지식 임베딩 심층학습을 이용한 단어 의미 중의성 해소

오동석°[1], 양기수[2], 김규경[2], 황태선[2], 임희석[2]
Human-inspired 복합지능연구센터[1]
고려대학교 컴퓨터학과[2]
{inow3555,willow4,overmind22,hts920928,limhseok}@korea.ac.kr

Word Sense Disambiguation Using Knowledge Embedding

Dongsuk Oh°[1], Kisu Yang[2], Kuekyeng Kim[2], Taesun Whang[2], Heuiseok Lim[2]
Human-inspired AI & Computing Research Center[1]
Department of Computer Science and Engineering, Korea University[2]

요 약

단어 중의성 해소 방법은 지식 정보를 활용하여 문제를 해결하는 지식 기반 방법과 각종 기계학습 모델을 이용하여 문제를 해결하는 지도학습 방법이 있다. 지도학습 방법은 높은 성능을 보이지만 대량의 정제된 학습 데이터가 필요하다. 반대로 지식 기반 방법은 대량의 정제된 학습데이터는 필요없지만 높은 성능을 기대할수 없다. 최근에는 이러한 문제를 보완하기 위해 지식내에 있는 정보와 정제된 학습데이터를 기계학습 모델에 학습하여 단어 중의성 해소 방법을 해결하고 있다. 가장 많이 활용하고 있는 지식 정보는 상위어(Hypernym)와 하위어(Hyponym), 동의어(Synonym)가 가지는 의미설명(Gloss)정보이다. 이 정보의 표상을 기존의 문장의 표상과 같이 활용하여 중의성 단어가 가지는 의미를 파악한다. 하지만 정확한 문장의 표상을 얻기 위해서는 단어의 표상을 잘 만들어줘야 하는데 기존의 방법론들은 모두 문장내의 문맥정보만을 파악하여 표현하였기 때문에 정확한 의미를 반영하는데 한계가 있었다. 본 논문에서는 의미정보와 문맥정보를 담은 단어의 표상정보를 만들기 위해 구문정보, 의미관계 그래프정보를 GCN(Graph Convolutional Network)를 활용하여 임베딩을 표현하였고, 기존의 모델에 반영하여 문맥정보만을 활용한 단어 표상보다 높은 성능을 보였다.

주제어: 단어 중의성 해소, Graph Convolution Network, Word Embedding, WordNet

1. 서론

자연어처리에서 두 개 이상의 의미를 가진 단어를 문장의 쓰임에 따라 정확하게 분석하는 것을 단어 중의성 해소라고 한다. 사람은 의사 소통을 하기 위해 많은 경험을 토대로 쌓인 지식정보들을 활용하듯이 기계도 역시 그러한 과정을 통해 문장을 이해시켜야한다. 단어 중의성 해소 연구는 이러한 과정을 반영하여 다양한 연구가 진행되었고 크게 두 가지 방법이 있다.

첫번째는 문장에 등장한 단어들을 사전에 정의된 어휘 지식을 활용하여 예측하는 지식 기반 방법이다. 지식 기반 방법은 사전 정의 기반 방법과[1] 그래프 기반 방법이[2, 3, 4] 있다. 사전 정의 기반 방법은 사전에 정의된 단어의 설명을 기반으로 의미를 추론하는 방법이고, 그래프 기반 방법은 단어의 시소러스 정보를 활용하여 의미 관계를 가지는 단어들의 관계성을 보고 의미를 추론하는 방법이다.

