Towards a Unified Supervised Approach for Ranking Triples of Type-like Relations

Mahsa S. Shahshahani[1], Faegheh Hasibi[2], Hamed Zamani[3], and Azadeh Shakery[1]

[1] School of ECE, College of Engineering, University of Tehran, Iran
[2] Norwegian University of Science and Technology, Trondheim, Norway
[3] Center for Intelligent Information Retrieval, University of Massachusetts Amherst, USA
{ms.shahshahani,shakery}@ut.ac.ir
faegheh.hasibi@ntnu.no zamani@cs.umass.edu

Abstract. Knowledge bases play a crucial role in modern search engines and provide users with information about entities. A knowledge base may contain many facts (i.e., RDF triples) about an entity, but only a handful of them are of significance for a searcher. Identifying and ranking these RDF triples is essential for various applications of search engines, such as entity ranking and summarization. In this paper, we present the first effort towards a *unified supervised approach* to rank triples from various type-like relations in knowledge bases. We evaluate our approach using the recently released test collections from the WSDM Cup 2017 and demonstrate the effectiveness of the proposed approach despite the fact that no relation-specific feature is used.

Keywords: Knowledge bases, triple scoring, entity facts.

1 Introduction

Knowledge bases (KBs) are now a commodity in modern search engines and various semantic search systems. They are structured repositories of entities (such as people, locations, and organizations), where the knowledge about entities is stored in the form of ⟨*subject, predicate, object*⟩ triples, referred to as RDF triples. While knowledge bases contain a large amount of RDF triples about entities, only a handful of them might be of significance for a searcher. Ranking and scoring of these triples is a common step in various semantic search applications, such as entity summarization [5] and generating content for entity cards [3]. Consider for example RDF triples related to the *profession* of the entity *Oscar Wilde*, where the top-ranked profession can be displayed next to his name in the entity card. Another example application is incorporating triple scores for answering queries such as "german politicians", where the ideal answer should contain entities with politician and German as their primary *profession* and *nationality*, respectively.

The triple scoring task has been defined as "computing a score that measures the relevance of the statement expressed by the triple compared to other triples from the same relation" [1]. The task has been introduced by Bast et al. [1] and further received attention at the WSDM Cup 2017 [7]. It is specifically focused on the ranking of triples related to the type-like relations, i.e., the triples belonging to an abstract group or type. For example, considering *profession* as a type-like relation, the following scores can be obtained:

⟨*Oscar Wilde,*	*profession,*	*Playwright*⟩	1.0
⟨*Oscar Wilde,*	*profession,*	*Poet*⟩	0.4
⟨*John F. Kennedy,*	*profession,*	*Politician*⟩	1.0
⟨*John F. Kennedy,*	*profession,*	*Author*⟩	0.2

A number of supervised approaches have been proposed for the triple scoring task at the WSDM Cup 2017. In this paper, we argue that these approaches all suffer from a fundamental drawback: different feature sets are extracted for different relations. This is not a desired solution for many real-world scenarios. The reason is that knowledge bases contain a large number of type-like relations that makes extracting a separate set of features for each relation infeasible. This calls for a *unified* approach that can be used for every type-like relation. Designing a unified supervised approach for this task is the main motivation of this paper. To this aim, we propose a set of features that can be extracted irrespective of a specific type-like relation. We further train a learning to rank model based on the defined features. Our experiments on the WSDM Cup 2017 dataset suggest that although the proposed approach does not use relation-specific features, our approach performs on a par with the winners of the WSDM Cup. We further demonstrate that in addition to relation-independent features, a single model trained on triples of different relations can bring solid performance. In essence, a single model can suffice to achieve good performance, sometimes even better than the specific purpose-learned models, and this is due to the availability of more training data. We also study the importance of each defined feature in our experiments. This paper presents the first efforts on unified supervised approaches for the triple scoring task and we believe our findings can smooth the path towards effective methods for real-world scenarios.

2 A Unified Supervised Approach for Triple Ranking

Problem Statement. Let $T(e, r) = \{t_1, t_2, \cdots, t_n\}$ be the set of all triples from the type-like relation r for entity e, where each triple $t_i = \langle e, r, . \rangle$ denotes "a relation between an entity and an abstract group or type" [1]. For instance, ⟨*Abraham Lincoln, profession, Statesman* ⟩ is a triple from the type-like relation "profession". These relations can be extracted from knowledge bases, such as DBpedia and Freebase. The aim of the task is to *rank* or *score* the triples in a given set $T(e, r)$.

