지식 그래프 기반 Entity 검색 모델

2021. 12. 03

수탁 기관: 연세대학교 산학협력단

제 1 장 서론

Entity 검색 모델은 정보 검색 (Information Retrieval) 분야의 중요한 임무로써 사용자의 특정한 질의에 대해서 순서를 가지는 entity 의 리스트를 리턴하는 것을 목표로 한다. 지식 그래프는 다양한 소스로부터 축적한 시맨틱 검색 정보를 담고 있는 Knowledge Base 로 만들어진 그래프로 대량의 정보를 인코딩하여 지식과 지식 간의 관계를 이해하기 쉽게 구조화한다. 지식 그래프의 예시로는 FreeBase[11], DBpedia[12], WikiData[13], YAGO[14] 등 여러 기업과 재단에서 배포한 온라인 컬렉션들이 있다. 관련 있는 지식을 노드와 간선으로 연결하여 그래프로 표현한 지식 그래프의 장점은 정보 검색, 특히 Entity 검색 분야에서 널리 활용되고 있으며 실제 구글의 검색 엔진 등의 실용 분야에서도 널리 사용되고 있다.

기존의 정보 검색 분야에서 주로 제시되었던 직접적인 용어 일치 (term matching) 기반 entity 검색 모델은 사용자 질의 문장의 단어와 entity 사이의 단어 격차(vocabulary gap)로 인한 문제를 겪어왔다. 이러한 문제를 해결하기 위해 지식 그래프를 저차원 공간으로 벡터화 하는 임베딩 방법들이 제시되었다. 그러나 많은 온라인 지식 그래프들은 구조화된 트리플과 구조화되지 않은 텍스트를 모두 포함하고 있기 때문에 통일된 형태로 데이터를 표현하지 않는다. 따라서 사용자의 질의와 지식 그래프에 포함된 트리플 사이의 단어에 격차가 생기게 된다.

본 문서에서는 이러한 문제를 해결하기 위해서 entit 문서에서 토픽을 추출하고 entity 검색에 활용하는 LDA (Latent Dirichlet Allocation) 모델을 사용한다. 본 문서에서 제안하는 방법을 통해 지식 그래프와 사용자 질의 사이의 단어 격차를 줄여 기존의 지식 그래프 기반 entity 검색 모델[15]의 취약점을 보간하고 entity 검색 성능을 향상시킨 통합 모델을 제시한다.

다음으로 2 장에서는 지식 그래프 기반 entity 검색 연구에 대한 기존 연구의 특징 및 장단점에 대해 기술한다. 3 장에서는 지식 그래프 기반 Entity 검색 모델의 세부 동작 과정을 제시한다. 4 장에서는 제안하는 방법을 적용한 지식 그래프 기반 entity 검색 모델의 성능 분석 결과를 제시한다. 5 장에서는 본 문서에 제안된 기법의 의미와 결론을 제시한다.

제 2 장 기술 동향 분석

최근 정보 검색 분야에서는 기존 문서에 있는 데이터의 의미와 구조를 정의한 그래프 형식의 지식 그래프를 활용하여 주어진 질의와 관련 높은 entity 검색 방법에 대한 연구가 활발히 이루어지고 있다.

초기 연구는 질의에 있는 단어들과 위키피디아와 같이 글로 서술된 문서에 등장하는 단어들의 횟수를 비교하여 연관된 entity 를 추출하는 문서 중심 entity 검색 방법[6]이 시도되었고, 지식 그래프에서 entity 를 구성하고 있는 요소들을 사용하여 entity 를 표현하는 하나의 문서를 생성한 후 질의에 대해 가장 관련된 문서를 추출하면 결과적으로 해당 문서가 표현하고 있는 entity 가 추출되는 방식[20, 7]이 제안되었다. 하지만, 질의에 쓰인 단어와는 연관이 있지만 명시적으로 언급되지 않은 단어와의 관계에 대한 정보를 포함할 수 없다는 한계가 있다. 이를 보완하기 위해 인공신경망(neural network)를 기반으로 단어 벡터를 학습하는 word2vec 이나 GloVe 를 사용하여 지식 그래프의 구성 요소인 entity 와 relation 을 가리키는 자연어들이 고정된 차원의 실수 벡터로 표현되어 질의에 대해 관련도가 높은 entity 를 지식 그래프에서 검색해주는 방식[1]이 많이 활용되고 있다.

