Knowledge Graph Embeddings: Are Relation-Learning Models Learning Relations?

Andrea Rossi
Roma Tre University
andrea.rossi3@uniroma3.it

Antonio Matinata
Roma Tre University
ant.matinata@stud.uniroma3.it

ABSTRACT

Link Prediction (LP) is the task of inferring relations between entities in a Knowledge Graph (KG). LP is difficult, due to the sparsity and incompleteness of real-world KGs. Recent advances in Machine Learning have led to a large and rapidly growing number of *relation learning* models, from the seminal work of Bordes et al. [4] to the recent model in [2]. Despite the flurry of papers in this area, just a few datasets and evaluation metrics have emerged as *de facto* benchmarking criteria. In our work, we question the effectiveness of these benchmarks in establishing the state-of-the-art. The use of unreliable benchmarking practices can have hidden ethical implications, as it may yield distorted evaluation results and overall lead the research community into adopting ineffective design choices. To this end, we consider key desiderata of a benchmark formulated as specific questions relevant to the LP task, and provide empirical evidence to answer those questions. Our analysis shows that existing datasets and metrics fall short in capturing a model's capability of solving LP. Specifically, we show that a model can score very high by learning to predict facts about a small fraction of the entities in the training set. Our study provides a more robust evaluation direction for future research on relation learning models, stressing that understanding why LP models reach certain performances is a crucial step towards explaining predicted relations.

1 INTRODUCTION

Knowledge Graphs (KG) are structured representations of facts in the real world. In a KG, each *node* represents an entity, e.g. a person, a place or a concept; each *label* represents a relationship usable to link entities; each *edge* in the form ⟨ *subject*, *predicate*, *object* ⟩, represents a fact connecting entity *subject* with entity *object* through the relationship *predicate*. Examples of KGs are FreeBase [3], WikiData [20], DBPedia [1], Yago [16] and – in industry – Google KG [15] and Microsoft Satori [12]. Such KGs can contain even billions of facts, yet only a small subset of all the facts in the real world.

KG embeddings are a way of representing the components of a KG as vectors or matrices (*embeddings*) in a low-dimensional hyperspace, called *latent space*. Embeddings are computed by training a model on the KG data, and thus carry the semantic meaning of the original KG relations. In other words, given the embeddings of two elements, it should be possible to identify their semantic correlations.

Knowledge Graph Completion is the task of identifying missing edges (facts) in KGs, by either extracting them from external corpora, or inferring them from the ones already in the KG [11]. The latter approach, called Link Prediction (LP), typically requires defining a *scoring function* that estimates the plausibility of any fact. With the rise of machine learning techniques, this has naturally combined with the use of KG embeddings: on the one hand, in training phase, LP models learn entity and relationship embeddings that optimize the scores for the facts already contained in the KG; on the other hand, in prediction phase the scoring function is applied on the embeddings of subject, predicate and object of a fact to compute its plausibility.

LP models can be queried by providing a subject (or an object), e.g., "Barack Obama", and a predicate, e.g., "place of birth", representing a question of the form "What is Barack Obama's place of birth?". Answering such a query amounts to compute the score of each potential object with respect to the current subject and predicate, and to find which one yields the best value. That is, the answer to a LP query is a *ranking* of the KG entities by decreasing plausibility.

This approach for was explored in the seminal work by Bordes *et al.*, describing the `TransE` LP model [4]. `TransE` interprets relations as translations operating on low-dimensional embeddings of the entities. In just a few years, `TransE` has inspired dozens of new relation-learning systems (see [21] for a survey), and few datasets and metrics have emerged as a *de facto* benchmark.

In this paper we critically analyze current benchmarks for LP models. Our study is motivated by the observation that, in current datasets, less than 15% of entities cover more than 80% of facts. Such a skew casts doubts on the suitability of these benchmarks for evaluating LP models. Indeed, we have empirically observed that a model can achieve state-of-the-art scores by learning to predict facts about a tiny fraction of the entities with highest degree in the training set, which are also the most mentioned entities in the test set as well. As an informative example, in FB15k, the most commonly used among the LP datasets, the entity node with the highest degree is by far "United States", with $\approx$ 2% of all edges; the vast majority of the "nationality" facts in the test set refer to "United States" as well. Therefore a model that learns to predict U.S. citizens only can obtain results comparable or even better than one that attempts to learn the nationality relation in detail.

