KnowFIRES: a Knowledge-graph Framework for Interpreting Retrieved Entities from Search

Negar Arabzadeh[1⋆], Kiarash Golzadeh[*1], Christopher Risi[*1], Charles L.A. Clarke[1], and Jian Zhao[1]

[1] University of Waterloo
[2] {narabzad, kgolzade, cjrisi, claclark, jianzhao}@uwaterloo.ca

Abstract. Entity retrieval is essential in information access domains where people search for specific entities, such as individuals, organizations, and places. While entity retrieval is an active research topic in Information Retrieval, it is necessary to explore the explainability and interpretability of them more extensively. `KnowFIRES` addresses this by offering a knowledge graph-based visual representation of entity retrieval results, focusing on contrasting different retrieval methods. `KnowFIRES` allows users to better understand these differences through the juxtaposition and superposition of retrieved sub-graphs. As part of our demo, we make `KnowFIRES` [3] web interface and its source codepublicly available[4].

1 Introduction

Entity retrieval is crucial in information access domains where users search for specific items, such as individuals, organizations, and places, that are distinguishable by their characteristics, attributes, and connections to other entities [12, 7, 15]. Researchers estimate that over 40% of web search queries are for entities and major web search engines use extensive knowledge bases to respond to these requests [4, 24, 27, 9, 38, 32, 23, 33]. While entity retrieval has been investigated extensively, little attention has been given to the interpretability and explainability of such systems [34]. Increased explainability of search results has been shown to [35, 1]: 1) increase searcher's trust in the system and thus the searcher's satisfaction rate, 2) increase the probability of satisfying the information need behind the query, and 3) decrease the chance of spreading misinformation due to more informative reasoning. In this demo, we introduce an explainable entity retrieval system called the Knowledge-graph Framework for Interpreting Retrieved Entities from Search (`KnowFIRES`) that not only presents the entity names in a ranked list but also visualizes the entities through a knowledge graph representation. `KnowFIRES` focuses on highlighting similarities and differences of the retrieved entities on a per-retriever basis. By including information beyond the entity names, which reflects the relationship between them, we hope to allow searchers to gain additional insights through more explainable results.

⋆ Equal Contributions.
[3] Demo: http://knowfires.live, Source: https://github.com/kiarashgl/KnowFIRES
[4] A demonstration of the tool: https://www.youtube.com/watch?v=9u-877ArNYE

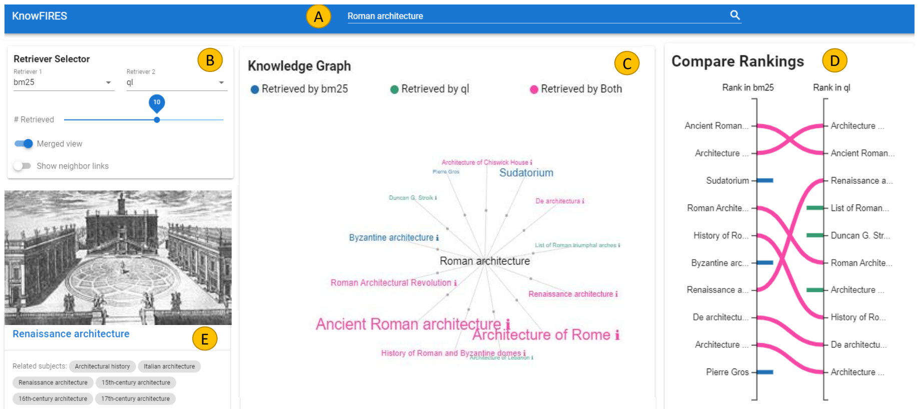

Fig. 1. *KnowFIRES* interface. A) Search Bar, B) Entity Retrieval Selection Panel, C) Knowledge Graph View, D) Entity Ranking Comparison View, and E) Related content.