이러한 지식 그래프 임베딩 방식에서는 entity 들의 관계를 저차원에 임베딩된 구성요소 간의 전환(translation)으로써 표현하는 연구[2,3,4]가 시도되었고, 하나의 entity 에 존재하는 여러 relation 들과의 연관관계를 통해 내제된 의미를 표현하여 entity 의 주변 관계들에 대한 정보까지 포함하는 방법[5], 트리플을 구성하는 주어, 목적어로서의 entity 를 서로 다르게 학습하여 비대칭 관계 정보를 얻는 방법[8] 등이 제안되었다.

제 3 장 모델 설계

본 문서에서 제시하는 지식 그래프 기반 Entity 검색 모델은 기존의 연구 모델[15]의 취약점을 보완하는 모델을 추가하여 검색 성능을 향상시켰다. 기존의 연구 모델[15]은 2 개의 독립적인 모듈로 이루어져 있으며, 첫 번째 모듈로 제안하는 방법인 KEWER 와 두번째 모듈로 제안하는 방법으로 정통적인 정보 검색 방법인 BM25F[16, 17] 모듈로 이루어져 있다. 여기에 확률적 토픽 모델 LDA[22] 모듈을 더해 entity 검색 성능을 향상시켰다. 이번 장에서는 각각의 모듈에 대한 동작과정을 설명한다. 그리고 각 모듈의 검색 결과를 통합하여 entity 검색 결과를 보간 하는 방법에 대해 설명한다.

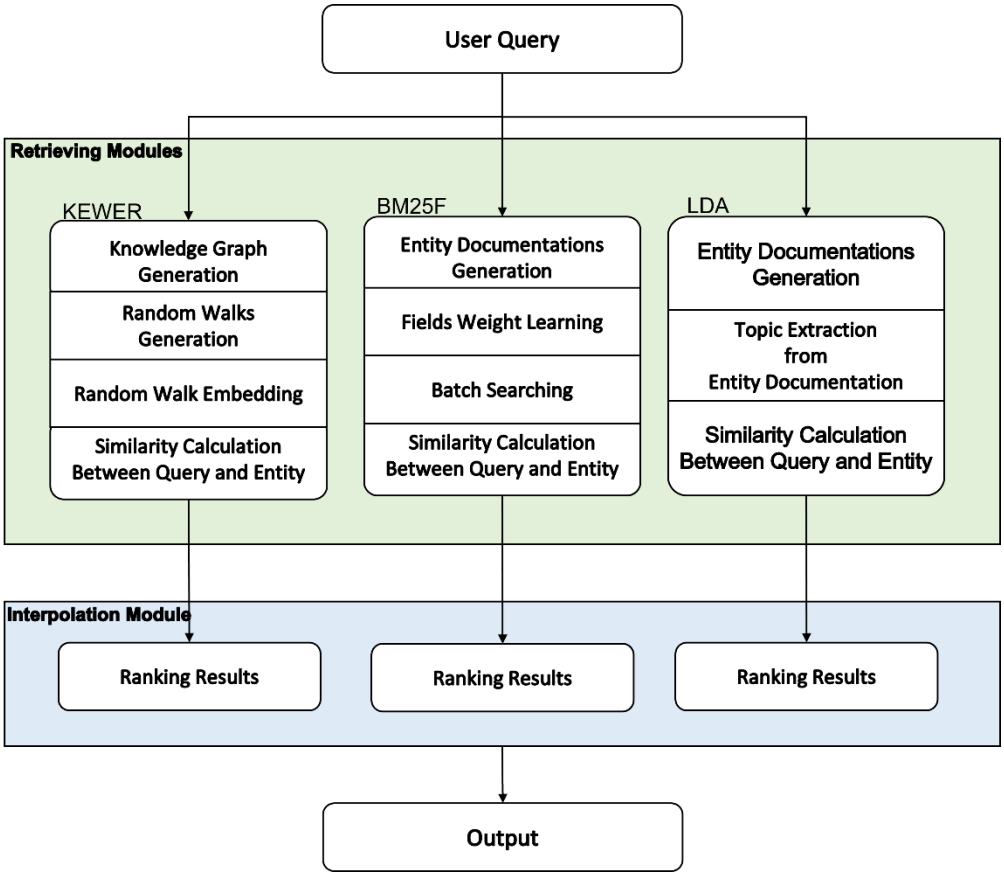

그림 1. 전체 구조도

위의 그림 1 과 같이 지식 그래프 기반 entity 검색 모델은 3 개의 독립적인 모듈로 이루어져 있다. 첫번째 모듈인 KEWER 는 지식 그래프의 구조적 특성과 지식 그래프의 요소들에 대한 특성을 반영하도록 그래프의 랜덤 워크를 생성하고 이를 Skip-Gram-based Word2Vec 모델을 통해 Entity 를 임베딩하는 모듈이다. KEWER 모듈은 Entity 임베딩 값을 이용하여 사용자 질의에 대해서 연관도 점수가 높은 entity 들을 코사인 유사도 계산을 통해 검색하고 평가한다.

