Facts That Matter: Dynamic Fact Retrieval for Entity-Centric Search Queries

Atharva Prabhat Paranjpe[1], Rajarshi Bhowmik[1], and Gerard de Melo[2]

[1] Rutgers University–New Brunswick, Piscataway, NJ, USA
[2] Hasso Plattner Institute, University of Potsdam, Potsdam, Germany
atharva.paranjpe@gmail.com, rajarshi.bhowmik@rutgers.edu, gdm@demelo.org

Abstract. Entity-centric queries constitute a significant proportion of all search queries processed by the popular search engines. Answering such queries often involves selecting facts pertaining to an entity from an underlying knowledge graph. Prior work on this draws on hand-crafted features that require scanning the entire knowledge graph beforehand. Instead, we propose a neural method that exploits the linguistic and semi-linguistic nature of the entity search queries and the facts, and can hence be applied dynamically to entirely new sets of candidate facts. We optimize our model using a pairwise loss function to correctly predict the *relevance* and *importance* scores for each fact for a given query entity, while the overall fact ranking is based on a linear combination of these scores. We show that our simple approach outperforms previous work, ensuring better fact retrieval for entity-centric search queries.

1 Introduction

In recent years, knowledge cards have become an integral part of popular Web search engines. Knowledge cards are information boxes that appear on the search engine result pages when a user searches for entity-related information [1, 2]. Such cards provide a series of facts taken from a knowledge graph [6] and enable the user get a brief overview of pertinent key facts about the entity without the need to navigate to various individual web pages.

In practice, different entity-related queries may pertain to quite different aspects of an entity. A search engine query such as "*einstein education*" ought to give preference to other facts than a query such as "*einstein family*". To address this task of dynamic query-specific fact ranking, Hasibi et al. proposed a model called DynES [5] that performs fact retrieval and entity summarization based on a linear combination of two measures: *importance* and *relevance*, and compared the results to human judgments.

However, DyNES is based on hand-crafted features that are cumbersome to compute, as they need to be extracted beforehand from the set of all facts in the large-scale knowledge graph, rendering this method unsuitable for ad hoc settings. Moreover, DynES performs a simple pointwise ranking of the facts, where each fact is considered in isolation, using Gradient Boosted Regression Trees, which learn an ensemble of weak prediction models.

In contrast, we propose a novel model that obviates the need for a cumbersome process of extracting hand-crafted features from the large knowledge graph. Our key contributions are as follows. (1) We propose a deep neural model with a pairwise loss function to address the task of query-dependent fact retrieval for entity-centric search queries. (2) Rather than depending on a large knowledge graph for feature extraction, our model draws on recent advances in Transformers with self attention [9] to better model the linguistic connection between the query and the candidate facts, and thus can be applied even to entirely novel sets of candidate facts. (3) We conduct a set of experimental evaluations showing that our approach outperforms previous work.

2 Preliminaries

In the following, we define relevant terminology that is used in the remainder of the paper. We consider a fact f as a predicate–object pair returned when a query is made with regard to an entity, with that entity serving as the subject.

Definition 1. *(Importance) Importance is an attribute of a fact f that determines its relation to the subject entity s in absolute terms, irrespective of the provided query. It is denoted as $i_s(f)$.*

Definition 2. *(Relevance) Relevance, in turn, describes to what extent a given candidate fact f is pertinent with regard to a given natural language search query q issued by the user along with the entity s as the subject. It is denoted as $r_{s,q}(f)$.*

Definition 3. *(Utility) The overall utility of a fact f with respect to a query q and entity s is defined as a weighted sum of the importance and relevance scores of the fact with respect to query and entity. It is denoted as $u_{s,q}(f)$ and computed as $u_{s,q}(f) = \alpha\, i_s(f) + \beta\, r_{s,q}(f)$*

The weights α, β may be adjusted freely to account for application scenario-specific considerations. Thus, utility relates the fact to the query in a more comprehensive manner than the importance and relevance scores alone can.

3 Model

Given the natural language input query Q as well as a candidate fact $f_i = \langle p, o \rangle \in \mathcal{F}$, where $\mathcal{F}$ is the set of all candidate facts for Q, our model accepts the query along with the natural language labels of p and o and invokes BERT [4], a deep neural Transformer encoder, to encode bidirectional contextual information for the given sequence of input tokens. Since we simultaneously supply both the query and the candidate fact to the Transformer, the self-attention layers are able to establish connections between (parts of) these two inputs.

