Re-thinking Knowledge Graph Completion Evaluation from an Information Retrieval Perspective

Ying Zhou
zhouying20@mails.ucas.ac.cn
University of Chinese Academy of
Sciences
Institute of Software, Chinese
Academy of Sciences
Beijing, China

Xuanang Chen
chenxuanang19@mails.ucas.ac.cn
University of Chinese Academy of
Sciences
Institute of Software, Chinese
Academy of Sciences
Beijing, China

Ben He*
benhe@ucas.ac.cn
University of Chinese Academy of
Sciences
Institute of Software, Chinese
Academy of Sciences
Beijing, China

Zheng Ye*
yezheng@scuec.edu.cn
South-Central University for
Nationalities
Wuhan, China

Le Sun
sunle@iscas.ac.cn
Institute of Software, Chinese
Academy of Sciences
Beijing, China

ABSTRACT

Knowledge graph completion (KGC) aims to infer missing knowledge triples based on known facts in a knowledge graph. Current KGC research mostly follows an entity ranking protocol, wherein the effectiveness is measured by the predicted rank of a masked entity in a test triple. The overall performance is then given by a micro(-average) metric over all individual answer entities. Due to the incomplete nature of the large-scale knowledge bases, such an entity ranking setting is likely affected by unlabelled top-ranked positive examples, raising questions on whether the current evaluation protocol is sufficient to guarantee a fair comparison of KGC systems. To this end, this paper presents a systematic study on whether and how the label sparsity affects the current KGC evaluation with the popular micro metrics. Specifically, inspired by the TREC paradigm for large-scale information retrieval (IR) experimentation, we create a relatively "complete" judgment set based on a sample from the popular FB15k-237 dataset following the TREC pooling method. According to our analysis, it comes as a surprise that switching from the original labels to our "complete" labels results in a drastic change of system ranking of a variety of 13 popular KGC models in terms of *micro* metrics. Further investigation indicates that the IR-like macro(-average) metrics are more stable and discriminative under different settings, meanwhile, less affected by label sparsity. Thus, for KGC evaluation, we recommend conducting TREC-style pooling to balance between human efforts and label completeness, and reporting also the IR-like *macro* metrics to reflect the ranking nature of the KGC task.

_____
*Corresponding author

CCS CONCEPTS

• **Computing methodologies → Knowledge representation and reasoning**; **Semantic networks**; • **General and reference → Evaluation**.

KEYWORDS

Knowledge Graph Completion; Link Prediction; Entity Ranking; TREC Pooling; Evaluation; Label Sparsity

1 INTRODUCTION

Knowledge graphs (KGs) are collections of real-world factual triples, wherein each triple is often presented as a subject *s*, a predicate *p*, and an object *o*. The rich information leveraged by KGs has become more and more important to various natural language processing (NLP) tasks, such as question answering [56], dialogue system [22], or text generation [26]. Meanwhile, the inherent incompleteness of KGs has motivated active research to automatically complement the missing links by knowledge graph completion (KGC). A general KGC approach is to embed entities and relations into a vector space, then assign every possible triple a score to measure the plausibility of its existence. The *entity ranking* evaluation protocol is usually adopted to assess the performance of KGC models. Specifically, using the entity ranking protocol, every test triple is converted to two triple questions: *i.e.,* (?, *p, o*) and (*s, p,* ?), where the masked head or tail entity is the prediction target. All candidate entities are then ranked by likelihood of being the masked true entity. Precision of all these masked true entities, e.g., reciprocal rank, are finally averaged to evaluate the performance. Note that a triple question could correspond to more than one answers, e.g., a question (*Trump, visited,* ?) has multiple answers like *China, Japan,* and *India.* The

evaluation uses answer-wise micro(-average) metrics such that each answer entity contributes equally to the final performance.

As KGC can be seen as essentially a ranking task with question answering nature [51], it shares many similarities with document ranking in information retrieval (IR). However, current KGC evaluation is largely based on extremely sparse labels with micro metrics, unlike in IR evaluation where top-ranked documents are usually densely labeled, and the performance estimation is macro-averaged over individual queries. Such incompatibility has motivated this work to re-think KGC evaluation from an IR perspective. Actually, as current KGC datasets are all derived from incomplete knowledge bases, there exist inevitably a large quantity of unlabelled positive triples that lead to biased estimation of model performance [51]. Given the large sizes of existing knowledge bases, it is unfeasible in practice to manually annotate all unobserved positive triples. Hence, label sparsity indeed exists in current KGC evaluation, but its impact remains to be investigated. To bridge this gap, inspired by the TREC pooling method [49], this paper presents a comprehensive study on label sparsity in KGC evaluation. TREC pooling is a standard evaluation framework typically employed in the TREC experimentation for IR research on large-scale datasets, in which the top-ranked results returned by a diverse set of systems are manually annotated to make up for the label incompleteness. Specifically, this work manually labels a subset of the widely-used FB15k-237 dataset [45], coined as `FB-Test-S-C`. Our `FB-Test-S-C` set is composed of 22,492 triples pooled from the top-10 ranked candidates for 1,023 sampled test triples returned by six diverse KGC models, mimicking a relatively complete human judgement set.

After constructing this relatively "complete" `FB-Test-S-C` test set, comprehensive analysis is conducted to answer the following three research questions. **RQ1: Does label sparsity indeed affect the KGC evaluation?** According to the re-evaluation results obtained on `FB-Test-S-C`, drastic changes in system ranking are observed after adding the supplemental labeled triples to the original KGC test set. The results also raise the concern whether the widely used answer-wise micro metrics are sufficient for the KGC evaluation. Thereby, we introduce the IR-like question-wise macro metrics by treating triple questions as queries, and entity candidates as documents. With this evaluation setting, another question arises: **RQ2: How do the macro metrics differ from the micro metrics?** Following recent research on IR evaluation [6, 37, 39], we comprehensively compare micro and macro metrics in this study. As shown by the results, the macro metrics are less affected by label sparsity, and are less sensitive to change of test data, while exhibiting better ability in differentiating between different model performances. Based on all above observations in RQ1 and RQ2, for another question **RQ3: What evaluation protocol is suitable for KGC evaluation?**, we suggest to employ a TREC-style pooling for reducing performance biases caused by label sparsity, and additional to the widely-used micro metric, report IR-like macro metrics to reflect the per-question answering performance. In summary, the main contributions of this paper are three-fold:

- A relatively "complete" annotated test set based on the widely-used FB15k-237 dataset is constructed following the TREC pooling method. The data and related resources are available online[1].

---

[1]https://github.com/zhouying20/kgc-sparsity

- Extensive experiments on our "complete" test set show that there is a non-trivial impact of label sparsity on the KGC evaluation. The findings suggest the necessity in conducting TREC-pooling for KGC evaluation in order to negate the impact of label sparsity with acceptable manpower cost.
- Comprehensive analyses on the data sensitivity and discriminative power of evaluation metrics indicate the necessity of reporting macro metrics to reflect the per-question performance.

Remainder of this paper is organized as follows. Section 2 introduces the research background of this study. Section 3 describes our construction procedure of the densely annotated `FB-Test-S-C` dataset using the TREC pooling method. Based on this `FB-Test-S-C` set, analyses in Section 4 show non-negligible impact of label sparsity on the current KGC evaluation. In Section 5, analyses on the stability and discriminative power of evaluation metrics suggest the need for reporting macro metrics. Section 6 recaps the related works, followed by the concluding remarks in Section 7.

2 PRELIMINARIES

Knowledge Graph Completion (KGC). Given a set of entities $\mathcal{E}$ and a set of relations $\mathcal{R}$, a Knowledge Graph (KG) is a set of fact triples, namely, $\mathcal{K} = \{(s, p, o) \mid s, o \in \mathcal{E}, p \in \mathcal{R}\} \subseteq \mathcal{E} \times \mathcal{R} \times \mathcal{E}$, where s, p, o stand for subject (entity), predicate (relation) and object (entity), respectively. The KGC task aims to infer missing triples based on the known facts in the knowledge graph $\mathcal{K}$. To evaluate a KGC model, the knowledge graph is usually split into training, validation, and test sets, namely, $\mathcal{K} = \mathcal{K}_{train} \cup \mathcal{K}_{valid} \cup \mathcal{K}_{test}$ and $\mathcal{K}_{train} \cap \mathcal{K}_{valid} \cap \mathcal{K}_{test} = \emptyset$. A general framework for KGC consists of an entity/relation mapper and a scoring function, the former maps the entities and relations into hidden embeddings, and the latter assigns scores to each possible candidate. Up to now, KGC models can be divided into three categories, namely, translational models (e.g., [10, 42]), decomposition models (e.g., [20, 31]), and neural network models (e.g., [14, 18]). Unified frameworks like PyKEEN [3] and LibKGE [35] are provided to reproduce the previously published results and investigate the impact of hyper-parameters such as loss functions and negative sampling strategies.