두번째는 문장 내 단어의 의미가 레이블된 데이터를 이용하여 기계학습 모델에 학습하고 단어의 의미를 예측하는 지도학습 방법이다.[5, 6, 7, 8, 9] 지도학습 방법은 기계학습을 이용하기 때문에 높은 성능을 보이지만, 대량의 학습데이터를 구축해야하는 단점이 있다. 이러한 단점 외에도 하나의 의미도 다양한 문맥 패턴을 가지기 때문에 학습데이터를 구축하기 위해서는 많은 시간이 필요하다. 사람은 단어의 의미를 파악하기 위해서 위의 두가지 방법을 상호 보완할수 있지만 기계는 그렇지 않다. 하지만 딥러닝 모델 연구가 활발하게 이루어 지면서 상호 보완을 하는 연구가 진행되고 있다. 딥러닝은 다양한 작업들을 해결하기 위해 여러 모델들이 개발되었는데 이 모델들이 서로 부족한 부분을 보완할수 있도록 설계할수 있다는 장점이 있다. Luo et al.[9]논문에서는 WordNet의[10] 의미설명(Gloss)정보를 활용하여 데이터 부족 문제를 보완하였다. 중의성 단어가 가지는 상위어(Hypernym)와 하위어(Hyponym), 동의어(Synonym)가 가지는 의미설명(Gloss)정보를 표상으로 만들어내고, 학습데이터의 문장정보를 표상으로 만들어낸다. 이 두개의 표상을 활용하여 의미적 관계를 파악하고, 중의성 단어의 문제를 해결한다. 이는 단어의 의미를 파악하는데 있어 어느 한쪽이 부족한 표상을 담고 있더라도 서로 상호 보완이 가능하게 한다.

서로의 정보를 상호 보완하더라도 문장의 표상을 정확하게 나타내는것은 굉장히 중요하다. 정확한 표상을 나타내기 위해서는 다양한 문맥 패턴정보가 필요한데 부족한 학습데이터에서는 여전히 이러한 문제를 남기고 있다. 이러한 문제를 해결하기 위해서 여러 문장을 수집하여 비지도 학습 방법으로 사전 학습을 통해 단어 표상을 얻는 연구가 많이 진행되어 왔다.[11, 12, 13, 14] 하지만, 이러한 연구들도 문장 내에 나타난 문맥 패턴만을 고려하기 때문에 단어간의 구문 관계나 단어의 의미 관계를 반영하지 못한다. 본 논문에서는 단어의 표상에 구문 정보와 의미 관계를 가질수 있도록 Vashishth et al.[14]

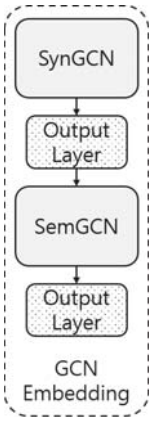

그림 1. 구문정보와 의미관계정보를 이용한 단어 임베딩

에서 제안한 그래프 임베딩을 활용하였다. Vashishth et al.[14]에서 제안한 단어 표상 방법은 구문 그래프와 시소러스 의미 관계 그래프를 GCN(Graph Convolutional Network)을 통해서 나타낸다. 이 표상을 활용한 모델은 이전 모델보다 높은 성능을 보였다.

2. 지식 임베딩 심층학습을 이용한 단어 중의성 해소

2.1 구문정보와 의미관계정보를 이용한 단어 임베딩

본 논문에서는 단어 표상에 구문 정보와 의미 관계 정보를 반영하기 위해 GCN(Graph Convolutional Network)를 활용하였다. 구문 정보를 반영하기 위해 Stanford CoreNLP parser에서 표현되는 의존 관계 정보를 활용하였고, 의미 관계 정보를 나타내기위해 WordNet 정보를 활용하였다. 구문, 의미 관계 그래프 정보는 $G = (V, E, X)$으로 정의되며, Node 세트 $|V| = n$이고 E는 Edge 세트를 나타낸다. $X \in \mathbb{R}^{n \times d}$는 d-차원 입력 Node 특징을 나타낸다. 레이블 l_{uv}가있는 Node u에서 v까지의 $E(Edge)$는 (u, v, l_{uv})로 표시된다. 정보가 항상 한방향으로만 전파될 필요는 없으므로 $E(Edge)$에 역방향 (u, v, l_{uv}^{-1})를 포함한다. $h_v^{k+1} \in \mathbb{R}^d$ k-GCN 층 이후의 Node v는 다음과 같이 주어진다.

$$h_v^{k+1} = f(\sum_{u \in N+(v)} (W_{l_{uv}}^k h_u^k + b_{l_{uv}}^k)) \quad (1)$$

