GloVe: Global Vectors for Word Representation

김호재

2024/03/19

TRAIN AND TEST

논문 목표

- 단어 의미에 대한 선형적 관계를 파악하는 모델의 특징 이해

- global log-bilinear regression 모델 소개

- 모델 학습 방법 이해: global log-bilinear regression 모델은 특정한 가중치를 사용하는 최소 제곱법을 통해 학습

TRAIN AND TEST

Relationship to Other Models

- 단어벡터를 학습하기 위한 비지도방법은 궁극적으로 co-occurrence statistics of the corpus에 기반하기 때문에, 모델들간 공통점이 있다.
- 이에 대해 논문은 ivLBL과 skip-gram같은 window-based 방법의 모델이 위에서 제안된 모델과 어떻게 연관되어있는지 분석함

Relationship to Other Models

- skip-gram과 ivLBL의 시작점은 단어 i의 context에서 단어 j가 나타날 확률에 대한 모델 Qij(softmax함수)

$$Q_{ij} = \frac{\exp(w_i^T \tilde{w}_j)}{\sum_{k=1}^{V} \exp(w_i^T \tilde{w}_k)}$$

- 두가지 모델에서, context window에 대한 로그 확률을 최대화하려는 의도는 논문에서의 목적과 같다.따라서, global 목적 함수는 다음과 같이 정의할 수 있다.

$$J = - \sum_{\substack{i \in \text{corpus} \\ j \in \text{context}(i)}} \log Q_{ij} \, .$$

TRAIN AND TEST

Relationship to Other Models

- 이 합산연산에서 각 항에 대해 소프트맥스 정규화 계수를 얻는 것은 비용이 많이 든다.
- 효율적으로 훈련하기 위해 skip-gram과 ivLBL 모델은 Qij를 근사한다.
- 여기서, 다음식으로 변환하면 더 빨라진다. term의 수가 동시발생행렬에서 얻어진다는 것을 이용했다.

$$J = -\sum_{\substack{i \in \text{corpus} \\ j \in \text{context}(i)}} \log Q_{ij} \,. \quad \longrightarrow \quad J = -\sum_{i=1}^{V}\sum_{j=1}^{V} X_{ij} \log Q_{ij}$$

TRAIN AND TEST

Relationship to Other Models

- 크로스 엔트로피 오차(Cross-entropy error)의 문제점: 확률 분포에서 거리를 계산하는 방법 중 하나인 크로스 엔트로피 오차는 꼬리가 긴 확률 분포의 경우 잘못 모델링되어서 자주 발생하지 않는 사건에 대해 매우 큰 가중치를 부여할 수 있습니다.

$$J = -\sum_{i=1}^{V} X_i \sum_{j=1}^{V} P_{ij} \log Q_{ij} = \sum_{i=1}^{V} X_i H(P_i, Q_i)$$

- 측정법의 한계와 계산 병목 현상: 모델 분포 Q가 적절하게 정규화되지 않으면 측정법에 한계가 있을 수 있으며, 이로 인해 전체 어휘 사전에 대한 sum 연산으로 인한 계산 병목 현상이 발생할 수 있습니다.
- 따라서 **최소제곱법**의 선택: 크로스 엔트로피 오차와 측정법의 한계를 고려하여 최소제곱법을 선택합니다. 이는 P와 Q의 정규화 계수를 무시할 수 있으며, 데이터를 필터링하여 성능을 높일 수 있다는 Mikolov의 발견을 토대로 합니다.
- 로그를 취한 제곱 오차의 활용: 큰 값의 오차가 최적화를 방해하는 경우가 발생할 수 있으므로 로그를 취한 제곱 오차를 사용하여 이를 해결합니다.
- 미리 결정된 가중치 인자의 문제: 최적화가 보장되지 않는 미리 결정된 가중치 인자를 필터링하여 보다 더 일반적인 가중치 함수를 제시합니다.

$$\hat{J} = \sum_{i,j} f(X_{ij}) (w_i^T \tilde{w}_j - \log X_{ij})^2$$

TRAIN AND TEST

Complexity of the model

- 말뭉치의 개수는 행렬의 전체 크기보다 작거나 같을 것이므로, 모델의 복잡도는 **O(n^2)**보다 나쁘지 않을 것으로 예상.
- 그러나 일반적으로 수십만 개의 단어들이 있기 때문에 모델의 크기는 수천억까지 커질 수 있음. 이렇게 되면 말뭉치보다 훨씬 큰 크기가 되므로, 이러한 문제를 고려해야 함.
- 즉, 모델의 복잡도는 행렬의 크기와 0이 아닌 원소들의 개수에 의해 결정되며, 이러한 요소들을 고려하여 모델의 크기를 적절히 조절해야 함.