Entity Ranking with Micro Metric. KGC models are typically evaluated under the entity ranking (aka. link prediction) protocol, wherein the goal is to find the missing target entity given a *triple question* consisting of a source entity and a relation. KGC models rank all entities in the entity set $\mathcal{E}$ to seek the true object o in $(s, p, ?)$ or the true subject s in $(?, p, o)$. To avoid misleading evaluation, a relaxed *filtered* setting [10] is usually adopted, wherein all observed true triples in the graph $\mathcal{K}$ are removed from candidate entities except the one being evaluated, namely, removing all (s, p, o') where $o' \neq o$ and $(s, p, o') \in \mathcal{K}$ when determining the rank of triple (s, p, o). KGC evaluation usually adopts the *micro* metrics such as the mean rank (MR), mean reciprocal rank (MRR), or Hits@K over individual answers to reflect the performance of KGC models. Formally, given a true triple $(s, p, o) \in \mathcal{K}_{test}$, all candidate triples $\{(s, p, o') \mid o' \in \mathcal{E} \wedge (s, p, o') \notin \mathcal{K}\}$ are assigned a score by a KGC model, then $rank(o|s, p)$ is denoted for the *filtered* rank of the true object o among all candidate entities. If predicting a head entity, $rank(s|p, o)$ is defined similarly. Thus, micro metrics can be defined
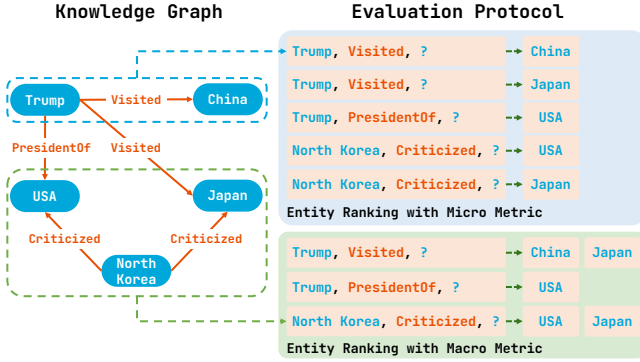
Figure 1: Illustration of entity ranking evaluation protocol with micro or macro metrics. Left side: a small knowledge graph with facts about Donald Trump and the USA. Blue box: commonly used entity ranking with micro metric that constructs questions from KG triples for answer-wise evaluations. Green box: entity ranking with macro metric that aggregates the same triple questions for IR-like question-wise evaluations.

uniformly as in Eq. 1 to estimate the per-answer performance.

$$\text{Micro Metric} = \sum_{p \in \mathcal{R}} \left(\frac{c_p}{|\mathcal{K}_{test}|} \cdot \frac{1}{2c_p} \cdot \sum_{(s,p,o) \in \mathcal{K}_{test}} \left(f(s|p,o) + f(o|s,p) \right) \right) \quad (1)$$

wherein c_p is defined as the frequency of occurrences of predicate (relation) p for further analysis on relation distribution in Section 4.2; and $f(x) = rank(x)$ for MR; $f(x) = \frac{1}{rank(x)}$ for MRR; $f(x) = 1$ if $rank(x) \leq$ K, else 0 for Hits@K. Hereinafter, we refer to them as Micro MR, Micro MRR, and Micro Hits@K, respectively.

Entity Ranking with Macro Metric. Additional to entity ranking with micro metric, we also introduce the common information retrieval evaluation methodology and macro metrics to entity ranking. The illustration of entity ranking with micro and macro metrics is shown in Figure 1. Analogously, in the entity ranking protocol with macro metric, the triple questions (same triple questions are aggregated) play the role of queries, and all target entities that answer the question act as the relevant documents. Specifically, with regard to a KG, all triple questions in the test set are aggregated and collected as a triple question set Q_{test}. For example, the triple question *(Trump, visited, ?)* is considered twice in the micro metrics, but they are aggregated and only considered once in the macro metrics. Then, for each aggregated question $q \in Q_{test}$, the corresponding target entities in the test set are considered relevant to form the answer set $\mathcal{A}_q$, while other entities are considered as unjudged (their relevance is unknown). When evaluating a KGC model, for each $q \in Q_{test}$, we rank all entities that have not been seen in the training and validation sets. Then, we use IR evaluation metrics as implemented in the standard trec_eval[2] toolkit that is widely-used for the TREC evaluation [49], including macro-average reciproal rank (MRR), MAP@20 [28], and nDCG@20 [23]. To be consistent with micro metrics, we also implement the widely-used Hits@K in a macro setting, wherein, for an aggregated triple question, we regard any answer ranked in top-K candidates as a hit. Similar to

[2]https://trec.nist.gov/trec_eval/

Table 1: Statistics of the original FB15k-237 test set (named FB-Test-O), the sampled subset (FB-Test-S), and the relatively "complete" test set (FB-Test-S-C). The original set does not have any negative triple. Our annotation finds more positive and many negative triples.

Test Set	#Triples	#Positives	#Negatives
FB-Test-O	20,466	20,466	0
FB-Test-S	1,023	1,023	0
FB-Test-S-C	22,492	2,738	19,754

micro metrics in Eq. 1, we also define macro metrics uniformly as in Eq. 2.

$$\text{Macro Metric} = \frac{1}{|Q_{test}|} \sum_{q \in Q_{test}} f(q) \quad (2)$$

wherein $f(q) = 1$ if $min(rank(a|q)) \leq K, a \in \mathcal{A}_q$, else $f(q) = 0$ for Hits@K; $f(q) = \frac{1}{min(rank(a|q))}, a \in \mathcal{A}_q$ for MRR. To distinguish from the micro metrics, we refer to them as Macro MRR, Macro Hits@K. As for MAP@20 and nDCG@20, please refer to trec_eval for implementation details. To be consistent with previous works, in the following experiments, we employ *filtered* setting in both micro and macro evaluations, namely, for macro metric, we only rank candidates that are not observed in the training and validation sets; and for micro metric, we additionally remove other answer entities such that only the target answer entity is reserved.

3 DATA CONSTRUCTION

As mentioned, there is a sparsity issue in the current KGC datasets. Herein, in order to investigate the influence of label sparsity on the evaluation and comparison of various KGC systems, our work constructs a relatively "complete" test set by adding manually labeled triples into one existing "incomplete" test set from widely-used FB15k-237 dataset [45]. In this section, we first introduce the construction procedure of our "complete" test set, followed by data statistics and analyses on the created dataset.

3.1 Annotation Procedure

We introduce the annotation methodology based on the TREC pooling method in this section. Recent works [14, 18, 35, 48, 60] on entity ranking for KGC almost all adopt two datasets for the evaluation, namely, FB15k-237 [45] derived from Freebase [9], and WN18RR [18] derived from WordNet [19]. Between those two datasets, FB15k-237 [45] is larger, which comes with 20,466 test triples (denotes as FB-Test-O), covering a wide range of knowledge extracted from Freebase. We therefore base our annotation on this dataset. As supplementing judgement on this entire test set requires exhaustive manpower, we rather annotate based on a small but representative subset of FB-Test-O. Specifically, we randomly sample 5% test triples, which is 1,023 out of 20,466, to obtain a relatively smaller initial triple seed set, denoted as FB-Test-S. Meanwhile, the original relation distribution (actually only 0.0889 KL divergence between the sampled and original relation distribution) is maintained to avoid introducing data bias to this subset, which will be further analyzed in Section 3.2.