Our contribution. We argue that the research community is not best served by benchmarks that allow for such a discrepancy to go unnoticed. Therefore, we provide a constructive contribution towards the definition of more effective benchmarks for LP models: (*i*) we formulate some key questions to highlight some of the most desirable properties for LP benchmarks; (*ii*) we conduct an extensive experimental analysis to understand whether the currently employed benchmarks satisfy such properties or not, and why. In doing this, we also highlight the ethical implications potentially connected with the limitations of the current benchmarks.

2 BENCHMARKS FOR LP

In this section we describe the currently employed LP benchmarks; analogously to [19], we consider a benchmark as the

whole workload employed to evaluate competing systems, composed of both datasets and metrics.

The most popular datasets for LP benchmarking consist of facts sampled from the FreeBase [3] and WordNet [13] KGs. FreeBase is an open KG with billions of facts about millions of real world entities and thousands of different relationships. WordNet is a lexical KG whose entities are English words grouped by their sense, and whose edges describe relations among words. The main features of such datasets are described in Table 1.

The *FB15K* dataset has been extracted by the `TransE` authors selecting all facts containing the 100 most mentioned entities in FreeBase also featured in the Wikilinks database[1] (thus including their low-degree neighbors). Defining a full-fledged benchmarking workload was beyond their intentions; nonetheless, most of the approaches inspired by `TransE` have been evaluated against the same dataset and metrics, making them a de facto benchmark.

The *FB15K-237* dataset is a FB15K subset built by [17] after observing that FB15K suffers so much from *test leakage* that a simple model based on observable features can reach state-of-the-art performances on it. The authors only considered facts for the most occurring 401 relationships in FB15K, and filtered away those with implicitly same meaning or inverse meaning. In order to take away trivial facts, they also removed from validation and test sets any facts linking entities already connected in the training set. We note that is nonetheless a biased approach, as we cannot evaluate the ability of a model to learn useful patterns such as, for instance, $\langle x, father\_of, y \rangle$ entails $\langle y, child\_of, x \rangle$.

The *WN18* dataset, analogously to FB15K, was built by the authors of `TransE`. They used the WordNet ontology [13] as a starting point, and then filtered out over multiple iterations entities and relationships with too few mentions.

The *WN18-RR* dataset was built by [5] applying similar policies to [17], after performing further investigations on test leakage in FB15K and WN18.

	Entities	Relations	Triples		
			Train	Valid	Test
FB15K	14951	1345	483142	50000	50971
WN18	40943	18	141442	5000	5000
FB15k-237	14541	237	272115	17535	20466
WN18-RR	40943	11	86835	3034	3134

Table 1: Standard datasets for LP.

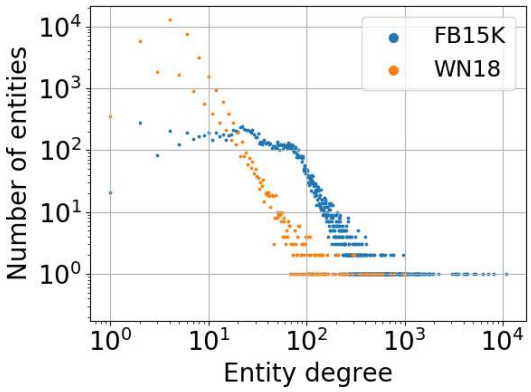
Datasets structural analysis. We define the *degree* of an entity and the *number of mentions* of a relationship as, respectively, the number of times that this entity or relationship is mentioned in different facts of a dataset.

Both FB15K and WN18 show severe skew in both entity degrees and relationship mentions. Figures 1(a) and 1(b) plot their distributions, showing that the large majority of entities (relationships) have a very low degree (number of mentions), whereas a small minority of them can reach massive representation.

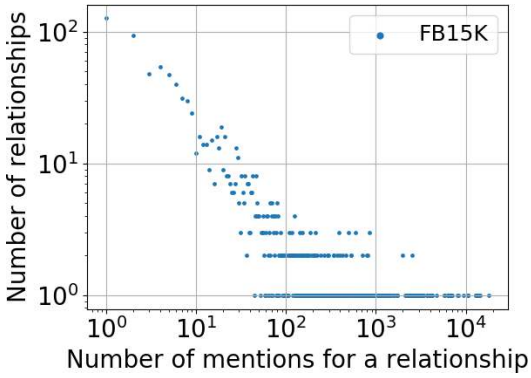
2.1 Metrics

Evaluation for LP models is typically performed on the task of Entity Prediction. Given the number of entities n, for any test fact $\langle s, p, o \rangle$: (i) s is removed from the triple, obtaining $\langle ?, p, o \rangle$; (ii) all entities are tested as the triple subject and ranked according to the resulting score: the original subject s should thus rank as low as possible. An analogous pipeline is used for predicting object o.