TRAIN AND TEST

Complexity of the model

- 동시발생 횟수의 멱함수: 단어 i와 j의 동시발생 횟수는 해당 단어쌍의 frequency rank의 멱함수로 나타낼 수 있습니다
- 전체 단어 수와 동시발생 행렬의 관계: 말뭉치 내의 전체 단어 수는 동시발생 행렬 X의 원소의 총 합에 비례합니다
- 단어 rank의 최댓값과 동시발생 행렬의 크기: 식에서 r의 최대값은 |V|와 같습니다.
- 결국 **모델의 복잡도 개선을 위한 연구**: 실험 결과에서는 모델의 복잡도가 가장 최악의 경우에도 훨씬 개선되었음을 확인했습니다. 이에 따라 on-line window-based 방법에 비해 더 나은 결과를 보여주었습니다.

$$|X| = \begin{cases} O(|C|) & \text{if } \alpha < 1 \\ O(|C|^{1/\alpha}) & \text{if } \alpha > 1 \end{cases}$$

TRAIN AND TEST

Complexity of the model

- 이 실험에서는 α=1.25일때, |X|=O(|C|0.8)이 되는데 이때 모델의 복잡도가 가장 최악의 경우 였던 O(V2) 보다 훨씬 개선되었다는 것을 관찰했다. 또한, O(|C|) 였던 on-line window-based 방법에 비해서도 어느정도 더 나은 결과를 보여주었다.

$$|X| = \begin{cases} O(|C|) & \text{if } \alpha < 1 \\ O(|C|^{1/\alpha}) & \text{if } \alpha > 1 \end{cases}$$

TRAIN AND TEST

Evaluation methods

- 단어 유추 작업 (Word Analogy Task): "a와 b의 관계는 c와 ?와의 관계이다."와 같은 유추 문제를 푸는 작업이 진행.

- 단어 유사도 작업 (Word Similarity Task): WordSim-353, MC, RG, SCWS, RW 등의 데이터셋을 사용하여 단어 간 유사도를 평가하는 작업이 이루어짐

- 개체명 인식 작업 (Named Entity Recognition, NER): CoNLL-2003 데이터셋을 사용하여 사람, 장소, 조직, 기타 등의 객체를 인식하는 작업이 수행됨. 이를 위해 CRF 모델을 사용하여 훈련 및 평가되었음.

Model	Dim.	Size	Sem.	Syn.	Tot.
ivLBL	100	1.5B	55.9	50.1	53.2
HPCA	100	1.6B	4.2	16.4	10.8
GloVe	100	1.6B	67.5	54.3	60.3
SG	300	1B	61	61	61
CBOW	300	1.6B	16.1	52.6	36.1
vLBL	300	1.5B	54.2	64.8	60.0
ivLBL	300	1.5B	65.2	63.0	64.0
GloVe	300	1.6B	80.8	61.5	70.3
SVD	300	6B	6.3	8.1	7.3
SVD-S	300	6B	36.7	46.6	42.1
SVD-L	300	6B	56.6	63.0	60.1
CBOW†	300	6B	63.6	67.4	65.7
SG†	300	6B	73.0	66.0	69.1
GloVe	300	6B	77.4	67.0	71.7
CBOW	1000	6B	57.3	68.9	63.7
SG	1000	6B	66.1	65.1	65.6
SVD-L	300	42B	38.4	58.2	49.2
GloVe	300	42B	**81.9**	**69.3**	**75.0**

TRAIN AND TEST

학습 관련 detail

- 주어진 말뭉치 5개를 이용하여 모델을 학습. 말뭉치는 Stanford tokenizer를 사용하여 토큰화되고 소문자로 변환. 이 중에서 가장 자주 등장한 400,000개의 단어로 voabulary를 생성하고, 동시발생횟수 행렬 X를 만듬.
- X 생성 시에는 context window의 크기와 왼쪽, 오른쪽 context를 구분할지를 결정. 거리가 먼 단어쌍은 관련성이 낮은 정보를 담고 있을 것으로 예상되므로 감소하는 가중치 함수를 사용하여 단어 간의 관련성을 조절. 모든 실험에서 초기 학습률을 0.05로 하고 AdaGrad를 사용하여 모델을 훈련.
- 모델은 두 개의 단어벡터를 생성했고, 네트워크를 여러 인스턴스를 훈련한 뒤 결과를 결합하여 과적합과 잡음을 줄였음. word2vec와 SVD 베이스라인 모델로 학습된 결과를 비교. SVD 베이스라인에는 SVD-S와 SVD-L 두 가지 방법을 사용하였고, 각각의 방법은 X값의 범위를 압축시킴.

Conclusion

- 당시에 연구는 ' count-based ' 방법과 ' prediction-based ' 방법 사이에서 이뤄지고 있습니다. ' count-based ' 방법은 단어가 얼마나 자주 함께 등장하는지를 세어서 단어를 표현하고, ' prediction-based ' 방법은 단어가 주변 단어를 얼마나 잘 예측하는지를 기반으로 단어를 표현합니다. 당시에는 'prediction-based' 모델이 더 많은 지지를 받고 있는데, 다양한 작업에서 더 좋은 성능을 보인다고 주장했다.
- 그러나 해당 논문에서는 두 방법이 근본적으로 크게 다르지 않다고 주장합니다. 왜냐하면 두 방법 모두 말뭉치 내에서 단어들 간의 기본적인 관계를 조사하기 때문입니다. 하지만 'count-based' 방법은 전체적인 통계를 더 효율적으로 포착할 수 있다는 장점이 있다.
- 이 논문에서는 이러한 'count-based' 방법의 장점을 활용하면서도 최근에 많이 사용되는 'prediction-based' 방법에서 **주로 나타나는 의미 있는 구조**를 동시에 포착하는 모델을 만들었습니다. 이 모델은 GloVe라고 불리며, 다른 모델보다 단어 유추, 단어 유사도 및 개체명 인식과 같은 작업에서 더 우수한 성능을 보였다.

TRAIN AND TEST

TRAIN AND TEST