Knowledge graph presentation of entities has been explored in domains such as biomedical and graph visualization [18, 5, 39, 40, 10]. However, to the best of our knowledge, `KnowFIRES` is the first visualization tool that combines *ad hoc* retrieval results with entity presentation in a unified framework and recreates the knowledge graph on top of these results. In `KnowFIRES`, we first retrieve relevant entities from a collection of entities and their metadata. Then, given the top-k most relevant entities, we build a knowledge graph and allow searchers to traverse this graph to better understand relationships between the entities, as well as their relationship to the original query. Furthermore, `KnowFIRES` offers searchers the option of accessing related content associated with each displayed entity, providing a more comprehensive understanding. More importantly, `KnowFIRES` allows easier comparison between different entity retrievers, by following the visual comparison conceptual framework proposed by Gleicher [14].

While the community has explored many different methods, there is no single retriever that can satisfy all possible use cases and applications [16, 2]. Reasons include the variability of data sources and contextual variability. Additionally, performance trade-offs matter in real-world applications in terms of accuracy, speed, and resource requirements. Given these considerations, it is often necessary to use a combination of retriever methods that could complement each other for different purposes to satisfy the requirements of different use cases and applications [6, 3, 25]. To tackle this problem, in `KnowFIRES`, we visualize the degree of complementarity of a set of retrievers of interest and notably visualize the similarities and differences between their retrieved results. As such, `KnowFIRES` can be utilized for deciding the appropriate retriever on per application basis and even help researchers on developing enhanced entity retrievers. To the best of our knowledge, this feature has not been explored in previous knowledge graph-based visualization tools. Our demo provides a pioneering solution in entity retrieval, merging trusted techniques from information visualization to compare results from different entity retrieval methods with the power of knowledge graphs to offer users a more interpretable, trustworthy search experience.

2 Overview of `KnowFIRES` Design

We frame our design using Gleicher's visual comparison framework [13]. Our *comparative elements* are query results from different entity retrievers. Each entity retrieval method returns separate knowledge graphs to the front-end, and we contrast these results through our visualization. As for our *comparative challenges*, knowledge graphs' size and complexity are the greatest areas of concern. The visualization could become cluttered and uninterpretable for retrieved entities with a very dense set of relations. We address these challenges by incorporating various designs and making our graphs interactive, so the searcher can drag around the frame and zoom in on specific nodes. Our *comparative strategy* for dealing with the dense knowledge graph challenge is to trim our results with *subset selection*. The slider in our design enables the user to limit the number of retrieved entities from the entity retrievers. We typically expect a small number of entities to provide more than enough information. We include a toggle for visualizing one-hop connections from the retrieved entities. The one-hop linked entities add important contextual information, but allowing the users to remove them from the visualization was an important addition to enhance interpretability. Finally, our *comparative design*, *`KnowFIRES`* (Figure 1), is a *superposition* or *juxtaposition* of knowledge graphs highlighting the similarities and differences between the results by colour-coding the knowledge graph's nodes.

2.1 Entity Retrieval Methods

We employ various entity retrieval method that are categorized into two groups: 1) *sparse retrievers*, which match exact keywords using inverted indexes, and 2) *dense retrievers*, which measure semantic similarity using dense vector representations [29, 30, 8, 19]. More specifically, our sparse retrievers include **BM25** [30] and **QL** [36], with **BM25-PRF** and **QL-PRF** augmented by RM3 pseudo-relevance feedback. For dense retrievers, we use **SentenceBERT** [28, 37] and **ColBERT** [20, 31], each offering unique approaches to estimating relevance. This diversity enables us to explore the impact of sparse versus dense retrievers.