Approach. A number of supervised ranking approaches have been proposed for this task in the WSDM Cup 2017 [2]. The main shortcoming of these approaches is that they are relation-dependent. Therefore, these approaches can only be applied to a single relation. While there is a general consensus that supervised methods outperform unsupervised ones, no general supervised method has been proposed to date for the triple scoring task; only a set of unsupervised approaches has been proposed in [1]. This motivates us to propose a *unified* supervised approach, which is necessary for real-world applications, such as search engines. Our approach is to use a learning to rank framework with relation-independent features, which can be used for all type-like relations in a knowledge base.

Features. In total, we defined a set of 14 features that can be extracted from each triple of type-like relations. The features are listed in Table 1. As shown in the table, the features extracted for each triple $\langle e_1, r, e_2 \rangle$ are all relation-independent, i.e., independent of r, which is necessary for a unified approach. A number of these features have been used

Table 1: The relation-independent features extracted for a triple $\langle e_1, r, e_2 \rangle$ where e_1 and e_2 are two entities and r denotes a relation between e_1 and e_2.

ID	Feature	Description
1	inSent	Presence of e_2's name in the first sentence of e_1's Wikipedia document
2	inPar	Presence of e_2's name in the first paragraph of e_1's Wikipedia document
3	TF-Par	The TF of e_2's name in the first paragraph of e_1's Wikipedia document
4	EFirstPar	Is e_2 the first entity mentioned in the first paragraph of e_1's Wikipedia document or not.
5	#Rel	Total number of relations for e_1
6	#CmnWords	Number of common words between the first paragraph of the Wikipedia documents of e_1 and e_2
7	#UniqWords	Number of unique common words between the first paragraph of the Wikipedia documents of e_1 and e_2
8	LenPar$_2$	Length of the first paragraph of e_2's Wikipedia document
9	LenPar$_1$	Length of the first paragraph of e_1's Wikipedia document
10	PosSent	Position of e_2 in the first sentence of e_1's Wikipedia document
11	NormPosSent	Feature #10 / total number of relations for e_1
12	PosPar	Position of e_2 in the first paragraph of e_1's Wikipedia document
13	NormPosPar	Feature #12 / total number of relations for e_1
14	CosSim	Cosine similarity between the embedding vectors of the first paragraph of the Wikipedia documents of e_1 and e_2

in prior work [6]. The only assumption here is that there exists a short textual description, i.e., a paragraph, for each entity. In our experiments, we use the first paragraph of the Wikipedia documents corresponding to the entities. The feature set consists of term counting (e.g., feature $\#1$), semantic matching (e.g., feature $\#14$), graph information from the knowledge base (e.g., feature $\#5$), and hybrid features (e.g., feature $\#11$). The features are either binary or numerical. Note that in the first three features, each entity might have multiple names, e.g., US, USA, and United States, and appearance of at least one of them is sufficient. For the last feature, the average word embedding vector for all terms in the paragraph is calculated as the paragraph's embedding vector.

3 Experimental Setup

Data. We evaluate our approach using two type-like relations: profession and nationality. The dataset contains 1,387 triples (1,028 for profession and 359 for nationality). The entities and relations were extracted from Freebase. A relevance score from $\{0, 1, \cdots, 7\}$ has been assigned to each triple via crowdsourcing as described in [1]. This dataset has been used in the WSDM Cup 2017 [7]. Similar to the WSDM Cup setup, we use about half of these triples (i.e., 677 triples) as training set and the remaining part as the test set. Following [1], we show the generecity of our approach using the *profession* and *nationality* relations; experimenting with other type-like relations requires building new test collections, which is not the focus of this paper but is an obvious future direction.

To extract the features, all documents were stemmed using the Porter stemmer and were stopped using the standard INQUERY stopword list. We indexed the documents using the Lemur toolkit.[4] For feature $\#14$, we used the pre-trained embedding vectors

[4] See `https://www.lemurproject.org/lemur.php`.

with 300 dimensions learned by GloVe on Wikipedia dump 2014 plus Gigawords 5.[5] All features were normalized using $l2$ normalization. The parameters of the learning algorithms were set using 5-fold cross-validation over training set.