[1] https://code.google.com/archive/p/wiki-links/



(a) Distribution of entity degrees in FB15K and WN18.



(b) Distribution of relationship mentions in FB15K. WN18 is omitted as it only features 18 relationships.

Figure 1: Skew analysis for entity degrees distribution and relationship mentions on training, validation and test facts for FB15K and WN18.

These resulting rankings enable the following global metrics:

- *Mean Rank* (MR), i.e. the average rank of the correct subject (object) over all predictions.
- *Mean Reciprocal Rank* (MRR), i.e. the average of the inverse ranks of the correct subject (object) over all predictions.
- *Hits@K*, i.e. the fraction of correct subject (object) predictions with rank equal or lesser than K. The most common choices for K are 10 and 1.

The above described metrics can be computed either in two different settings, dubbed *raw* scenario and *filtered* scenario. As a matter of fact, an incomplete triple $\langle ?, p, o \rangle$ might accept multiple entities as correct answers; an answer is correct if the resulting fact is already contained in the training, validation or test set. In *raw* scenario these entities are still considered "mistakes", and therefore, if they outscore the expected answer, they affect the prediction rank. On the contrary, in *filtered* scenario they are considered acceptable, so if they outscore the expected entity they are just ignored.

3 THE CASE FOR BENCHMARKING

In this section we define key questions that a good LP benchmark should answer in the affirmative, and investigate whether the current benchmarks satisfy them or not, providing experimental

evidence to all our claims. We finally provide an overall discussion for the potential ethical implications of the defined questions and their extent in this research field.

In our experiments, we take into account two representative models, namely `TransE` [4] and `DistMult` [23].

`TransE` is one of the first KG embedding systems, and has inspired dozens of successors. It represents facts as translations in the latent space: its scoring function uses the relationship embedding as a translation vector to move from the embedding of the subject to the one of the object.

`DistMult` is very popular due to its simplicity: its scoring function is a bilinear product among the embedding of the subject, a diagonal matrix based on the embedding of the relationship, and the embedding of the object. If properly fine-tuned, it has been recently shown to surpass most models in the state of the art [7].

We stress that our goal is not to determine which one, among `TransE` and `DistMult`, is better: our purpose is rather to investigate the effectiveness of current benchmarks of highlighting their differences.

	FB15K			
	MR	MRR	H@1	H@10
TransE	70.6	0.497	33.50%	75.68%
DistMult	156.3	0.469	18.04%	74.03%

	FB15K-237			
	MR	MRR	H@1	H@10
TransE	353.5	0.272	18.51%	44.29%
DistMult	741.9	0.139	7.81%	26.08%

	WN18			
	MR	MRR	H@1	H@10
TransE	494.5	0.445	16.00%	81.37%
DistMult	928.9	0.811	70.00%	93.66%

	WN18-RR			
	MR	MRR	H@1	H@10
TransE	5304.3	0.180	2.39%	40.67%
DistMult	9953.5	0.373	36.05%	39.81%

Table 2: Global performances of models in our experiments (filtered scenario).

Table 2 reports the global values of Hits@1, Hits@10, Mean Rank and Mean Reciprocal Rank for both models on all datasets. For our results, we focus on the *filtered* scenario; we have observed analogous findings in *raw* scenario as well. We show that analyzing the behaviour of `TransE` and `DistMult` on the current benchmarks can lead to surprising (and even contradictory) conclusions.

3.1 Experimental Setup

Our experiments have been performed on a server environment with a CPU Intel Core(TM) i7-3820 at 3.60GH, 16GB RAM and a GPU NVIDIA Quadro M5000. We have employed the Tensorflow implementation of `TransE` and `DistMult` provided by the OpenKE toolkit [6]; our Tensorflow version is 1.9. Since comparing `TransE` and `DistMult` is out of the scope of this paper, we have not performed a full-fledged hyperparameter tuning, keeping our setting as similar as possible to the default OpenKE configuration even across different datasets. For the sake of verifiability and reproducibility we report the resulting combination in Table 3.