여기서 $W_{l_{uv}}^k \in \mathbb{R}^{d \times d}$ 및 $b_{l_{uv}} \in \mathbb{R}^d$는 레이블 별 모델의 매개변수이고, $N_+(v) = N(v) \cup v$ (v 자체 포함)의 인접 세트이며 $h_v^k \in \mathbb{R}^d$는 $k-1$ 계층 이후 Node u의 표현이다. Edge Label Gating Mechanism은 실제 그래프에서 일부 Edge는 특정 Task와 관련이 없거나 오류를 제공할수 있다. 이것은 텍스트의 의존성 구문 분석과 같은 자동으로 구성된 그래프에서 나타난다. 이 문제를 해결하기 위해 Marcheggiani et al.[15]에서 제안하는 방법을 사용한다. 각 Node v에 대해 연결된 모든 Edge에 Score를 $g_{l_{uv}}^k \in \mathbb{R}$로 계산한다. Score는 아래와 같이 각 Layer에 대해 독립적으로 계산된다.

$$g_{l_{uv}}^k = \sigma(\hat{W}_{l_{uv}}^k h_u^k + \hat{b}_{l_{uv}}^k) \quad (2)$$

식(2)에서 $\hat{W}_{l_{uv}}^k \in \mathbb{R}^{1 \times d}$ 및 $\hat{b}_{l_{uv}}^k \in \mathbb{R}$은 학습 가능한 파라미터이고 $\sigma(\cdot)$는 sigmoid 함수이다. k번째 Layer에서 업데이트된 GCN(Graph Convolutional Network)전파 규칙은 식(3)과 표현할 수 있다.

$$h_v^{k+1} = f(\sum_{u \in N+(v)} g_{l_{uv}}^k \times (W_{l_{uv}}^k h_u^k + b_{l_{uv}}^k)) \quad (3)$$

이와 같이 GCN(Graph Convolutional Network) 표현 방법을 그림 1과 같이 Pipeline구조로 구현했다. 입력 문장의 단어들 간에 의존 관계 그래프를 SynGCN으로부터 단어를 표현하였고, 이 단어 벡터를 입력으로 WordNet으로부터 의미 관계 그래프 만든 후에 SemGCN으로 단어를 표현하였다.

2.2 단어 중의성 해소 모델

단어 중의성 해소는 Luo et al.[9]에서 제안한 방법을 사용하였으며, 그림 2와 같이 Context, Gloss, Memory, Scoring 4개의 모듈로 구성되어 있다. 모든 단어 벡터는 2.1에서 표현한 SemGCN 단어 표상 결과를 사용하였다. Context 모듈은 중의성 단어를 가지는 단어의 문장을 Bi-LSTM을 통해 순방향, 역방향으로부터 나온 벡터값을 concatenate하여 표현한다. Gloss 모듈은 중의성 단어의 의미설명(Gloss)정보를 같은 방법으로 Bi-LSTM을 통해 표현하였다. 본 논문에서는 Gloss Expansion 방법을 사용하였고, 동사와 명사품사를 가지는 상위어, 하위어의 모든 의미설명(Gloss)정보들도 Bi-LSTM을 통해 표현한다. 상위어, 하위어의 정보는 BFS(Breadth First Search)를 통해 깊이 K만큼 추출하여 관련된 Gloss정보를 Context 모듈과 같이 표현한다. 이와같이 표현된 Gloss 정보들을 Relation Fusion Layer을 통해 상위어는 순방향 LSTM에 나열하고, 하위어는 역방향 LSTM에 나열하여 벡터로 표현한 후 concatenate하여 표현한다. 메모리 모듈에서는 Context 모듈에서의 벡터 결과와 Gloss Expansion 모듈에서의 벡터 결과를 Attention을 통해 계산한다음 메모리를 업데이트 한다. Scoring 모듈에서는 Context 모듈에서의 벡터 결과와 Memory 모듈에서의 마지막 Attention 결과값을 사용하여 중의성 단어의 의미를 선택한다.