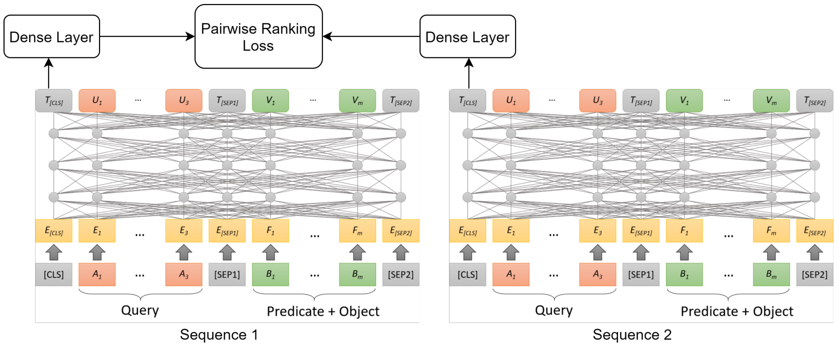

Fig. 1. A schematic diagram of the model architecture.

Before feeding the tokenized sequences Q, P, O to the model, special tokens `[CLS]` and `[SEP]` are inserted into the input sequence. The `[CLS]` token signifies the start of each sequence, while the `[SEP]` token serves as a demarcation point separating the query segment from the fact segment in the input sequence. The resulting sequence of input token identifiers now becomes:

$$\texttt{[CLS]},\, q_1, \ldots, q_l,\ \texttt{[SEP]},\, p_1, \ldots, p_m,\, o_1, \ldots, o_n,\ \texttt{[SEP]}$$

The most relevant component for our task lies in the encoded representation of the `[CLS]` token, which serves as a representation of the entire input sequence. This representation is passed through a fully-connected layer followed by a sigmoid activation function to yield a ranking score, which is then compared to the ground truth. Formally, $g(f_i) = \sigma(\mathbf{W}\mathbf{h_p} + b)$, where $\mathbf{h_p} \in \mathbb{R}^d$ is the `[CLS]` representation from the final hidden layer of the BERT encoder, $\mathbf{W} \in \mathbb{R}^{1\times d}$ and b are trainable parameters, and $\sigma(x) = \frac{1}{1+e^{-x}}$.

The model is trained to minimize a pairwise ranking loss that considers pairs of facts (f_s and f_i) and encourages the model to predict scores for the two involved facts that reflect the correct relative ordering between them. Ideally, the difference between the two predicted scores $(g(f_s) - g(f_i))$ should equal the difference $(r(s) - r(i))$ between the corresponding ground truth ranking scores. These differences are computed as signed values rather than absolute values, so the ordering is crucial. Based on this intuition, we define the loss function as a pairwise mean squared error as follows:

$$\mathcal{L}(g;\mathcal{F},\mathcal{R}) = \sum_{s=1}^{n-1} \sum_{\substack{i=1\\ r(i)<r(s)}}^{n} \Big[\big(r(s) - r(i)\big) - \big(g(f_s) - g(f_i)\big)\Big]^2$$

The final ranking is created by ordering the candidate facts $f_i \in \mathcal{F}$ by $g(f_i)$ in descending order, breaking ties arbitrarily. Thus, if $g(f_i) > g(f_j)$, then f_i should be ranked higher than f_j.

4 Evaluation

We perform our experiments on two dataset variants put forth by Hasibi et al. [5]. For the first variant, `Complete Dataset`, the entire collected data is considered.

This data consists of 100 English language queries, 4,069 facts, and 41 facts per query on average. Their second variant, `URI-only Dataset`, keeps only the subset of facts for which the objects are genuine entities identified by a URI, while facts with literal values are omitted. It contains the same 100 queries, 1,309 facts, and 14 facts per query on average.

We compare several different models, including DynES [5], a BiLSTM Dual Encoder, a $\text{BERT}_{\text{BASE}}$ variant of our model that does not fine-tune the BERT encoder, a pointwise score prediction variant of our model, and our pairwise model. For reproducibility and future research, we release the source code of our model[2]. We use the standard NDCG metric for evaluation with ranked lists of length 5 (NDCG@5) and 10 (NDCG@10), and report the evaluation scores obtained using 5-fold cross validation.

	Utility		Importance		Relevance	
Model	NDCG@5	NDCG@10	NDCG@5	NDCG@10	NDCG@5	NDCG@10
RELIN [3] †	0.4680	0.5322	0.4733	0.5261	0.3514	0.4255
DynES [5] †	0.7547	0.7873	0.7672	0.7792	0.5771	0.6423
Bi-LSTM Dual Encoder	0.4699	0.5357	0.5172	0.5622	0.4613	0.5127
$\text{BERT}_{\text{BASE}}$	0.5092	0.5589	0.5421	0.5871	0.4607	0.5127
Our Model (pointwise)	0.7653	0.7965	0.8358	0.8435	**0.5906**	0.6348
Our Model (pairwise)	**0.7980**	**0.8258**	**0.8635**	**0.8821**	0.5902	**0.6426**

Table 1. 5-fold cross-validation results on the `Complete Dataset`. †: results taken from Hasibi et al. [5].