	Epochs	Batches per epoch	Embedding dimension	Learning Rate	Optimizer
TransE	1000	100	100	0.001	SGD
DistMult	1000	100	100	0.0005	Adam

Table 3: Hyperparameter configurations.

3.2 Questioning current benchmarks

Our questions on the current LP benchmarks refer to their relevance, fairness and capability to highlight overfitting. For each question we provide a formulation, an analysis and an overall answer.

Q1: Does the benchmark measure the ability of the system to learn relations?

Analysis. This question is related to the *relevance* of the benchmark. An LP benchmark is relevant if it actually measures how good the system is at learning relations.

A limitation of current benchmarks lies in their use of global metrics (Hits@K, Mean Rank, Mean Reciprocal Rank) that relate to the *overall number* of accurate predictions rather than their quality. This practice does not take into account that some facts may be inherently different than others. In other words, global metrics do not let the specific strengths and weaknesses of different models surface: this does not allow to investigate in which aspects a model performs better or worse than the others, and why. Ultimately, we believe that this hinders our understanding of what our systems are actually learning.

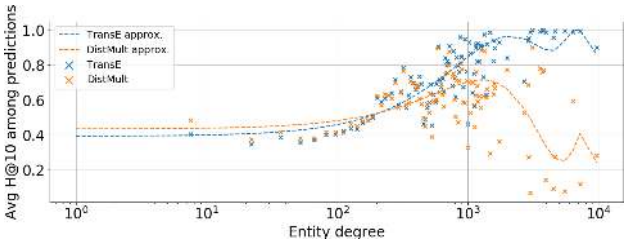
Furthermore, as pointed out by [22], current evaluation metrics are based on positive test facts only, and do not check if false or even nonsensical facts receive low scores in turn.

Finally, it has been recently observed [7] that the extensive use of the Hits@10 metric might be misleading when comparing different models: many systems achieve similar, very good Hits@10 values, but they show marked differences with more selective versions of the same metric, such as Hits@1.

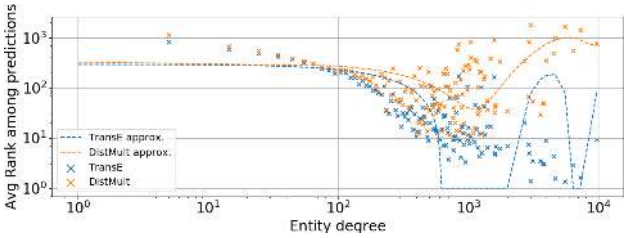
To prove our claim we observe that most datasets display very skewed degree distributions. Our experiments show that the degree of an entity in training set largely affects the LP prediction accuracy in testing; nonetheless, this strong correlation is completely overlooked by the commonly employed global metrics. We plot in Figure 2 the correlation between the entity degree and the prediction performances for the entities with that degree. We measure performances with Hits@10 and MR metrics. Our results provide strong evidence that a higher degree yields better predictions; this pattern holds for the vast majority of entities, up to 1K mentions. We note that despite reaching comparable Hits@10 overall, `DistMult` can significantly outperform `TransE` on low degree entities, while `TransE` is better on the few high degree entities.

We believe that insights like these are vital to understand what our models are actually learning, and to choose the most suitable model for a specific setting; nonetheless, they are completely unobtainable by just relying on current benchmarking metrics.

In order to provide an explanation for the correlation between degree and LP performances, we have analyzed how the degree in training facts correlates to the average distance between the

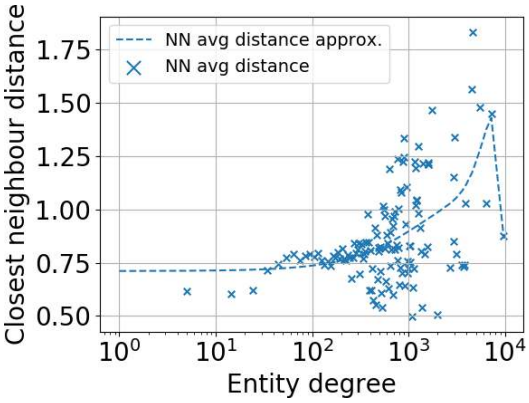


(a) Entity degree in training set and average Hits@10.