사용자 질의 Q를 이루고 있는 각 단어들 $q_1, \ldots, q_k$에 대해서 가중치를 부여한 임베딩 값을 모두 더하여 질의벡터를 만드는 수식은 아래와 같다. (단, $p(q_i)$은 지식 그래프에서 질의 단어 q_i의 빈도확률, v_{q_i}는 q_i의 임베딩 값, a는 자유 파라미터[22])

$$q = \sum_{i=1}^{k} \frac{a}{p(q_i) + a} v_{q_i} \tag{1}$$

그 다음으로 계산된 질의 벡터 q와 entity 벡터 v_e 와의 코사인 유사도 계산을 통해 질의와 연관되는 entity 를 검색하는 수식은 아래와 같다.

$$\text{KEWER}(Q, e) = \cos(q, v_e) \tag{2}$$

$$\cos(q, v_e) = \frac{q \cdot v_e}{\|q\| \cdot \|v_e\|} \tag{3}$$

두번째 모듈로는 BM25F 가 있다. BM25F 는 Bag-of-words 개념을 기반으로 질의에 있는 용어가 각각의 문서에 얼마나 자주 등장하는지를 평가하여 사용자 질의에 대해 문서와의 연관성을 평가하는 순위 알고리즘인 BM25(Best Match 25)[18, 19]의 확장이다. 기존의 BM25 알고리즘에서 생략한 문서 구조에 따른 가중치를 부과한 평가 방법을 사용한다. 이 문서에서 제안하는 모델은 FDSM[20]에서 제안한 entity representation 의 스키마를 기반으로 Galago 검색 엔진[21]을 이용하여 entity 를 5 가지 영역(names, category, similar entity names, attributes, related entity names)으로 인덱싱하고 사용자 질의와 연관되는 entity 를 검색한다.

세번째 모듈은 주어진 문서에 대하여 각 문서의 토픽을 추출하는 확률적 토픽 모델 기법인 LDA(Latent Dirichlet Allocation)[22]이다. LDA 모델을 사용하여 entity 에 대한 토픽을 추출한 뒤, 사용자 질의와의 연관성 계산을 통해 entity 를 검색한다[23]. 사용자 질의는 entity linking 모델인

DBpedia Spotlight API 를 이용하여 생성된 주석을 이용하였다. 그리고 LDA 모델의 문서로 FDSM[20]에서 제안한 entity representation 을 각각의 entity 에 대응되는 문서로 사용하였다. 그 다음으로 문서에서 추출된 토픽과 사용자 질의에 대해 가장 연관도가 높은 entity 를 검색한다. q는 사용자 질의, w는 사용자 질의에 속하는 단어, z는 토픽, d는 문서를 의미한다. $p(w|z)$는 토픽 z가 질의에 속한 단어 w를 포함할 확률을 의미하며, $p(z|d)$는 문서 d가 토픽 z를 포함할 확률을 의미할 때, 사용자 질의 Q와 연관 있는 entity e를 계산하는 식은 아래와 같다.

$$\mathrm{LDA}(Q,e) = \sum_{w}^{|q|} \sum P(w|z)P(z|d) \tag{4}$$

마지막으로 entity 검색 모듈의 결과를 통합하여 최종 검색 결과를 생성하는 Interpolation 모듈에 대해 설명한다. 앞서 설명한 세 개의 entity 검색 모듈은 각각 독립적으로 작동하여 사용자 질의에 대해서 연관되는 entity 를 검색하고 사용자 질의와의 연관성에 따라 entity 를 순서대로 나열하여 하나의 리스트를 반환하게 된다. 반환된 entity 리스트는 Interpolation 모듈의 입력으로 쓰여, $nDCG_{10}$를 평가 기준으로 삼아 5 cross validation folds 를 통해 파라미터 α, β를 최적화한다. Interpolation 모듈은 파라미터 α, β를 0 에서부터 시작하여 0.025 씩 값을 증가시켜가며 최적의 성능을 보이는 값을 찾아낸다. 사용자 질의 Q와 entity e에 대한 세 개 모듈의 검색 결과를 통합하는 수식은 아래와 같다. (단, $0 \le \alpha + \beta \le 1$)

$$\begin{aligned} &Interpolation(Q,e) \\ &\quad = \alpha \cdot \mathrm{KEWER}(Q,e) + \beta \cdot \mathrm{BM25F}(Q,e) \\ &\quad +(1-\alpha-\beta) \cdot \mathrm{LDA}(Q,e) \end{aligned} \tag{5}$$