	Utility		Importance		Relevance	
Model	NDCG@5	NDCG@10	NDCG@5	NDCG@10	NDCG@5	NDCG@10
RELIN [3] †	0.6300	0.7066	0.6368	0.7130	N/A	N/A
LinkSum [7] †	0.6504	0.6648	0.7018	0.7031	N/A	N/A
SUMMARUM [8] †	0.6719	0.7111	0.7181	0.7412	N/A	N/A
DynES [5] †	0.8164	0.8569	0.8291	0.8652	N/A	N/A
Bi-LSTM Dual Encoder	0.6416	0.7225	0.6821	0.7508	N/A	N/A
$\text{BERT}_{\text{BASE}}$	0.7055	0.7675	0.6521	0.7274	0.4563	0.5498
Our Model (pointwise)	0.7850	0.8285	**0.8635**	**0.8821**	0.6165	0.6741
Our Model (pairwise)	**0.8515**	**0.8761**	0.8454	0.8743	**0.6621**	**0.7269**

Table 2. 5-fold cross-validation results on the `URI-only Dataset`. †: results taken from Hasibi et al. [5], who did not report separate relevance prediction results apart from the overall utility prediction results.

The results of our experiments on the `Complete Dataset` and `URI-only Dataset` are given in Tables 1 and 2, respectively. Our model outperforms the DynES model with absolute gains of 4.9% and 13.2% in terms of the NDCG@10 metric on the `Complete Dataset` for the utility and importance-based rankings (as defined in Section 2), respectively. We observe a similar trend for

[2] `https://github.com/AtharvaParanjpe/Dynamic-Fact-Ranking-For-Entity-Centric-Queries`

the `URI-only Dataset`, where our model consistently outperforms the DynES model, with respective absolute gains of 4.3% and 2.0% in the NDCG@5 metric for utility and importance rankings.

The moderate performance of the pre-trained BERT$_{\text{BASE}}$ model suggests that BERT$_{\text{BASE}}$ already has sufficient linguistic information embedded in it to be able to rank the facts to a certain degree. In fact, without any fine-tuning, BERT$_{\text{BASE}}$ outperforms the BiLSTM Dual Encoder baseline in most of the cases.

Our model outperforms the pointwise ranking variant with as high as 8.5% and 5.7% absolute gain in NDCG@5 and NDCG@10 metrics for the utility scores on the `URI-only Dataset`. We conjecture that this is because a pairwise loss function allows the model to better assess the differences between different facts and because this training regime better exploits the available training data.

5 Conclusion

In this paper, we propose a new neural method to learn entity-centric fact rankings, accounting for both the saliency and the relevance of facts with regard to the query. Our method adopts a pairwise ranking approach while drawing on state-of-the-art deep neural modeling techniques to analyze the semantics of queries and candidate facts along with their semantic connections. Unlike previous work, it can dynamically be applied to entirely new candidate facts without the need to compile knowledge graph statistics. In our experimental evaluation, we observe substantial improvements over previous work.

References

1. Bhowmik, R., de Melo, G.: Generating fine-grained open vocabulary entity type descriptions. In: Proceedings of ACL 2018 (2018)
2. Bhowmik, R., de Melo, G.: Be concise and precise: Synthesizing open-domain entity descriptions from facts. In: Proceedings of The Web Conference 2019. ACM (2019)
3. Cheng, G., Tran, T., Qu, Y.: RELIN: Relatedness and informativeness-based centrality for entity summarization. In: Proceedings of ISWC 2011. Springer (2011)
4. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: Pre-training of deep bidirectional transformers for language understanding. In: Proc. of NAACL 2019 (2019)
5. Hasibi, F., Balog, K., Bratsberg, S.E.: Dynamic factual summaries for entity cards. In: Proceedings of SIGIR 2017. pp. 773–782. ACM (2017)
6. Hogan, A., Blomqvist, E., Cochez, M., d'Amato, C., de Melo, G., Gutierrez, C., Labra Gayo, J.E., Kirrane, S., Neumaier, S., Polleres, A., Navigli, R., Ngonga Ngomo, A.C., Rashid, S.M., Rula, A., Schmelzeisen, L., Sequeda, J., Staab, S., Zimmermann, A.: Knowledge graphs. ArXiv **2003.02320** (2020)
7. Thalhammer, A., Lasierra, N., Rettinger, A.: LinkSUM: Using Link Analysis to Summarize Entity Data. In: Proceedings of ICWE 2016. Springer (2016)
8. Thalhammer, A., Rettinger, A.: Browsing DBpedia entities with summaries. In: ESWC (Satellite Events). LNCS, vol. 8798, pp. 511–515. Springer (2014)
9. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. In: Advances in Neural Information Processing Systems, pp. 5998–6008. Curran Associates (2017)