은 word2vec의 2가지 단점을 보완하여 나옴

1. negative sampling
2. embedding

word2vec parameters

- word2vec의 유일한 parameter는 단어벡터
- ‘Bags of words’ model
  - 단어의 순서와 위치 고려 X
  - 즉, 문맥 안에 있는 단어들은 모두 같은 취급
- 각 단어에 대해 외부단어(outside) vector, 중앙단어(center) vector 존재

- 계산 과정

  1) 특정 외부 단어가 중심단어와 함께 발생할 가능성에 대한 점수 얻기 위해 target 벡터와 context 벡터를 내적

  2) 이 정수를 확률로 변환하기 위해 softmax 사용

Optimization : Stochastic Gradient Descent

- Gradient Descent : 모든 벡터를 한번에 업데이트
  - 느리다
- Stochastic Gradient Descent : corpus를 batch 단위로 쪼개서 업데이트

- 빠드라
- 많이 사용한다

- Optimization 과정

  1) 무작위(0에 가까운 숫자들)로 단어벡터의 가중치 초기화 (random initial value)

  2) 손실함수 J(θ)로 loss 계산 → loss가 낮은 곳으로 기울기 하강

- step size(learning step) : 작은 걸음의 보폭
  - size가 너무 크면 왔다갔다 반복함
  - size가 너무 작으면 학습 시간 오래 걸림
  - 적절한 size로 조절하는 것이 중요
- 최적화 과정을 통해 loss가 낮은 좋은 단어 벡터를 만드는 것이 목표

Word2Vec algorithm family : more details

- 왜 2개의 벡터만 보는가?
  - 2개 이상의 벡터로 내적을 수행하면 계산이 복잡해지므로 오히려 더 비효율적
  - 특히 분모를 계산할 때 cost가 높다
- word2vec 모델

  1) skip-grams (SG) : target 단어로 context 단어 예측

  2) Continuous Bag of Words (CBOW) : context 단어로 target 단어 예측

03. GLOVE

Toward GloVe : Cound based vs. direct prediction

- count based
  - 학습이 빠르다

    - 통계를 효율적으로 사용
    - 단어 유사성을 포착하는데 주로 사용
    - 주어진 corpus에 대한 단어 개수 불균형
    - ex) LSA, HAL, COALS, Hellinger-PCA
- direct prediction
    - corpus 크기 Scales
    - 통계를 비효율적으로 사용
    - 다른 작업에서 향상된 성능 생성
    - 단어의 유사성을 넘은 복잡한 패턴 포착 가능
    - ex) Skip-gram/CBOW, NNLM, HLBL, RNN

word vector 평가 방법

크게 intrinsic(내적), extrinsic(외적)의 방법으로 단어 벡터들이 잘 설정되었는지 평가

1. Intrinsic 내적
    a. 특정/중간 하위 작업에서의 평가
    b. 계산 속도 향상
    c. 시스템 이해하는데 도움
    d. 실제 작업에 대한 상관관계 설정되지 않으면 실제로 도움이 되는지 명확하지 않음
    e. 평가방법) 단어벡터를 추가한 후 코사인 거리가 직관적인 의미론적 및 구문적 유사 질문을 얼마나 잘 포착하는지 평가
    f. 만일 정보가 있지만 선형적이지 않으면 문제

2. Extrinsic 외적
    a. 실제 작업에 대한 평가
    b. 정확성 계산 시간이 오래 걸릴 수 있다

c. 하위 시스템 문제인지, 상호 작용인지, 또는 다른 하위 시스템인지 불분명

d. 정확히 하나의 서브 시스템을 다른 서브시스템으로 교체하면 정확도 향상됨

04. Word senses and word sense ambiguity

Word sense ambiguity

단어 의미의 모호성 문제 : 한 단어가 여러개의 의미를 가질 수 있는 문제

흔한 단어에서 많이 나타나며, 존재한지 오래된 단어에서 많이 발생

같은 단어라도 각 의미마다 다른 벡터를 갖는 것이 목표

Solution1) Improving Word Representations Via Global Context And Multiple Word Prototypes (Huang et al. 2012)

- 단어를 학습시키고 군집마다 다르게 위치한 같은 벡터는 따로 벡터를 부여하고 따로 학습
- 다른 클러스터에 있는 같은 단어는 다른 단어로 취급
- 하지만 corpus마다 편차가 크게 학습된다는 단점

Solution2) Linear Algebraic Structure of Word Senses, with Applications to Polysemy

- 다른 단어는 따로 벡터를 만들어 학습시키고 이에 대한 평균을 구함
- 하지만 평균값은 더욱 더 모호성을 갖고 올 수 있다는 단점
- 그럼에도 좋은 결과를 갖기도 함