Table 2: Examples of our annotations. The original answers exist in the original FB15k-237 test set. Our annotations are carried out on both positive and <u>negative</u> entities.

| Triple Question	Original Answers	Annotations
(Shia LaBeouf, /people/person/profession, ?)	Actor	Voice Actor, Comedian, <u>Model</u>, <u>Television Director</u>
(European American, /people/ethnicity/languages_spoken, ?)	American English	Spanish Language, Danish Language, <u>Mandarin Chinese</u>
(United States of America, /location/location/contains, ?)	Valdosta, Concord	Manhattan, Boston, Chicago, <u>Chelsea</u>, <u>Marrakesh</u>
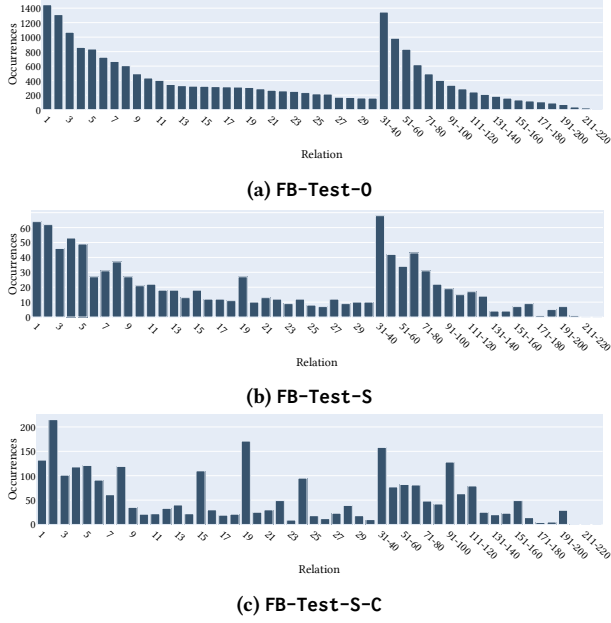
Figure 2: Relation distribution in the original FB15k-237 test set (FB-Test-O), sampled test subset (FB-Test-S), and human-labeled relatively complete test subset (FB-Test-S-C).

Pooling. For every seed triple in the FB-Test-S set, we collect its top-10 predicted triples from a series of KGC models to form the pool of triples waiting for judgement, namely, the top-10 ranked entities for both the head questions $(?, p, o)$ and the tail questions $(s, p, ?)$. To reduce the bias towards a specific KGC model, we select six diverse baseline systems with regard to their model architectures, including the translational models TransE [10] and RotatE [42], the decomposition models ComplEx [46], DistMult [55] and RESCAL [31], and the neural model ConvE [18], to obtain the less overlapped pooled triples. We use the well-tuned model checkpoints from LibKGE [35]. Finally, we obtain a pool of triples, consisting of 7,087 entities, 151 relations, and 21,469 triples.

Filtering Trivial Triples. To reduce the cost of human annotation, a certain number of trivial triples are automatically filtered, which are largely false triples: 1) the types of the subject and the object are not consistent with the relation; 2) the triples collected from head questions $(?, p, o)$ with obvious one-to-one and one-to-many relations; 3) the triples collected from tail questions $(s, p, ?)$ with obvious one-to-one and many-to-one relations. Note that we treat

all these selected trivial triples as negatives without the need for human annotation. Thereby, the final pool for later manually labeling contains 12,599 triples.

Annotating Triple Questions. Akin to [32, 36], we annotate both positive and negative triples in the pool. In order to facilitate the judgment of the annotator, we first manually create templates to transform triples to natural language questions, e.g., converting (*Donald Trump, /people/person/nationality, USA*) to "is Donald Trump's nationality USA?". During the annotation, annotators were asked to search for information on the Internet and answer Yes or No to each triple question. In particular, for the reliability of annotations, annotators were asked to provide the source link which contains the information for answering the question. Meanwhile, the provided source link should be in the list of verified data sources including Wikipedia and other domain-specific sources like All-Music, IMDB, etc. A total of five graduate students majored in computer science participated in the annotation. Each triple was labeled by two independent annotators. When there is a conflict, a third annotator steps in to provide the final annotation.

Finally, all pooled triples are annotated as positive (Yes) or negative (No), and this set of triples could be considered as a relatively "complete" test set, denoted as FB-Test-S-C. The statistics of these three test sets, namely, FB-Test-O, FB-Test-S, and FB-Test-S-C are summarized in Table 1. With adding supplemental annotated 21,469 triples into FB-Test-S set, the relatively "complete" FB-Test-S-C set contains much more judged triples, namely, 22,492 triples with 2,738 positives and 19,754 negatives. In other words, our annotations mainly complement the number of positive triples (which is directly related to the data sparsity problem), and also a large number of judged negative triples.

3.2 Analysis of Annotation

Annotated Examples. To ensure the quality of our annotations, we randomly selected 200 triplets from the newly annotated test set (half positives and half negatives) and gathered the relabeled results from an expert in the KG field. We found that 92% of the annotations are consistent with the pooled labels. Additionally, we present three selected annotated examples in Table 2. As seen in the table, the examples demonstrate that the label sparsity indeed exists in the current FB15k-237 dataset. Take the triple question *(European American, /people/ethnicity/languages_spoken, ?)* for example, if two KGC models assign the same rank for the true answer '*American English*', while one ranks '*Mandarin Chinese*' higher and the other ranks '*Spanish Language*' higher, according to the current entity ranking protocol, since both '*Mandarin Chinese*' and '*Spanish Language*' are unjudged (considered as negative), these

two KGC models are given the same performance metric. However, intuitively, the KGC model that ranks the false negative 'Spanish Language' higher should be considered better on this triple question. Thereby, the evaluation on the test set with incomplete judgement could lead to a misleading comparison of KGC models.

Relation Distribution. As mentioned, the relation distribution in our sampled `FB-Test-S` set is maintained as that in the original `FB-Test-O` test set by random sampling. Herein, there are totally 224 different relations in the `FB-Test-O` set, with different frequencies. Besides, to better display in Figure 2, low frequency relations are merged, resulting in 50 relation classes. As shown in Figures 2a and 2b, the frequency of all relations in the `FB-Test-S` set are highly analogous to the original `FB-Test-O` set. Specifically, the KL divergence (KLD) between `FB-Test-S` and `FB-Test-O` is 0.0889, while the KLD between `FB-Test-O` and the full FB15k-237 set (including train, validation and test sets, whose relation distribution is not plotted for space reason) is 0.2789. Therefore, the relation distribution in the sampled `FB-Test-S` set is close enough to the original `FB-Test-O` set. Furthermore, we present the relation distribution of our labeled "complete" `FB-Test-S-C` set in Figure 2c. We can see that the overall distribution of `FB-Test-S-C` has slightly changed, resulting in a relatively larger KLD of 0.2386 to the `FB-Test-O` set. The influence of this changed relation distribution will be investigated in Section 4.2. Note that we can only add new positive entities to one-to-many relations, but not to one-to-one relations, which will also be further discussed in Section 4.2.

4 STUDY ON LABEL SPARSITY

In this section, to answer RQ1, we conduct experiments to investigate how the label sparsity affects the KGC evaluation. We first introduce the experimental settings in Section 4.1, followed by the results and analyses in Section 4.2, in which inconsistent system rankings are observed with the increase of label completeness.

4.1 Experimental Design

Akin to studies on label sparsity in IR evaluation [11, 12, 21, 25, 33, 58], we explore the influence of label sparsity on KGC evaluation by comparing the rankings of KGC systems on an "incomplete" test set and its relatively "complete" versions.

Datasets. As described in Section 3, we have constructed a relatively "complete" `FB-Test-S-C` test set based on the `FB-Test-S` set using the pooling technique. Herein, we carry out experiments on these two paralleled `FB-Test-S` and `FB-Test-S-C` test sets, to see the changes in the rankings of KGC systems. When constructing the `FB-Test-S-C` set, we observe that the amount of unobserved true triples actually consistently grows with the pooling depth. Thereby, additional to the full `FB-Test-S-C` test set, we also investigate the influence of the different levels of judgement completeness to the KGC evaluation, by controlling the pooling depth.