3. 실험 결과

실험에 사용한 데이터셋은 Senseval-2, Senseval-3 task 1, SemEval-07 task 17, SemEval-13 task 12, SemEval-15 task 13 에서 제공한 테스트 데이터셋을 사용하였고, 학습데이터는 SemCor3.0을 사용하였다. 이 데이터는 WordNet 3.0에서 제공하는 의미태그 기반으로 총 352문서에 225,036개의 의미가

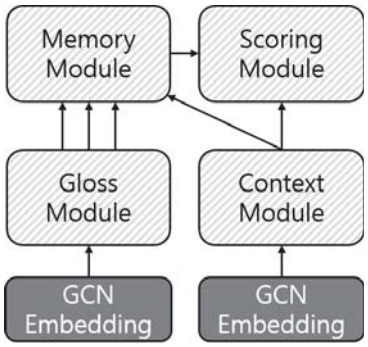

그림 2. 단어 중의성 해소 모델

레이블되어있다. 현재까지 단어 중의성 해소 문제에서 가장 많이 사용되고 있는 데이터이다. 표 1과 같이 비교분석은 총 5개의 모델과 진행하였다. MFS은 단어 중의성 해소 문제에서 Baseline으로 잡고 있는 모델로써 중의성 단어들이 가지는 의미들중 가장 많이 쓰는 의미를 선택하는 방법이다. Lesk$_{ext+emd}$모델은[16] Lesk알고리즘의[1] 변형으로써 Lesk 알고리즘이 가지고 있는 문제점을 보완한 모델이다. 의미적 공간에서 의미설명(Gloss) 문맥간에 단어 유사도를 계산하여 가장 많이 중첩된 의미를 선택하는 모델로 의미설명(Gloss)의 문맥정보가 많을수록 성능이 높아짐을 보여주었다. Babelfy는[3] BabelNet의[17] 의미망 정보를 활용하여 단어 중의성 문제를 해결한 그래프 기반 모델이다. IMS$_{emd}$은[6] 선형 SVM(Support Vector Machine)을 활용하여 특정 윈도우 내의 POS태그와 단어들을 기반으로 분류하는 IMS 모델을 확장한 모델이다. 단어 임베딩을 추가하여 이전 IMS보다 성능을 높였다. Bi-LSTM+att+LEX+POS은[7] Bi-LSTM+Attention을 이용한 다중 작업 모델로써 POS 태그 정보와 WordNet의 사전 편찬자 (lexicographer)을 추가적으로 구분하게 함으로써 기존 Bi-LSTM+Attention 모델보다 높은 성능을 보였다. GAS$_{ext}$ 모델은 상위어(Hypernym)와 하위어(Hyponym), 동의어(Synonym)가 가지는 의미설명(Gloss)정보의 표상과 학습데이터 문장에 대한 표상의 관계성을 메모리 모듈에서 파악함으로써 스코어 모델을 통해 중의성 단어에 대한 의미를 선택한다. 본 논문에서 제안하는 모델은 GAS$_{ext}$에 GCN(Graph Convolutional Network)을 이용한 단어 임베딩 결과를 추가한 모델이다. 단어 임베딩은 Vashishth et al.[14]논문에서 사용한 학습데이터와 파라미터를 사용하였고, 단어 중의성 해소 모델의 파라미터는 Luo et al.[9]과 동일하게 적용하였다. 결과적으로, 기존보다 0.2높은 성능을 보였다.

4. 결론

단어의 의미를 정확하게 분석하는 것은 자연어처리의 가장 중요한 부분 중 하나이다. 사람은 문장에서 단어들의 의미를 정확하게 파악하기 위해 문맥정보를 고려하는데 그때 문법적 정보와 사전 정보를 함께 사용한다. 이러한 방법을 반영하여 단어의 임베딩을 얻기위해 문법 정보와 의미 관계 정보를 활용하였고, 기존의 문맥정보만을 파악한 결과보다 높은 성능을 보였다. 하지만 단어 임베딩에 사용한 학습데이터의 제한적인 단어 수 때문에 의미 분석에 오류를 나타내었다. 또한 단어가 고정된 벡터값으로만 표현되기 때문에 단어를 표현하는데 한계가 있다. 향후에 다양한 사전정보를 통해 단어 임베딩을 표현하여 제한된 단어수를 해결하고, 문맥정보에 따른 유동적인 의미 벡터값을 표현할 수 있다면 더 좋은 성능을 보일 거라 기대한다.