Evaluation Metrics. To evaluate our models, we consider two score-based metrics: average score difference (ASD) and accuracy. The former is calculated based on the average of the absolute difference between the predicted and the ground truth scores, while the latter is the ratio of predicted scores with the difference of at most 2 from the ground truth score. To be consistent with the runs submitted to the WSDM Cup 2017, the predicted scores should be integers, and thus the scores were rounded.

Since predicting accurate rankings, rather than scoring, might be the main objective in many applications (cf. Section 1), we also consider ranking-based metrics in our experiments. We use Kendall's τ distance and normalized discounted cumulative gain (NDCG) with three different cutoffs, i.e., 1, 3, and 5. NDCG@1 lets us know how accurate is the highest ranked triple, while the other ones demonstrate the quality of the generated ranked list. The Kendall's τ distance as well as the score-based metrics (ASD and accuracy) have been used in the WSDM Cup 2017 [7] as evaluation metrics. For accuracy and NDCG, higher values are better, while lower ASD and τ distance show better performance. Statistical significant differences between the results are determined using the paired t-test with 95% confidence interval.

4 Results and Discussion

We explored different learning to rank models including point-wise (linear regression and gradient boosting regression trees (GBRT)), pair-wise (AdaRank and RankBoost), and list-wise (ListNet and LambdaMART) ones. Although pair-wise and list-wise approaches often outperform point-wise models in many ranking scenarios, we observed that GBRT achieves the highest performance in our experiments. We attribute this to the limited amount of training data available for the task. The learning curve (see Fig. 1) also validates that the amount of training data provided by the WSDM Cup 2017 is not enough for training. In the following experiments, we focus on GBRT as a well-performing ranking model for our task. For the sake of space, we do not report the results achieved by different ranking models.

Comparison with baselines. In the next set of experiments, we compare our approach with the winners of the WSDM Cup 2017. The results are reported in Table 2. Bok-choy [4], Cress [6], and Goosefoot [8] respectively achieved the best accuracy, ASD, and the Kendall's τ distance at the WSDM Cup 2017. According to Table 2, our model outperforms all the models in terms of ASD and performs comparably in terms of the Kendall's τ distance. Our model also performs better than Cress and Goosefoot, in terms of accuracy. It is notable that all the baselines use relation-specific methods. These results suggest that although our approach only uses relation-independent features, the performance is still good compared to the winners of the WSDM Cup 2017, and even better in terms of ASD.

Single model vs. separate models for the relations. We show that we can achieve a solid performance by only using relation-independent features, but the models were trained separately for each relation. An important research question here is how does our model

[5] See `https://nlp.stanford.edu/projects/glove/`.

Table 2: Comparison with the top-performing systems at the WSDM Cup 2017.

Method	Acc	ASD	Kendall
Bokchoy [4]	**0.87**	1.63	0.33
Cress [6]	0.78	1.61	0.32
Goosefoot [8]	0.75	1.78	**0.31**
Our [GBRT]	0.80	**1.57**	0.32

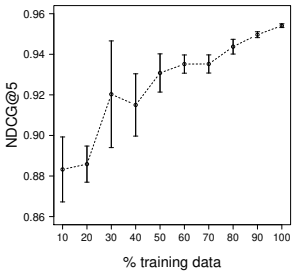

Fig. 1: Learning curve trained for all relations

Table 3: Performance of single model vs. separate models for different relations.

Model	Test relation	Acc	ASD	Kendall	NDCG@1	NDCG@3	NDCG@5
Separate Models	profession	0.79	1.55	0.27	0.8827	0.9288	0.9396
	nationality	0.85	1.64	0.39	0.8113	0.9453	0.9453
	total	0.80	1.57	0.32	0.8649	0.9394	0.9472
Single Model	profession	0.79	1.55	0.28	0.8947	0.9341	0.9465
	nationality	0.81	1.52	0.39	0.8858	0.9670	0.9670
	total	0.80	1.54	0.33	0.8905	0.9479	0.9547

perform if we do not train different models for different relations? To address this question, we combine the training data of the two relations and learn a single GBRT model, which is further used for both relations. Table 3 presents the results and signifies that our single generic model performs on a par with the ones trained for a single relation, in terms of accuracy, ASD, and τ. No significant difference is observed. Interestingly, although different relations may have different feature distributions, putting all of them together and training a single model lead to a better performance in terms of NDCG@k for different k. The reason is due to the limited amount of training data.