제 4 장 구현 및 검증

이번 장에서는 본 문서에서 제안하는 모델에서 사용하는 데이터 셋과 검증 결과에 대해 설명한다. 지식 그래프 기반 Entity 검색 모델에서 사용하는 지식 그래프는 DBpedia[12]의 2015-10 영어 버전을 사용한다. 사용자 질의 및 평가 데이터로는 정보검색 표준 데이터 셋인 DBpedia-entity v2[24]를 사용한다. DBpedia-entity v2 는 4 개의 카테고리로

이루어진 데이터 셋으로 총 467 개의 질의로 이루어져 있으며 각각의 카테고리에 대한 설명은 아래의 표와 같다.

카테고리	설명	예시	개수
SemSearch_ES	개체명 질의	"brooklyn bridge"	113
INEX-LD	IR 형식 키워드 질의	"electronic music genres"	99
QALD2	자연어 질의	"Who is the mayor of Berlin?"	115
ListSearch	리스트 검색 질의	"Professional sports teams in Philadelphia"	140

표 1. DBpedia-entity v2 데이터 셋 설명

지식 그래프 기반 entity 검색 모델의 성능을 평가하는 지표는 $nDCG$ (Normalized Discounted Cumulative Gain)[25]을 사용한다. 이 지표는 지식 그래프에서 질문에 대해 검색된 entity 간의 상대적인 순위의 적합성을 평가하는 성능 지표이다. $nDCG$ 평가 지표는 검색 결과의 연관성에 대한 평가뿐만 아니라 연관 점수가 높은 검색 결과가 더 상위에 노출되었는지 평가하는 지표이며 정보 검색 모델에 널리 사용되는 평가 지표이다. $nDCG$ 지표의 계산 과정은 아래 수식과 같다. (단, rel_i 는 i 번째 요소의 연관 점수, rel_i^{opt}은 rel_i 값을 오름차순으로 재배열한 값)

$$\mathrm{DCG_k} = \sum_{i=1}^{k} \frac{rel_i}{\log_2(i+1)} \tag{6}$$

$$\mathrm{IDCG_k} = \sum_{i=1}^{k} \frac{rel_i^{opt}}{\log_2(i+1)} \tag{7}$$

$$\mathrm{nDCG_k} = \frac{DCG_k}{IDCG_k} \tag{8}$$

지식 그래프 기반 Entity 검색 모델은 LDA 모델의 토픽 개수 별 성능을 측정하여 실험을 진행했다. LDA 모듈을 추가한 실험 결과는 기존의 연구 모델인 KEWER 모듈과 BM25F 모듈의 entity 검색 결과를 결합한 검색 성능보다 높은 검색 성능을 보였다. 특히, LDA 모듈의 토픽 개수를 90 개로 설정하여 실험을 진행했을 때 기존의 연구모델의 $nDCG_{10}$ 과

$nDCG_{100}$에서 각각 1.4%, 11.8%의 가장 높은 향상을 보여주었다. 토픽의 개수에 따른 지식 그래프 기반 Entity 검색 모델의 성능은 아래와 같다.

	nDCG@10	nDCG@100
KEWER+BM25F	0.483	0.56
KEWER+BM25F+LDA20	0.491	0.675
KEWER+BM25F+LDA50	0.493	0.676
KEWER+BM25F+LDA70	0.495	**0.678**
KEWER+BM25F+LDA90	**0.497**	**0.678**
KEWER+BM25F+LDA100	0.494	0.675
KEWER+BM25F+LDA110	0.492	0.675
KEWER+BM25F+LDA120	0.492	0.676
KEWER+BM25F+LDA150	0.491	0.675