표 1. fine-grained English all-words WSD 테스트 셋에 대한 F1 Score

System	All
MFS	65.5
Lesk$_{ext+emb}$	64.2
Babelfy	66.4
IMS$_{emb}$	70.1
Bi-LSTM(att+LEX+POS)	69.9
GAS$_{ext}$(*concat*)	70.6
GCN+GAS$_{ext}$(*concat*)	**70.8**

감사의 글

본 연구는 과학시술정보통신부 및 정보통신기획평가원의 대학ICT연구센터지원사업의 연구결과로 수행되었음(IITP-2018-0-01405). 이 논문은 2017년도 정부(미래창조과학부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(No.NRF-2017M3C4A7068189).

참고문헌

[1] M. Lesk, “Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone,” *Proceedings of the 5th annual international conference on Systems documentation*, pp. 24–26, 1986.

[2] E. Agirre, O. López de Lacalle, and A. Soroa, “Random walks for knowledge-based word sense disambiguation,” *Computational Linguistics*, Vol. 40, No. 1, pp. 57–84, 2014.

[3] A. Moro, A. Raganato, and R. Navigli, “Entity linking meets word sense disambiguation: a unified approach,” *Transactions of the Association for Computational Linguistics*, Vol. 2, pp. 231–244, 2014.

[4] O. Dongsuk, S. Kwon, K. Kim, and Y. Ko, "Word sense disambiguation based on word similarity calculation using word vector representation from a knowledge-based graph," *Proceedings of the 27th International Conference on Computational Linguistics*, pp. 2704–2714, 2018.

[5] D. Ustalov, D. Teslenko, A. Panchenko, M. Chernoskutov, C. Biemann, and S. P. Ponzetto, "An unsupervised word sense disambiguation system for under-resourced languages," *arXiv preprint arXiv:1804.10686*, 2018.

[6] I. Iacobacci, M. T. Pilehvar, and R. Navigli, "Embeddings for word sense disambiguation: An evaluation study," *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 897–907, 2016.

[7] A. Raganato, C. D. Bovi, and R. Navigli, "Neural sequence learning models for word sense disambiguation," *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 1156–1167, 2017.

[8] A. Raganato, J. Camacho-Collados, and R. Navigli, "Word sense disambiguation: A unified evaluation framework and empirical comparison," *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pp. 99–110, 2017.

[9] F. Luo, T. Liu, Q. Xia, B. Chang, and Z. Sui, "Incorporating glosses into neural word sense disambiguation," *arXiv preprint arXiv:1805.08028*, 2018.

[10] G. A. Miller, "Wordnet: a lexical database for english," *Communications of the ACM*, Vol. 38, No. 11, pp. 39–41, 1995.

[11] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," *Advances in neural information processing systems*, pp. 3111–3119, 2013.

[12] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," *Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1532–1543, 2014. [Online]. Available: http://www.aclweb.org/anthology/D14-1162

[13] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," *Transactions of the Association for Computational Linguistics*, Vol. 5, pp. 135–146, 2017.

[14] S. Vashishth, M. Bhandari, P. Yadav, P. Rai, C. Bhattacharyya, and P. Talukdar, "Incorporating syntactic and semantic information in word embeddings using graph convolutional networks," *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 3308–3318, 2019.

[15] D. Marcheggiani and I. Titov, "Encoding sentences with graph convolutional networks for semantic role labeling," *arXiv preprint arXiv:1703.04826*, 2017.

[16] P. Basile, A. Caputo, and G. Semeraro, "An enhanced lesk word sense disambiguation algorithm through a distributional semantic model," *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pp. 1591–1600, 2014.

[17] R. Navigli and S. P. Ponzetto, "Babelnet: Building a very large multilingual semantic network," *Proceedings of the 48th annual meeting of the association for computational linguistics*, pp. 216–225, 2010.