Metrics. As described in Section 2, in the entity ranking evaluation, previous works usually use the micro metrics to report the per-answer performance of KGC models. Thereby, we mainly focus on micro metrics in this part to examine how label sparsity affects comparisons of KGC models. Herein, five commonly reported micro metrics, including `Micro Hits@1`, `Micro Hits@3`, `Micro Hits@10`, `Micro MR`, and `Micro MRR`, are used to evaluate and rank a series
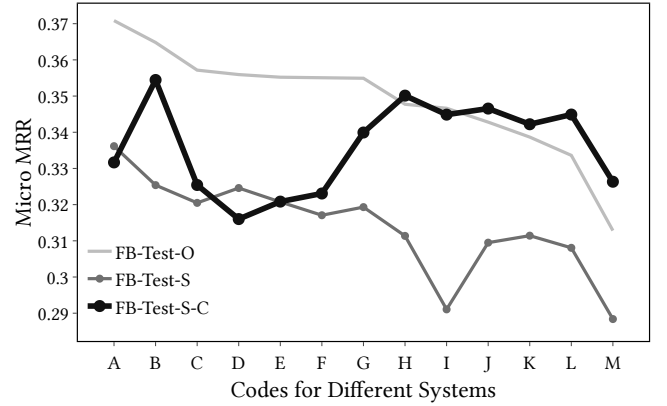

Figure 3: Effect of label sparsity. Micro MRR values of different KGC models obtained on the original sparse labels (`FB-Test-O`) are in decreasing order. The τ correlation between `FB-Test-S` and the entire original test set `FB-Test-O` is 0.7949. The use of added dense labels (`FB-Test-S-C`) leads to obvious system ranking change with Kendall's τ of -0.2308 to `FB-Test-S`.

of KGC models. Meanwhile, to answer RQ3, comparisons between micro and macro metrics need to be carried out, so we also provide the results on macro metrics, including `Macro Hits@10` and `Macro MRR`. As for the indicator of the changes in ranking of KGC systems, we choose Kendall's τ [24] following previous works on IR evaluation [11, 12, 21, 25, 37, 38]. Kendall's τ is a statistical metric used to verify ordinal association between two measured quantities, ranging from -1 to $+1$, wherein $+1$ for totally matching and -1 for the opposite. In other words, smaller Kendall's τ indicates larger divergence in system rankings.

Models. We implement 13 popular KGC models as follows:

- **Translational models** treat relation as the translation from source entity to target entity, including TransE [10], RotatE [42], HAKE [60], and RotH [13].
- **Decomposition models** match latent semantics of entity and relation to measure the plausibility, including DistMult [55], RESCAL [31], ComplEx [46], TuckER [7], and BiQUE [20].
- **Neural models** leverage neural networks to map entity and relation embeddings into matching scores, including ConvE [18], CompGCN [48], InteractE [47], and HittER [14].

Herein, for part of these KGC models, including TransE, RotatE, DistMult, ComplEx, RESCAL, and ConvE, we directly use the well-tuned checkpoints from LibKGE [35], which can surpass the performance in the original papers. For the remaining seven KGC models, we reproduce their results using the released source code following the original papers. For each KGC model, we repeatedly train for 10 times with the same hyper-parameters as in the original papers but with different random seeds, and then choose the one with the best micro MRR on the corresponding validation set. According to our results, almost all models reach the performance reported in their original papers, and the maximum performance difference is only 0.002 in micro MRR.
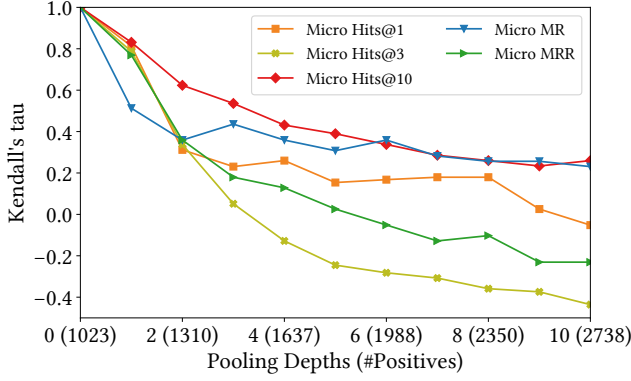
Figure 4: Correlations of system rankings in micro metrics between `FB-Test-S` set and `FB-Test-S-C` with different pooling depths. The system ranking changes rapidly when adding even a few labeled triples (e.g., pool depth of 2). The correlation coefficient even decreases to negative in terms of Hits@1, Hits@3 and MRR with increasing pooling depth.

4.2 Results

The system ranking drastically changes with the increased label completeness. As mentioned, we include 13 popular KGC models to empirically examine whether label sparsity indeed affects the evaluation of KGC models. The correlation between the system rankings obtained by the original `FB-Test-S` subset and by our annotated `FB-Test-S-C` with different pooling depths is summarized in Figure 4. It can be seen that, with the increase of label completeness (i.e., with larger pooling depth), the correlation of system rankings rapidly decreases for all micro metrics, which indicates a non-trivial change of system ranking brought by the use of more labels. Specifically, in terms of `Micro MRR`, the Kendall's τ drops from 1.0 to 0.359 with merely 287 added labeled positive triples at pooling depth of 2. Among these five micro metrics, in terms of `Micro Hits@1`, `Micro Hits@3`, and `Micro MRR`, the correlation of system rankings even drops to negative, while `Micro MR` and `Micro Hits@10` produce the most correlated results between "complete" and "incomplete" sets. A likely cause is that `Micro MR` is defined as the rank of true entity, which is less sensitive to the top-ranked true answers but is more dominated by the bottom-ranked outliers. And `Hits@K` shows to be largely affected by the choice of K, wherein the most correlated results come with $K = 10$ and the worst correlation is produced with $K = 3$. Additional to correlation of system rankings, we present an example in Figure 3, wherein the Kendall's τ between the system rankings in `Micro MRR` obtained on the sparse and the pooled dense labels (depth=10) is only -0.2308. It can be seen that the system ranking dramatically changes when switching from the original sparse labels to our more complete labels. Models are coded in A-M since our focus is the evaluation protocol rather than benchmarking current methods. Notably, model A, the 1st on the `FB-Test-O` labels, is only ranked in the middle (the 8th) on the complete labels. Meanwhile, model D drops from the 4th to the 13th (the last), and model L moves from the 12th to the 4th. When taking a closer look at the top-10

Table 3: Correlations of system rankings by relation category between `FB-Test-S-C` and `FB-Test-S`.

Categories		Micro Average			
		MRR	Hits@1	Hits@3	Hits@10
Head	**1-1**	1.0	1.0	1.0	1.0
	1-N	0.0256	0.0748	0.3731	0.4665
	N-1	0.1026	0.0996	0.1782	0.2857
	N-N	-0.3333	-0.0645	-0.4698	-0.1177
Tail	**1-1**	1.0	1.0	1.0	1.0
	1-N	-0.1026	0.1353	0.1988	-0.1420
	N-1	0.5641	0.7114	0.6123	0.7371
	N-N	-0.1282	0.0260	-0.2746	0.3007

predictions for model L, we find that 12.5% are predicted positives, in which more than half (6.7%) are newly added annotations by the pooling. By contrast, only 5.5% out of 11.8% of the predicted positives are newly labeled for model A (5.6% out of 11.5% for model D). From these analyses, it is obvious that false negatives have critical impact on the system rankings. Indeed, these obvious changes of system ranking suggest the necessity of re-evaluating current KGC approaches on FB15K-237, the most commonly used dataset, with increased label completeness from TREC pooling.

Label sparsity does affect all relation categories, apart from one-to-one category. We further investigate the changes in system ranking by different relation categories when the label completeness is improved. The relations in KG can be commonly categorized as one-to-one (1-1), one-to-many (1-N), many-to-one (N-1), and many-to-many (N-N) according to the average number of objects per subject and the average number of subjects per object [52]. Moreover, each question is either a Head question that predicts the subject $(?, p, o)$ or a Tail question that predicts the object $(s, p, ?)$. As seen in Table 3, comparing the correlations of systems rankings from `FB-Test-S-C` to `FB-Test-S`, the system ranking is fixed on 1-1 category (correlation = 1), while changes drastically on N-1 (Head), 1-N (Tail) and N-N (Head and Tail) categories, where even negative correlation can be observed. Herein, the above results are due to the fact that the 1-1 category has no sparsity issue, while N-1 (Head), 1-N (Tail) and N-N (Head and Tail) categories had many true answers which are added by our annotations. Those added labels are indeed shown to have important influence on the evaluation. As a result, non-negligible changes of system ranking are observed. However, the situations on 1-N (Head) and N-1 (Tail) categories are a little different. In principle, both should have little change in the correlation since there is only one answer as given by the original labels, but the correlations on 1-N (Head) in terms of `Micro MRR` and `Micro Hits@1` drastically decrease to close to zero. As our annotations only add more true *tail* entities for 1-N questions and true *head* entities for N-1 questions, the difference on 1-N (Head) or N-1 (Tail) categories between `FB-Test-S-C` and `FB-Test-S` sets is only the scale (i.e., the number of associated labeled questions like $(?, p, o')$ where o' are newly labeled) of the triple questions. Thereby, the changes on 1-N (Head) and N-1 (Tail) categories could attribute to that *micro* metrics may not stable on the size of test set, which is further analyzed in Section 5.2.
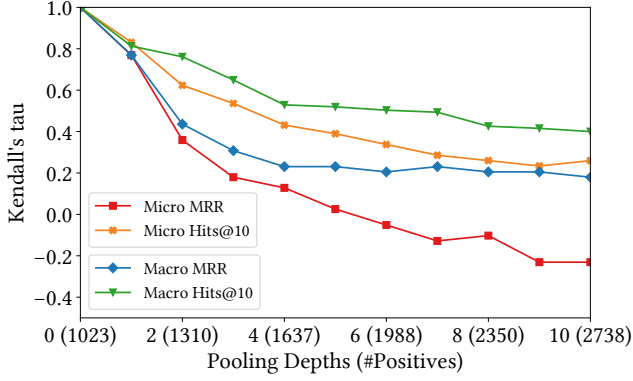
Figure 5: Correlations of system rankings between `FB-Test-S` and `FB-Test-S-C` with different pool depths using both *micro* and *macro* metrics. Comparing to micro metrics, the system rankings by macro metrics are more stable.