2.2 System Architecture

We employ DBpedia V2 [21, 16], a vast knowledge base extracted from Wikimedia projects, as our entity corpus. While adaptable to other corpora, our demonstration uses the English subset of DBpedia version 2015-10 [16, 17, 11, 26]. This subset mandates entities to have a title and abstract (rdfs:label and rdfs:comment predicates), excluding certain page types but including list pages. Our corpus comprises 4.6 million entities, each uniquely identified by URI. Entity retrievers were implemented in Python 3.8 using the pyserini library [22]. Data was served to the front-end via Flask and deployed on the backend server. The front-end communicates with the back-end through a REST API, enabling search queries, retriever selection, and result entity count specification, with responses provided in JSON format, including entity rank, score, relevant entities, and metadata. The front-end, implemented as a single-page *VueJS* application, utilizes the Vuetify framework for Material Design components like sliders, dropdowns, and grids. Visualizations rely on the *D3* library. For the knowledge graph, we employ the *Force-Graph* library, a D3 node-link diagram wrapper that integrates the D3-force library for node arrangement. The Entity Ranking Comparison View is custom-built with D3. During each search query, the front-end retrieves entities from each retriever through the backend API. JSON responses are parsed and stored to construct nodes and links in the Knowledge Graph view. In the final step, the code is production-ready, built using Webpack, and deployed on a Linux VPS using the *serve utility.* Finally, the code was built for production with Webpack and deployed on a Linux VPS using serve.

2.3 Visual Design

The Knowledge Graph View The primary visual representation, as shown in Figure 1-*C*. It displays the most relevant entities related to a query, with nodes color-coded based on each of the entity retrievers (green, blue, or magenta for common entities). The node size reflects the relevance of the entity to the query.
Superposition vs Juxtaposition. This feature offers two graph layouts. The superposition layout merges knowledge graphs from two retrieval methods, with color encodings helping differentiate between them. Alternatively, the juxtaposition layout shows two separate graphs, maintaining the color distinction.
Neighbour Links. It is possible to navigate through neighbor links of an entity, providing context about a particular node. This feature, includes grey links with descriptive text labels giving insight into the relationship between entities.
The Entity Ranking Comparison View. This interface, depicted in Figure 1-*D*, enables users to compare entity rankings between retrieval methods. Entities from each method are shown side by side, with a magenta line connecting common entities, highlighting ranking disparities between the two methods.
The Additional Content Panel. Beyond the entity's name, we provide supplemental information, including a link to the entity's Wikipedia page, the entity type, related tags, and occasionally an image as shown in Figure 1-*E*.

2.4 Interactivity

Our design prioritizes interactivity to empower users in exploring and comparing entity retrieval results. The entity retriever selection panel (Figure 1-B) offers dropdowns to select entities for comparison, a slider to determine linked entity quantity and a toggle switch for juxtaposition or superposition knowledge graph views. Another toggle hides neighbor links for decluttering dense queries. In the Knowledge Graph View (Figure 1-C), users can zoom using the scroll wheel, move nodes by click-and-drag, and click nodes for additional information in the *Related Content* panel. Clicking a node highlights its rank in the Compare Ranking panel (Figure 1-D).

3 Concluding Remarks

In `KnowFIRES`, we visualize differences between entity retrievers within a knowledge graph, enhancing explainability and interpretability of retrieved results. Our demo showcases its efficiency and potential for broader applications. Future improvements include user-friendly visualizations, API integration, customization in graph traversal, and addition of metadata like external links.