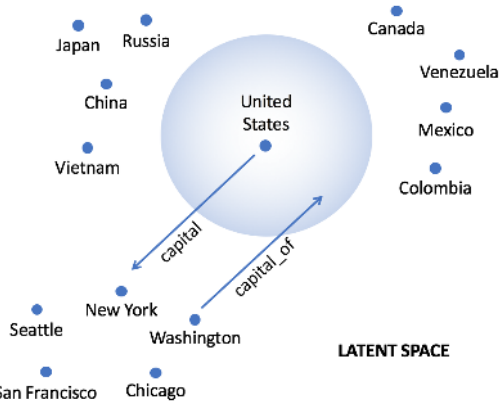


(b) Entity degree in training set and average Rank.

Figure 2: Training entity degree vs average performances when predicting an entity with that degree (FB15K). Dashed lines have been obtained by fitting a polynomial function of degree 4 with the least Squares technique.



(a) Closest neighbours distance by degree (TransE, FB15K).



(b) Stylized example of the high entity degree effect.

Figure 3: Degree in training set vs top 3 closest neighbours distance, and intuitive interpretation of its effects on predictions. The dashed line in (a) has been obtained fitting a polynomial function of degree 4 with least Squares technique.

entities with that degree and their closest neighbor. We report our findings in Figure 3a; in this chart, in order to yield more robust results, for each entity we actually consider an average of the distances from the top three closest neighbors. Interestingly, higher degrees typically correspond to more "isolated" embeddings in the latent space, with greater distances from their closest neighbors.

We interpret this as illustrated in Figure 3b: a "rich" entity such as *United States* has a very isolated embedding, while *Washington* and *New York* lie in a dense area. On the one hand, due to the lack of alternatives in the close neighborhood, it is reasonably easy to operate transformations in the latent space and answer correctly *United States* to the question $\langle Washington, capital\_of, ?\rangle$. On the other hand, the inverse question $\langle UnitedStates, capital, ?\rangle$ is much more difficult, because it requires to learn a very precise transformation, in order to disambiguate between *Washington* and *New York*.

Answer. Entities with high degree, like *United States*, can boost the ability of a model to predict relations mentioning them (e.g., *capital_of*). Therefore, it is hard to understand whether a model has learned a given relation precisely or only its top mentioned entities, by looking only at global metrics like Hits@10.

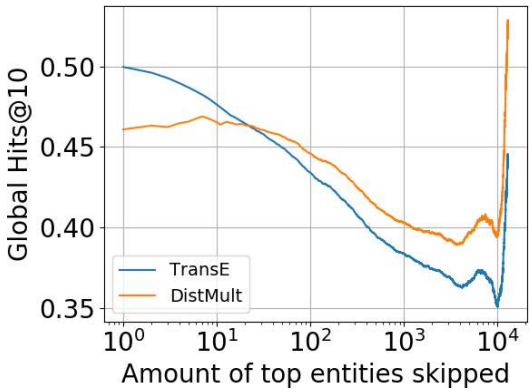
Q2: Does the benchmark measure the performances of models in a fair way?

Analysis. The *fairness* of a benchmark is the absence of unwanted biases in any operation of its workload. Fairness depends both on the metrics (i.e. *what* is measured) and on the composition of the test set (i.e., *how* the measure is computed). In the context of LP, fairness is compromised by the the same correlation observed in the previous section between entity degrees and prediction accuracy. Since both the training set and the test set are obtained from the same uniform sample of the KG, an entity with high degree in the training set will be mentioned more than the others in the test set too. The over-representation of some entities in the test set, combined to the fact that the same entities also enable better predictions, leads to an overall unfairness of the benchmark, favouring "easy" entities with high degrees over harder ones with medium and low degrees.

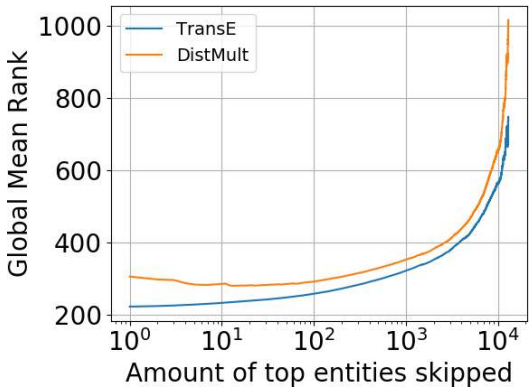
We demonstrate this by studying how progressively skipping test predictions for the top-degree entities affects global performances. We show our results in Figure 4: in both Mean Rank and Hits@10 curves, the more high-degree entities are ignored, the worse the performances become. At this regard, the Hits@10 graph also confirms the slightly different behaviours of `TransE` and `DistMult`, with the former seemingly more depending on the degree than the latter.