Table 4: Correlations between system rankings obtained on `FB-Test-S` and `FB-Test-S-C`. On `FB-Test-S-C` set, micro metrics are computed with different relation ratios derived from the relation distribution on `FB-Test-S-C` set (Complete) or `FB-Test-S` set (Sparse).

Distribution	MR	MRR	Hits@1	Hits@3	Hits@10
Complete	0.2308	-0.2308	-0.0519	-0.4358	0.2598
Sparse	0.2308	-0.2051	0.0256	-0.2051	0.3377

Macro metrics are more stable to label sparsity. With the adaption of IR-like evaluation, we can introduce macro metrics to the KGC task. To assess the influence of label sparsity on macro metrics, we compare the changes in system ranking between the micro and macro settings. Therefore, we choose MRR and `Hits@10` metrics to study the differences under micro or macro settings, as both are widely-used in previous works. The change curve of the correlation of system rankings under MRR and `Hits@10` metrics on `FB-Test-S-C` set with different pooling depths is shown in Figure 5. We can see that the correlations on all metrics decrease when the pooling depth increases, while macro metrics (namely, `Macro MRR` and `Macro Hits@10`) can achieve higher correlations than micro metrics (namely, `Micro MR` and `Micro Hits@10`) at any pooling depth. For example, when the pooling depth is 10, the gap between correlations of `Micro MRR` and `Macro MRR` is 0.41, and the gap for `Hits@10` is 0.14. These results are consistent with our previous observations in Section 4.2, namely, label sparsity indeed causes changes in the system ranking. But meanwhile, we can also see that macro metrics show the ability in negating the effect of label sparsity in that macro metrics produce relatively correlated system rankings between "complete" and sparse labels. Note that evaluation with micro metrics is still desirable since macro metrics may overemphasize one-to-one relations. Therefore, our suggestion is to report both *micro* and *macro* metrics to reflect both the per-answer and per-question performance.
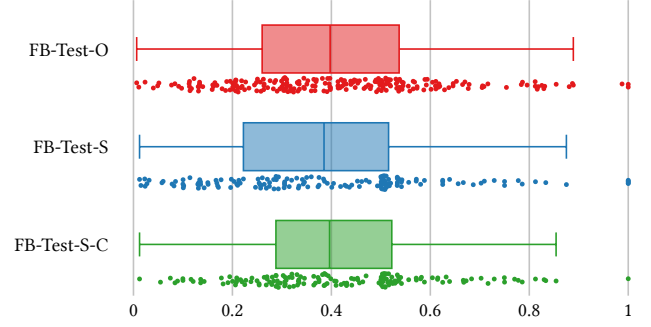

Figure 6: Performance of HittER [14] on different relations in Micro MRR. The median Micro MRR values on `FB-Test-O` and `FB-Test-S-C` sets are similar while is slightly smaller on `FB-Test-S` set.

Changes in relation distribution is not the main cause for the changes in system rankings. The added labels increase the proportion of the one-to-many, many-to-one, and many-to-many relations in the test set. As shown in Figure 2, the relation distribution of `FB-Test-S-C` slightly changes relative to that of `FB-Test-S` after complementing the judgment of missing triples, which raises the concern about whether the altered relation distribution is the main reason of the drastic changes of system ranking. Thereby, to answer this question, we examine the effect of relation distribution by using the relation ratio according to the `FB-Test-S` set when computing the final metrics. In other words, we try to maintain the ratios of relations in the final evaluation on `FB-Test-S-C`, namely, changing the $\frac{c_p}{|\mathcal{K}_{test}|}$ item in Eq. 1 to the corresponding occurrences ratios in the original `FB-Test-S` set. As seen in Table 4, although using original "sparse" distribution to compute the metrics, the system ranking still changes a lot, namely, the correlation coefficients for all micro metrics are still much less than 0.4. Accordingly, the results suggest that changes in relation distribution in the `FB-Test-S-C` test set do not contribute much to the changes in the system rankings.

Our supplemental annotated triples are not easy-to-predict. Previous work [36] has indicated that a reasonable test set should not be easy-to-predict. Herein, to verify the difficulty of our annotated triples, we use the state-of-the-art HittER model [14], to rank entities grouped by relations. Figure 6 shows the performance of HittER on different relations in terms of `Micro MRR`. We can see that the median micro MRR values on the `FB-Test-O` and `FB-Test-S-C` sets are much similar, while a slightly smaller median is produced on the sampled `FB-Test-S` set which may come from the sampling bias. Thereby, these three test sets are more or less of the same level of difficulty. In other words, our annotated triples are not easy-to-predict for a state-of-the-art KGC model, and the evaluation on our relatively "complete" `FB-Test-S-C` set is reasonable.

Answer to RQ1. Based on the observation on `FB-Test-S-C` set, we can conclude that label sparsity in KGC test set does affect the evaluation results of entity ranking regardless of micro or macro metrics are adopted. Even though, the macro metrics are shown to be more robust. In the next section, we further compare the use of
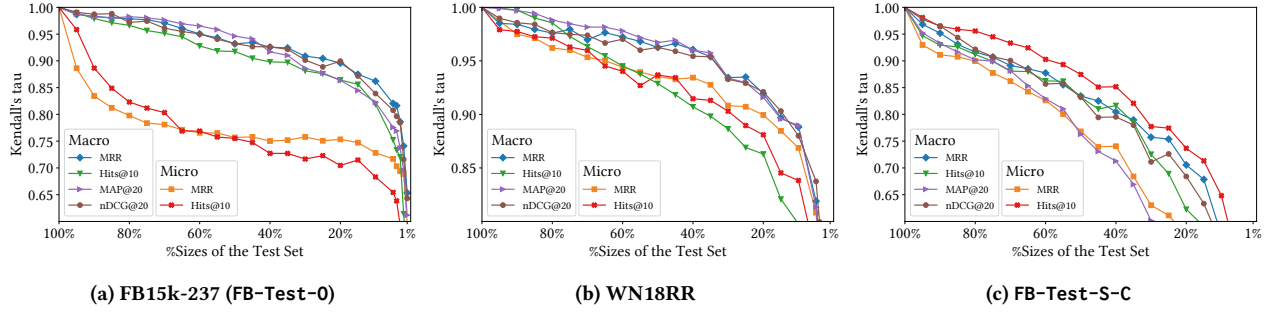
(a) FB15k-237 (FB-Test-0)

(b) WN18RR

(c) FB-Test-S-C

Figure 7: Correlations of system rankings on different sizes of test set relative to the whole original test set. Higher Kendall's τ indicates better stability to the change of test data.

macro and micro metrics to examine how they behave similarly or differ when used for KGC evaluation.

5 STUDY ON EVALUATION METRIC

The preceding Section 4.2 has shown that macro metrics are more stable to label sparsity than micro metrics. Since KGC is essentially a ranking task with question answering nature [51], question-wise macro metrics measure how well a KGC model answers a given question, unlike the micro metrics that measure how a specific answer is identified while ignoring other valid answers. As suggested by Sakai [37, 38], when choosing a metric for evaluation, it is expected to be stable and not sensitive to test data, namely, producing as consistent results as possible on different test data without losing the ability in discriminating different systems. Therefore, to answer RQ2 that how do macro metrics differ from micro metrics, especially that whether the introduction of the per-question macro metrics could be beneficial, we compare their stability and discriminative power in this section.