References

1. Anand, A., Lyu, L., Idahl, M., Wang, Y., Wallat, J., Zhang, Z.: Explainable information retrieval: A survey. arXiv preprint arXiv:2211.02405 (2022)
2. Arabzadeh, N., Mitra, B., Bagheri, E.: Ms marco chameleons: Challenging the ms marco leaderboard with extremely obstinate queries. In: Proceedings of the 30th ACM International Conference on Information & Knowledge Management. pp. 4426–4435 (2021)
3. Arabzadeh, N., Yan, X., Clarke, C.L.: Predicting efficiency/effectiveness trade-offs for dense vs. sparse retrieval strategy selection. In: Proceedings of the 30th ACM International Conference on Information & Knowledge Management. pp. 2862–2866 (2021)
4. Balog, K.: Entity retrieval. (2018)
5. Barsky, A., Munzner, T., Gardy, J., Kincaid, R.: Cerebral: Visualizing multiple experimental conditions on a graph with biological context. IEEE Transactions on Visualization and Computer Graphics **14**(6), 1253–1260 (2008). https://doi.org/10.1109/TVCG.2008.117, http://ieeexplore.ieee.org/document/4658137/
6. Cormack, G.V., Clarke, C.L., Buettcher, S.: Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In: Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval. pp. 758–759 (2009)
7. De Cao, N., Izacard, G., Riedel, S., Petroni, F.: Autoregressive entity retrieval. arXiv preprint arXiv:2010.00904 (2020)
8. Devlin, J., Chang, M., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. CoRR **abs/1810.04805** (2018), http://arxiv.org/abs/1810.04805
9. Fetahu, B., Gadiraju, U., Dietze, S.: Improving entity retrieval on structured data. In: The Semantic Web-ISWC 2015: 14th International Semantic Web Conference, Bethlehem, PA, USA, October 11-15, 2015, Proceedings, Part I 14. pp. 474–491. Springer (2015)
10. Fujiwara, T., Zhao, J., Chen, F., Ma, K.L.: A visual analytics framework for contrastive network analysis. In: 2020 IEEE Conference on Visual Analytics Science and Technology (VAST). pp. 48–59. IEEE (2020). https://doi.org/10.1109/VAST50239.2020.00010, https://ieeexplore.ieee.org/document/9308624/
11. Gerritse, E.J., Hasibi, F., de Vries, A.P.: Graph-embedding empowered entity retrieval. In: Advances in Information Retrieval: 42nd European Conference on IR Research, ECIR 2020, Lisbon, Portugal, April 14–17, 2020, Proceedings, Part I 42. pp. 97–110. Springer (2020)
12. Gillick, D., Kulkarni, S., Lansing, L., Presta, A., Baldridge, J., Ie, E., Garcia-Olano, D.: Learning dense representations for entity retrieval. arXiv preprint arXiv:1909.10506 (2019)
13. Gleicher, M.: Considerations for visualizing comparison. IEEE transactions on visualization and computer graphics **24**(1), 413–423 (2017)
14. Gleicher, M., Albers, D., Walker, R., Jusufi, I., Hansen, C.D., Roberts, J.C.: Visual comparison for information visualization. Information Visualization **10**(4), 289–309 (2011)
15. Hasibi, F., Balog, K., Bratsberg, S.E.: Exploiting entity linking in queries for entity retrieval. In: Proceedings of the 2016 acm international conference on the theory of information retrieval. pp. 209–218 (2016)

16. Hasibi, F., Nikolaev, F., Xiong, C., Balog, K., Bratsberg, S.E., Kotov, A., Callan, J.: Dbpedia-entity v2: a test collection for entity search. In: Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 1265–1268 (2017)
17. Jafarzadeh, P., Amirmahani, Z., Ensan, F.: Learning to rank knowledge subgraph nodes for entity retrieval. In: Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 2519–2523 (2022)
18. Kang, H., Getoor, L., Shneiderman, B., Bilgic, M., Licamele, L.: Interactive entity resolution in relational data: A visual analytic tool and its evaluation. IEEE Transactions on Visualization and Computer Graphics **14**(5), 999–1014 (2008). https://doi.org/10.1109/TVCG.2008.55
19. Karpukhin, V., Oğuz, B., Min, S., Lewis, P., Wu, L., Edunov, S., Chen, D., Yih, W.t.: Dense passage retrieval for open-domain question answering. arXiv preprint arXiv:2004.04906 (2020)
20. Khattab, O., Zaharia, M.: Colbert: Efficient and effective passage search via contextualized late interaction over bert. In: Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval. pp. 39–48 (2020)
21. Lehmann, J., Isele, R., Jakob, M., Jentzsch, A., Kontokostas, D., Mendes, P.N., Hellmann, S., Morsey, M., van Kleef, P., Auer, S., Bizer, C.: DBpedia – a large-scale, multilingual knowledge base extracted from wikipedia. Semantic Web **6**(2), 167–195 (2015). https://doi.org/10.3233/SW-140134, https://www.medra.org/servlet/aliasResolver?alias=iospressdoi=10.3233/SW-140134
22. Lin, J., Ma, X., Lin, S.C., Yang, J.H., Pradeep, R., Nogueira, R.: Pyserini: A python toolkit for reproducible information retrieval research with sparse and dense representations. In: Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 2356–2362 (2021)
23. Lin, X., Lam, W., Lai, K.P.: Entity retrieval in the knowledge graph with hierarchical entity type and content. In: Proceedings of the 2018 acm sigir international conference on theory of information retrieval. pp. 211–214 (2018)
24. Macdonald, C., Ounis, I.: Voting for candidates: adapting data fusion techniques for an expert search task. In: Proceedings of the 15th ACM international conference on Information and knowledge management. pp. 387–396 (2006)
25. McCabe, M.C., Chowdhury, A., Grossman, D.A., Frieder, O.: A unified environment for fusion of information retrieval approaches. In: Proceedings of the eighth international conference on Information and knowledge management. pp. 330–334 (1999)
26. Nikolaev, F., Kotov, A.: Joint word and entity embeddings for entity retrieval from a knowledge graph. In: Advances in Information Retrieval: 42nd European Conference on IR Research, ECIR 2020, Lisbon, Portugal, April 14–17, 2020, Proceedings, Part I 42. pp. 141–155. Springer (2020)
27. Pound, J., Mika, P., Zaragoza, H.: Ad-hoc object retrieval in the web of data. In: Proceedings of the 19th international conference on World wide web. pp. 771–780 (2010)
28. Reimers, N., Gurevych, I.: Sentence-bert: Sentence embeddings using siamese bert-networks. arXiv preprint arXiv:1908.10084 (2019)
29. Robertson, S., Zaragoza, H., et al.: The probabilistic relevance framework: Bm25 and beyond. Foundations and Trends® in Information Retrieval **3**(4), 333–389 (2009)