표 2. 지식 그래프 기반 Entity 검색 모델 성능 결과

제 5 장 주요 결론

Entity 검색 모델은 정보 검색 (Information Retrieval) 분야의 중요한 임무로써 지식 그래프를 이용한 정보 검색은 실제 구글의 검색 엔진 등의 실용 분야에서 널리 사용되고 있다. 본 문서에서는 사용자의 질의와 지식 그래프에 포함된 트리플 사이의 단어 격차(vocabulary gap)를 줄인 지식 그래프 기반 Entity 검색 모델을 제안한다. 기존의 연구 결과와 비교했을 때 $nDCG_{10}$과 $nDCG_{100}$에서 각각 최대 1.4%, 11.8%의 성능 향상을 보였다. 추후에는 다중 홉 복잡 질의에 대해서 지식 그래프 기반 entity 검색 모델의 성능을 높이기 위한 연구를 진행할 예정이다.

참고문헌

[1] Bordes, Antoine, et al. "Learning structured embeddings of knowledge bases." Twenty-Fifth AAAI Conference on Artificial Intelligence. 2011.

[2] Bordes, Antoine, et al. "Translating embeddings for modeling multi-relational data." Advances in neural information processing systems 26 (2013).

[3] Wang, Zhen, et al. "Knowledge gragh embedding by translating on hyperplanes." Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 28. No. 1. 2014.

[4] Lin, Yankai, et al. "Learning entity and relation embeddings for knowledge gragh completion." Twenty-ninth AAAI conference on artificial intelligence. 2015.

[5] Nickel, Maximilian, Volker Tresp, and Hans-Peter Kriegel. "A three-way model for collective learning on multi-relational data." Icml. 2011.

[6] Demartini, Gianluca, Tereza Iofciu, and Arjen P. De Vries. "Overview of the INEX 2009 entity ranking track." International Workshop of the Initiative for the Evaluation of XML Retrieval. Springer, Berlin, Heidelberg, 2009.

[7] Nikolaev, Fedor, Alexander Kotov, and Nikita Zhiltsov. "Parameterized fielded term dependence models for ad-hoc entity retrieval from knowledge gragh." Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval. 2016.

[8] Trouillon, Théo, et al. "Complex embeddings for simple link prediction." International conference on machine learning. PMLR, 2016.

[9] Naseri, Shahrzad, et al. "Exploring Summary-Expanded Entity Embeddings for Entity Retrieval." CIKM Workshops. 2018.

[10] Järvelin, Kalervo, and Jaana Kekäläinen. "Cumulated gain-based evaluation of IR techniques." ACM Transactions on Information Systems (TOIS) 20.4 (2002): 422-446.

[11] Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: A collaboratively created graph database for structuring human knowledge. In Proceedings of the 2008 ACM SIGMOD international conference on Management of Data.

[12] Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. 2007. Dbpedia: A nucleus for a web of open data. In The Semantic Web. Springer.

[13] Denny Vrandecic and Markus Krötzsch. 2014. Wikidata: A free collaborative knowledge base. Communications of the ACM 57, 10 (2014), 78-85

[14] Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. 2007. Yago: A core of semantic knowledge. In Proceedings of the 2007 International Conference on World Wide Web. ACM

[15] Nikolaev, Fedor, and Alexander Kotov. "Joint word and entity embeddings for entity retrieval from a 지식 그래프." Advances in Information Retrieval 12035 (2020): 141.

[16] Robertson, Stephen, and Hugo Zaragoza. The probabilistic relevance framework: BM25 and beyond. Now Publishers Inc, 2009.

[17] Zaragoza, Hugo, et al. "Microsoft Cambridge at TREC 13: Web and Hard Tracks." Trec. Vol. 4. 2004.

[18] Balog, Krisztian, and Robert Neumayer. "A test collection for entity search in DBpedia." Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval. 2013.

[19] Tonon, Alberto, Gianluca Demartini, and Philippe Cudré-Mauroux. "Combining inverted indices and structured search for ad-hoc object retrieval." Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval. 2012.

[20] Zhiltsov, Nikita, Alexander Kotov, and Fedor Nikolaev. "Fielded sequential dependence model for ad-hoc entity retrieval in the web of data." Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval. 2015.

[21] https://sourceforge.net/p/lemur/wiki/Galago/

[22] Blei, David M., Andrew Y. Ng, and Michael I. Jordan. "Latent dirichlet allocation." the Journal of machine Learning research 3 (2003): 993-1022.

[23] Wei, Xing, and W. Bruce Croft. "LDA-based document models for ad-hoc retrieval." Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval. 2006.

[24] Hasibi, Faegheh, et al. "DBpedia-entity v2: a test collection for entity search." Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval. 2017.