5.1 Experimental Design

Similar to the study on label sparsity in the previous section, analyses are carried out on the same 13 KGC models as described in Section 4.1. In addition to the original FB15k-237 test set and our annotated FB-Test-S-C set, we also use the popular WN18RR [18] dataset for the study on metrics in this section. As for the evaluation metrics, additional to Micro Hits@10, Micro MRR, Macro Hits@10 and Macro MRR, we also borrow two other macro metrics widely used in IR evaluation, MAP@20 [28] and nDCG@20 [23]. It should be noted that bpref [11] is not studied because the original FB15k-237 and WN18RR datasets do not come with any labeled negative triple.

Data Sensitivity. Similar to studies in IR evaluation [11, 12, 21, 33], we simulate a series of test subsets in different sizes (1%-95% of the total triples) by randomly discarding a proportion of triples from the test set. Then, we analyze the system rankings on each test subset using various metrics, and study how the system rankings change on these test subsets in different sizes, by computing the Kendall's τ between system rankings on test subsets and the whole test set. Besides, to eliminate the sampling bias, we repeat random discarding for 50 times on each sampling size, and report the average results as the final correlation.

Discriminative Power. When comparing the performance of two systems, statistical significance test is usually conducted. A significant difference (i.e., a p-value smaller than a given threshold) between two systems confirms that the better performance of one system over the other is not obtained by chance. Akin to Sakai [37], given a set of systems and a ranking metric, the discriminative power of this metric is defined as p-values on every system pair using a pairwise significance test. Herein, the paired two-tailed Student t-test is used in this work. Highly discriminative metrics are those who can obtain as many small p-values as possible [38, 39]. It should be noted that a highly discriminative metric does not always lead to correct conclusions since statistical significance is not necessarily correct [38], but we can still consider these metrics to be more sensitive to changes in model performance. Combined with the analysis on data sensitivity, we expect the results to give a clue about what evaluation paradigms and metrics are suitable for future KGC research.

5.2 Results

Macro metrics are more stable than micro metrics on incomplete test graph. The correlations of system rankings under different sample sizes are summarized in Figure 7. On FB15k-237 test set, as seen in Figure 7a, all macro metrics (including Macro MRR, Macro Hits@10, nDCG@20, and MAP@20) show strong stability across different sample sizes of test set, while micro metrics like Micro MRR and Micro Hits@10 produce very different system rankings on the FB15k-237 test set when the sample size decreases. Specifically, when the size of the sampled test set decreases to 80%, Kendall's τ values of Micro MRR and Micro Hits@10 are about 0.8, while almost all the τ values of the macro metrics are above 0.95. Then, as seen in Figure 7b, the stability results on WN18RR test set are a little different. We can see that micro metrics are relatively stable on different sizes of test set, likely due to the more "complete" nature of the domain specific knowledge graph WordNet, and containing more one-to-one relations in the test set. Even though, on WN18RR test set, slightly but consistently better stability of the macro metrics are still observed. Finally, according to Figure 7c, on our "complete" FB-Test-S-C set, it can be seen that the micro metrics are almost as stable as macro ones, while the Macro MRR is still relatively more stable, namely, the curve of Macro MRR is above its micro counterpart. Note that among all metrics on FB-Test-S-C,

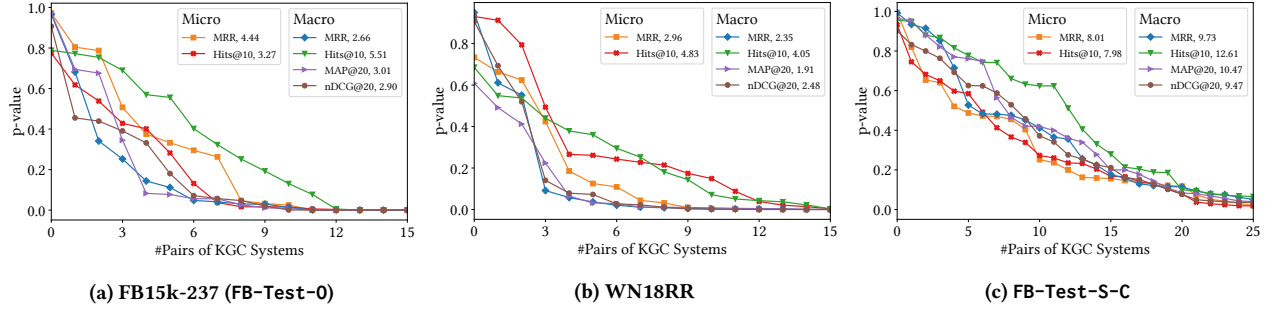(a) FB15k-237 (FB-Test-0)   (b) WN18RR   (c) FB-Test-S-C

Figure 8: Discriminative power analysis. The p-values between all pairs of KGC models are in descending order. Smaller area under the curve, as given next to the metric in the legend, indicates the relatively stronger discriminative power.

`Micro Hits@10` shows to be the most stable, suggesting that TREC-style KG augmentation indeed improves the evaluation stability of micro metrics. These above observations indicate that the stability of evaluation metric is related to the completeness of a KG test set, especially for the micro metrics. Despite the better stability macro metrics exhibit, as mentioned, it is necessary to take also the discriminative power of different metrics into consideration.

Both micro and macro metrics are discriminative, but macro metrics are better on incomplete test graph. The curves of p-values of all system pairs in a descending order on FB15k-237, WN8RR and `FB-Test-S-C` are summarized in Figure 8, wherein the smaller area under the curve of a metric indicates the relatively higher discriminative power of this metric. As shown in Figure 8, the main finding is that almost all macro metrics are more discriminative than micro metrics on incomplete test set (like FB15k-237), but the gap between micro and macro metrics decreases when the test set becomes "complete" (like `FB-Test-S-C`). As for `Hits@10`, we observe that it shows inconsistent changes in discriminative power between macro and micro settings compared with other metrics like `MRR`. We find that some triple questions from test set (especially on WN18RR) come with relatively bottom ranks for true entities, namely, the mean rank of all 13 models on WN18RR is around 3000 and the value on FB15k-237 and `FB-Test-S-C` is about 200. Thus, the micro and macro `Hits@10` may not be sensitive to bottom ranks since all answers with ranking larger than 10 are merely ignored. The same conclusion can also be drawn from Figure 7b, wherein both `Micro Hits@10` and `Macro Hits@10` show to be less stable than other metrics. Overall, results show that both micro and macro metrics are discriminative on a "complete" KGC test set, but micro metrics may less be capable of significantly differentiating the performance of KGC models on an incomplete test set.

Answers to RQ2&3. Based on the above analyses, for RQ2, we can see that the current micro metrics differ from macro metrics in that: (1) more affected by label sparsity; (2) more sensitive to change of test data; (3) not as sensitive to model performance as the macro metrics. Therefore, for RQ3, we suggest to employ TREC-style pooling to negate the impact of label sparsity. In addition to the widely used *micro* metrics, we recommend to report also the *macro* metrics to reflect both per-answer and per-question KGC performance. Moreover, on datasets with relatively complete labels

(e.g., our labeled pool), or with dominating one-to-one relations (e.g., WN18RR), since micro metrics behave similarly to macro metrics, we recommend to report only *macro* metrics for better stability, discriminative ability, and less sensitivity to label sparsity.

6 RELATED WORK

6.1 Problems behind Entity Ranking

Recently, quite a few works have raised the hidden problems behind entity ranking protocol. Akrami et al. [1, 2] conduct studies to analyze the influence of unrealistic triple whose the relation is an anomaly (near-reverse, near-same, or Cartesian product), and find that the current evaluations are much less compatible to reflect the model performance, and demonstrate that a re-investigation of possible effective approaches other than current entity ranking are indeed needed for following KGC researches. Sun et al. [43] argue that the way to select ranking position of a triple which has ties can drastically affect the model performance. Their main conclusion is that current entity ranking protocol is problematic in terms of data distribution and metric calculation, but they do not consider the label sparsity phenomenon. Meanwhile, recent studies [32, 41, 51] have revealed that the performance of KGC models may be misled by unobserved true (positive) triples, which are always treated as negatives in the current entity ranking protocol. Thereby, in spite of the advancements in effectiveness [13, 14], it also needed to investigate whether and how these false negative triples affect the evaluation of KGC systems, and whether the entity ranking protocol on the current "incomplete" test set corresponds to the actual prediction capability of KGC models. To the best of our knowledge, this paper is the first to systematically investigate the label sparsity issue in KGC evaluation.