30. Robertson, S.E., Walker, S., Jones, S., Hancock-Beaulieu, M.M., Gatford, M., et al.: Okapi at trec-3. Nist Special Publication Sp **109**, 109 (1995)
31. Santhanam, K., Khattab, O., Saad-Falcon, J., Potts, C., Zaharia, M.: ColBERTv2: Effective and efficient retrieval via lightweight late interaction, http://arxiv.org/abs/2112.01488
32. Sciavolino, C., Zhong, Z., Lee, J., Chen, D.: Simple entity-centric questions challenge dense retrievers. arXiv preprint arXiv:2109.08535 (2021)
33. Shehata, D., Arabzadeh, N., Clarke, C.L.A.: Early stage sparse retrieval with entity linking (2022). https://doi.org/10.48550/ARXIV.2208.04887, https://arxiv.org/abs/2208.04887
34. Shehata, D., Arabzadeh, N., Clarke, C.L.A.: Early stage sparse retrieval with entity linking (2022)
35. Singh, J., Anand, A.: Exs: Explainable search using local model agnostic interpretability. In: Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining. pp. 770–773 (2019)
36. Song, F., Croft, W.B.: A general language model for information retrieval. In: Proceedings of the eighth international conference on Information and knowledge management. pp. 316–321 (1999)
37. Thakur, N., Reimers, N., Daxenberger, J., Gurevych, I.: Augmented sbert: Data augmentation method for improving bi-encoders for pairwise sentence scoring tasks (2021)
38. Wu, L., Petroni, F., Josifoski, M., Riedel, S., Zettlemoyer, L.: Scalable zero-shot entity linking with dense entity retrieval. arXiv preprint arXiv:1911.03814 (2019)
39. Zhao, J., Cao, N., Wen, Z., Song, Y., Lin, Y.R., Collins, C.: Fluxflow: Visual analysis of anomalous information spreading on social media. IEEE Transactions on Visualization and Computer Graphics **20**(12), 1773–1782 (2014). https://doi.org/10.1109/TVCG.2014.2346922, http://ieeexplore.ieee.org/document/6876013/
40. Zhao, J., Glueck, M., Breslav, S., Chevalier, F., Khan, A.: Annotation graphs: A graph-based visualization for meta-analysis of data based on user-authored annotations. IEEE Transactions on Visualization and Computer Graphics **23**(1), 261–270 (2017). https://doi.org/10.1109/TVCG.2016.2598543, http://ieeexplore.ieee.org/document/7536110/