Learning curve. To analyze performance variations with different amounts of data, we create subsets of the whole training set, for 10 different sizes ranging from 10% to 100% of the instances. We repeat this random sampling process 10 times for each size. The learning curve in terms of NDCG@5[6] is plotted in Fig. 1. According to the figure, by increasing the amount of training data, the average performance increases and the standard deviation decreases. The learning curve is generally increasing and it does not get stable; which demonstrate that providing more training data would lead to even better ranking performance.

Feature analysis. To analyze the importance of each feature, we performed forward selection based on the Gini index and report the ranking metrics after selection of each feature. The results for both profession and nationality relations are plotted in Fig. 2. According to the plots, the paragraph lengths, semantic similarity based on word embeddings, and some term matching features are considered as the best features. The term matching features and semantic similarity are also among the best features for the nationality relation.

[6] For the sake of space, we only consider NDCG@5 as the evaluation metric in this experiment. The learning curves with respect to the other metrics, e.g., ASD, are also similar.

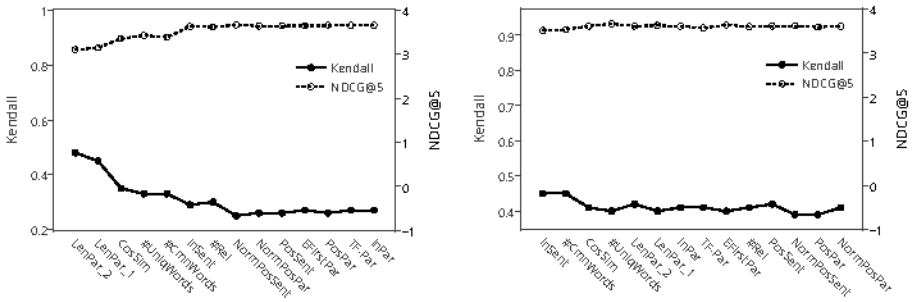

Fig. 2: Feature importance for profession (left) and nationality (right) relations.

5 Conclusions and Future Work

We proposed a *unified* supervised approach for ranking triples of type-like relations. Our approach is based on a set of relation-independent features and a learning to rank model. Our experiments on the WSDM Cup 2017 dataset suggested that good performance can be achieved using only relation-independent features. An interesting finding of the paper is that there is no need to train the model on different training sets for different relations, at least for the considered relations. Due to the lack of available data, we were only able to study two relations, which may not be sufficient to make general claims. This preliminary research magnifies the importance of developing relation-independent approaches and smooths the path towards studying unified approaches for the triple scoring task. Studying this task for several other relations will be the first step in our future work. Another avenue is to work on a unified semi-supervised approach for this task, as it is unlikely to have access to labeled training data for all relations.

Acknowledgements. This work was partially supported in part by the Center for Intelligent Information Retrieval. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the sponsors.

References

1. H. Bast, B. Buchhold, and E. Haussmann. Relevance scores for triples from type-like relations. In *SIGIR '15*, pages 243–252, 2015.
2. H. Bast, B. Buchhold, and E. Haussmann. Overview of the Triple Scoring Task at the WSDM Cup 2017. In *WSDM Cup*, 2017.
3. H. Bota, K. Zhou, and J. M. Jose. Playing Your Cards Right: The Effect of Entity Cards on Search Behaviour and Workload. In *CHIIR '16*, pages 131–140, 2016.
4. B. Ding, Q. Wang, and B. Wang. Leveraging text and knowledge bases for triple scoring: An ensemble approach - the bokchoy triple scorer at wsdm cup 2017. In *WSDM Cup*, 2017.
5. F. Hasibi, K. Balog, and S. E. Bratsberg. Dynamic factual summaries for entity cards. In *SIGIR '17*, pages 773–782, 2017.
6. F. Hasibi, D. Garigliotti, S. Zhang, and K. Balog. Supervised ranking of triples for type-like relations - the cress triple scorer. In *WSDM Cup*, 2017.
7. S. Heindorf, M. Potthast, H. Bast, B. Buchhold, and E. Haussmann. WSDM Cup 2017: Vandalism detection and triple scoring. In *WSDM '17*, pages 827–828, 2017.
8. V. Zmiycharov, D. Alexandrov, P. Nakov, I. Koychev, and Y. Kiprov. Finding people's professions and nationalities using distant supervision - the Goosefoot triple scorer. In *WSDM Cup*, 2017.