6.2 Alternative Evaluation Protocols

Apart from the entity ranking protocol, other evaluation protocols have been proposed for the KGC task. A few works [10, 50, 52] experiment with the triple classification protocol and find it easy to get a high classification accuracy on sampled negative triples. However, Safavi and Koutra [36] gather a novel KGC dataset with human-labeled hard negatives, suggesting that triple classification is far from being solved. Moreover, Pezeshkpour et al. [32] redefine the triple classification task and introduce YAGO3-TC dataset with a similar TREC-style annotation method, but only 2 baselines were

used to pool triples. Besides, Wang et al. [51] propose Pair-entity Ranking (PR) wherein a weighted relation-averaged MAP on every single predicate p is used, and the weight of p is based on the total number of the occurrences in the test set, namely, predicting all possible triples given (?, p, ?). Mei et al. [30] present Max-K criterion for entity ranking that ask KGC models to give at most K candidates as the predicted answers for each question, wherein macro averaged recall, precision, and F1 are calculated to measure the model performance. Notably, our work shares common views on the definition of macro metrics with [30], but differs in that Mei et al. [30] apply macro metrics in a Max-K setting, while our work mainly compares the use of macro and micro metrics for entity ranking evaluation over label sparsity, data sensitivity, and discriminative power. Meanwhile, other works focus on refining or extending the entity ranking evaluation by proposing new evaluation metrics. Tiwari et al. [44] present the weighted geometric mean rank (WMR) that considers the difficulty of ranking for each relation, and argue that anomaly, test data distributions, and lacking of statistical significant test in entity ranking can hinder the understanding of the accuracy of current models. Berrendorf et al. [8] propose the adjusted mean rank (AMR) that adjusts the MR for chance. The main difference between these work with ours is that we aim to investigate label sparsity in the current entity ranking protocol, rather than proposing a new evaluation protocol.

6.3 Judgement Incompleteness in IR Evaluation

As manually labeling all query-document pairs is often infeasible, IR evaluation has long been affected by the label sparsity problem. Apart from web search engines that infer relevance from huge amount of query logs, e.g., by the click rate per impression [28], IR research has advanced from the Cranfield paradigm [17] to adopt the TREC pooling method [49] to balance between label completeness and manpower cost [5, 33]. Specifically, for a given set of test queries that requires human annotations, the top-k documents returned by a diverse set of IR systems are merged into the so-called TREC pool, where k is the pooling depth. Relevance assessments are then conducted on this pool to produce the final human labels. Although TREC pools are still incomplete given the likely existence of many unlabelled positive documents in large-scale collections, they have been shown to be reliable in comparing IR systems using early precision metrics like (macro) MAP@20. The TREC pools have been used as the "complete" labels sets in many studies on IR evaluation, especially on the label sparsity [5, 11, 12, 21, 25, 33].

7 CONCLUSIONS

In this paper, we systematically study whether and how the label sparsity affects the KGC evaluation. We first construct a relatively "complete" KGC test set by manual annotations following the TREC pooling method. According to our analysis using 13 popular KGC models, the system ranking drastically changes with the increase of label completeness, demonstrating a strong impact of label sparsity. Then, we further introduce IR-like macro-average metrics into the evaluation of KGC models, and experiments demonstrate that macro metrics are less sensitive to label sparsity and more stable and discriminative under different settings. Thus, we recommend reporting both micro and macro metrics to reflect the different

aspects of the KGC task, while employing TREC-style pooling to deal with the label incompleteness. As label sparsity exists in quite a few ranking-oriented tasks such as multi-label classification [27, 53], label distribution learning [54], and entity linking [34], our findings may also suggest the need for TREC pooling in a wider range of research in order to create relatively complete datasets with limited human efforts. In future work, we plan to extend our study to other IR metrics like bpref [11] and infAP [57] for KGC evaluation. Other factors over meta-evaluation on metrics will also be examined, such as pairwise discriminative power [16], user satisfaction [15, 29, 59], and diversity [4, 40].

REFERENCES

[1] Farahnaz Akrami, Lingbing Guo, Wei Hu, and Chengkai Li. 2018. Re-evaluating Embedding-Based Knowledge Graph Completion Methods. In *CIKM*. ACM, 1779–1782.

[2] Farahnaz Akrami, Mohammed Samiul Saeef, Qingheng Zhang, Wei Hu, and Chengkai Li. 2020. Realistic Re-evaluation of Knowledge Graph Completion Methods: An Experimental Study. In *SIGMOD Conference*. ACM, 1995–2010.

[3] Mehdi Ali, Max Berrendorf, Charles Tapley Hoyt, Laurent Vermue, Mikhail Galkin, Sahand Sharifzadeh, Asja Fischer, Volker Tresp, and Jens Lehmann. 2020. Bringing Light Into the Dark: A Large-scale Evaluation of Knowledge Graph Embedding Models Under a Unified Framework. *CoRR* abs/2006.13365 (2020).

[4] Enrique Amigó, Damiano Spina, and Jorge Carrillo de Albornoz. 2018. An Axiomatic Analysis of Diversity Evaluation Metrics: Introducing the Rank-Biased Utility Metric. In *SIGIR*. ACM, 625–634.

[5] Negar Arabzadeh, Alexandra Vtyurina, Xinyi Yan, and Charles L. A. Clarke. 2021. Shallow pooling for sparse labels. *CoRR* abs/2109.00062 (2021).

[6] Azin Ashkan and Donald Metzler. 2019. Revisiting Online Personal Search Metrics with the User in Mind. In *SIGIR*. ACM, 625–634.

[7] Ivana Balazevic, Carl Allen, and Timothy M. Hospedales. 2019. TuckER: Tensor Factorization for Knowledge Graph Completion. In *EMNLP/IJCNLP (1)*. Association for Computational Linguistics, 5184–5193.

[8] Max Berrendorf, Evgeniy Faerman, Laurent Vermue, and Volker Tresp. 2020. On the Ambiguity of Rank-Based Evaluation of Entity Alignment or Link Prediction Methods. *arXiv preprint arXiv:2002.06914* (2020).

[9] Kurt D. Bollacker, Colin Evans, Praveen K. Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In *SIGMOD Conference*. ACM, 1247–1250.

[10] Antoine Bordes, Nicolas Usunier, Alberto García-Durán, Jason Weston, and Oksana Yakhnenko. 2013. Translating Embeddings for Modeling Multi-relational Data. In *NIPS*. 2787–2795.

[11] Chris Buckley and Ellen M. Voorhees. 2004. Retrieval evaluation with incomplete information. In *SIGIR*. ACM, 25–32.

[12] Stefan Büttcher, Charles L. A. Clarke, Peter C. K. Yeung, and Ian Soboroff. 2007. Reliable information retrieval evaluation with incomplete and biased judgements. In *SIGIR*. ACM, 63–70.

[13] Ines Chami, Adva Wolf, Da-Cheng Juan, Frederic Sala, Sujith Ravi, and Christopher Ré. 2020. Low-Dimensional Hyperbolic Knowledge Graph Embeddings. In *ACL*. Association for Computational Linguistics, 6901–6914.

[14] Sanxing Chen, Xiaodong Liu, Jianfeng Gao, Jian Jiao, Ruofei Zhang, and Yangfeng Ji. 2021. HittER: Hierarchical Transformers for Knowledge Graph Embeddings. In *EMNLP (1)*. Association for Computational Linguistics, 10395–10407.

[15] Ye Chen, Ke Zhou, Yiqun Liu, Min Zhang, and Shaoping Ma. 2017. Meta-evaluation of Online and Offline Web Search Evaluation Metrics. In *SIGIR*. ACM, 15–24.

[16] Zhumin Chu, Jiaxin Mao, Fan Zhang, Yiqun Liu, Tetsuya Sakai, Min Zhang, and Shaoping Ma. 2021. Evaluating Relevance Judgments with Pairwise Discriminative Power. In *CIKM*. ACM, 261–270.

[17] Cyril W. Cleverdon. 1991. The Significance of the Cranfield Tests on Index Languages. In *SIGIR*. ACM, 3–12.

[18] Tim Dettmers, Pasquale Minervini, Pontus Stenetorp, and Sebastian Riedel. 2018. Convolutional 2D Knowledge Graph Embeddings. In *AAAI*. AAAI Press, 1811–1818.

[19] Christiane Fellbaum (Ed.). 1998. *WordNet: An Electronic Lexical Database*. A Bradford Book, Cambridge, MA, USA.

[20] Jia Guo and Stanley Kok. 2021. BiQUE: Biquaternionic Embeddings of Knowledge Graphs. In *EMNLP (1)*. Association for Computational Linguistics, 8338–8351.

[21] Ben He, Craig Macdonald, and Iadh Ounis. 2008. Retrieval sensitivity under training using different measures. In *SIGIR*. ACM, 67–74.

[22] He He, Anusha Balakrishnan, Mihail Eric, and Percy Liang. 2017. Learning Symmetric Collaborative Dialogue Agents with Dynamic Knowledge Graph Embeddings. In *ACL (1)*. Association for Computational Linguistics, 1766–1776.

[23] Kalervo Järvelin and Jaana Kekäläinen. 2002. Cumulated gain-based evaluation of IR techniques. *ACM Trans. Inf. Syst.* 20, 4 (2002), 422–446.

[24] M. G. Kendall. 1938. A New Measure of Rank Correlation. *Biometrika* 30, 1/2 (1938), 81–93.

[25] Ömer Kirnap, Fernando Diaz, Asia Biega, Michael D. Ekstrand, Ben Carterette, and Emine Yilmaz. 2021. Estimation of Fair Ranking Metrics with Incomplete Judgments. In *WWW*. ACM / IW3C2, 1065–1075.

[26] Rik Koncel-Kedziorski, Dhanush Bekal, Yi Luan, Mirella Lapata, and Hannaneh Hajishirzi. 2019. Text Generation from Knowledge Graphs with Graph Transformers. In *NAACL-HLT (1)*. Association for Computational Linguistics, 2284–2293.

[27] Yongqiang Li, Baoyuan Wu, Bernard Ghanem, Yongping Zhao, Hongxun Yao, and Qiang Ji. 2016. Facial action unit recognition under incomplete data based on multi-label learning with missing labels. *Pattern Recognit.* 60 (2016), 890–900.

[28] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to information retrieval*. Cambridge University Press.

[29] Daniel McDuff, Paul Thomas, Nick Craswell, Kael Rowan, and Mary Czerwinski. 2021. Do Affective Cues Validate Behavioural Metrics for Search?. In *SIGIR*. ACM, 1544–1553.

[30] Jiajie Mei, Richong Zhang, Yongyi Mao, and Ting Deng. 2018. On Link Prediction in Knowledge Bases: Max-K Criterion and Prediction Protocols. In *SIGIR*. ACM, 755–764.

[31] Maximilian Nickel, Volker Tresp, and Hans-Peter Kriegel. 2011. A Three-Way Model for Collective Learning on Multi-Relational Data. In *ICML*. Omnipress, 809–816.

[32] Pouya Pezeshkpour, Yifan Tian, and Sameer Singh. 2020. Revisiting Evaluation of Knowledge Base Completion Models. In *AKBC*.

[33] Md. Mustafizur Rahman, Mücahid Kutlu, Tamer Elsayed, and Matthew Lease. 2020. Efficient Test Collection Construction via Active Learning. In *ICTIR*. ACM, 177–184.

[34] Henry Rosales-Méndez, Aidan Hogan, and Barbara Poblete. 2019. Fine-Grained Evaluation for Entity Linking. In *EMNLP/IJCNLP (1)*. Association for Computational Linguistics, 718–727.

[35] Daniel Ruffinelli, Samuel Broscheit, and Rainer Gemulla. 2020. You CAN Teach an Old Dog New Tricks! On Training Knowledge Graph Embeddings. In *ICLR*. OpenReview.net.

[36] Tara Safavi and Danai Koutra. 2020. CoDEx: A Comprehensive Knowledge Graph Completion Benchmark. In *EMNLP (1)*. Association for Computational Linguistics, 8328–8350.

[37] Tetsuya Sakai. 2006. Evaluating evaluation metrics based on the bootstrap. In *SIGIR*. ACM, 525–532.

[38] Tetsuya Sakai. 2021. Evaluating Evaluation Measures for Ordinal Classification and Ordinal Quantification. In *ACL/IJCNLP (1)*. Association for Computational Linguistics, 2759–2769.

[39] Tetsuya Sakai. 2021. On the Instability of Diminishing Return IR Measures. In *ECIR (1) (Lecture Notes in Computer Science, Vol. 12656)*. Springer, 572–586.

[40] Tetsuya Sakai and Zhaohao Zeng. 2019. Which Diversity Evaluation Measures Are "Good"?. In *SIGIR*. ACM, 595–604.

[41] Marina Speranskaya, Martin Schmitt, and Benjamin Roth. 2020. Ranking vs. Classifying: Measuring Knowledge Base Completion Quality. In *AKBC*.

[42] Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, and Jian Tang. 2019. RotatE: Knowledge Graph Embedding by Relational Rotation in Complex Space. In *ICLR (Poster)*. OpenReview.net.

[43] Zhiqing Sun, Shikhar Vashishth, Soumya Sanyal, Partha P. Talukdar, and Yiming Yang. 2020. A Re-evaluation of Knowledge Graph Completion Methods. In *ACL*. Association for Computational Linguistics, 5516–5522.

[44] Sudhanshu Tiwari, Iti Bansal, and Carlos R. Rivero. 2021. Revisiting the Evaluation Protocol of Knowledge Graph Completion Methods for Link Prediction. In *WWW*. ACM / IW3C2, 809–820.

[45] Kristina Toutanova and Danqi Chen. 2015. Observed versus latent features for knowledge base and text inference. In *Proceedings of the 3rd Workshop on Continuous Vector Space Models and their Compositionality*. Association for Computational Linguistics, Beijing, China, 57–66.

[46] Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard. 2016. Complex Embeddings for Simple Link Prediction. In *ICML (JMLR Workshop and Conference Proceedings, Vol. 48)*. JMLR.org, 2071–2080.

[47] Shikhar Vashishth, Soumya Sanyal, Vikram Nitin, Nilesh Agrawal, and Partha P. Talukdar. 2020. InteractE: Improving Convolution-Based Knowledge Graph Embeddings by Increasing Feature Interactions. In *AAAI*. AAAI Press, 3009–3016.

[48] Shikhar Vashishth, Soumya Sanyal, Vikram Nitin, and Partha P. Talukdar. 2020. Composition-based Multi-Relational Graph Convolutional Networks. In *ICLR*. OpenReview.net.

[49] Ellen M. Voorhees and Donna K. Harman. 2005. *TREC: Experiment and Evaluation in Information Retrieval (Digital Libraries and Electronic Publishing)*. The MIT Press.

[50] Yanjie Wang, Rainer Gemulla, and Hui Li. 2018. On Multi-Relational Link Prediction With Bilinear Models. In *AAAI*. AAAI Press, 4227–4234.

[51] Yanjie Wang, Daniel Ruffinelli, Rainer Gemulla, Samuel Broscheit, and Christian Meilicke. 2019. On Evaluating Embedding Models for Knowledge Base Completion. In *RepL4NLP@ACL*. Association for Computational Linguistics, 104–112.

[52] Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. 2014. Knowledge Graph Embedding by Translating on Hyperplanes. In *AAAI*. AAAI Press, 1112–1119.

[53] Baoyuan Wu, Siwei Lyu, and Bernard Ghanem. 2016. Constrained Submodular Minimization for Missing Labels and Class Imbalance in Multi-label Learning. In *AAAI*. AAAI Press, 2229–2236.

[54] Miao Xu and Zhi-Hua Zhou. 2017. Incomplete Label Distribution Learning. In *IJCAI*. ijcai.org, 3175–3181.

[55] Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. 2015. Embedding Entities and Relations for Learning and Inference in Knowledge Bases. In *ICLR (Poster)*.

[56] Michihiro Yasunaga, Hongyu Ren, Antoine Bosselut, Percy Liang, and Jure Leskovec. 2021. QA-GNN: Reasoning with Language Models and Knowledge Graphs for Question Answering. In *NAACL-HLT*. Association for Computational Linguistics, 535–546.

[57] Emine Yilmaz and Javed A. Aslam. 2006. Estimating average precision with incomplete and imperfect judgments. In *CIKM*. ACM, 102–111.

[58] Emine Yilmaz, Evangelos Kanoulas, and Javed A. Aslam. 2008. A simple and efficient sampling method for estimating AP and NDCG. In *SIGIR*. ACM, 603–610.

[59] Fan Zhang, Jiaxin Mao, Yiqun Liu, Xiaohui Xie, Weizhi Ma, Min Zhang, and Shaoping Ma. 2020. Models Versus Satisfaction: Towards a Better Understanding of Evaluation Metrics. In *SIGIR*. ACM, 379–388.

[60] Zhanqiu Zhang, Jianyu Cai, Yongdong Zhang, and Jie Wang. 2020. Learning Hierarchy-Aware Knowledge Graph Embeddings for Link Prediction. In *AAAI*. AAAI Press, 3065–3072.