We have computed the number of entities that contribute the most to the global Hits@10 metrics. The results are impressive: in FB15K 80% of the global Hits@10 come from 24.1% entities in `TransE`, from 28.5% entities in `DistMult`. An even more extreme situation is witnessed in WN18, where 80% of the global Hits@10 come from 9.87% entities in `TransE` and 11.6% entities in `DistMult`.

Answer. High degree entities, in addition to be more easily inferred, are also over-represented in the *test set* and a model may obtain significantly good evaluations by just focusing on a small number of high-degree entities.



(a) Global Hits@10 when progressively skipping tests on top degree entities (FB15K).



(b) Global Mean Rank when progressively skipping tests on top degree entities (FB15K).

Figure 4: Effects on global metrics when progressively ignoring test predictions on up to 95% of the the entities with highest degree (FB15K).

Q3: Does the benchmark discourage (or at least highlight) overfitting?

Analysis. Overfitting takes place when a model matches its dataset too closely, conforming to noise and irrelevant correlations in its samples: an overfitted model fails to generalize, and will not behave correctly when dealing with unseen data. A good benchmark can highlight the emerging of overfitting.

In the LP scenario, as already pointed out by [17] and [5], FB15K and WN18 significantly suffer from test leakage, with inverse triples from the training occurring in the test set. As a consequence, even extremely simple systems can reach state-of-the-art performances on those datasets [7], thus casting doubt on the generality of reported results.

We have also observed that, when removing from both training and test sets the top-degree entities and retraining the model, performances improve instead of worsening as one would expect from the previous findings (note that in this experiment we are also retraining, differently from the experiment reported in Figure 4). This counter-intuitive pattern is steadily visible in Table 4 when removing the top 10, 25 and 100 entities. This phenomenon may be partly caused by the fact that, when removing a high-degree entity, along with all of its facts, a large number of test "questions" about its low-degree neighbors (on which the model would perform badly) are removed as well. Nonetheless, obtaining paradoxically better results when removing training samples that the system would apparently learn well is a typical sign of overfitting. In our case, removing from the training set entities whose embeddings would take very large portions of the embedding space may allow the other entities to be placed in better positions; this can be seen as a form of regularization.

	MR	MRR	H@10
Complete	70	0.49	75.67%
Top 10 entities removed	66	0.49	76.55%
Top 25 entities removed	64	0.50	78.22%
Top 100 entities removed	68	0.53	80.64%

Table 4: TransE performances in filtered scenario when removing the top degree entities from FB15K.

Finally, all the currently employed benchmarks display a static separation between training set, validation set and test set. This is known to be a bad practice, because in time it may favour models overfitted on this configuration. Running K-folding cross-validation for the two models on both FB15K and WN18 we did not observe significant signs of this form of overfitting yet. Nonetheless, we advice to employ K-folding whenever possible as a way to prevent it in the future.

Answer. The counter-intuitive boost of performances when removing high-degree entities from both the training and test set suggests an undetected form of overfitting towards these entities.

The above mentioned observations on the relevance, fairness and capability to discourage overfitting of the current benchmarking practices can have interesting implications within the ethics of information processing.

Relying on ineffective benchmarks undermines the capability to assess the quality of the software and of the data it should manage. We have highlighted that the current evaluation practices may not detect, or may even penalize desirable properties in LP models. For instance, the results currently yielded in evaluation can not tell whether a model is learning a large set of relations or, rather, a narrow set of entities. This can lead to systematic overlooking of underrepresented entities, because systems that actually reason on the the entire set of entities can be outranked by models that overfit on few over-represented ones.

We also point out that the opacity of current results, computed with global metrics over large batches of test facts, makes it almost impossible to interpret the behaviours of models. This, of course, has negative effects on their explainability and, ultimately, on their trustworthiness.

4 RELATED WORKS

To the best of our knowledge, there are just a few papers investigating the validity of current LP benchmarks, and providing interpretation for the performances of relation-learning models. Works related to ours can be roughly divided into two main categories, depending on whether they address limitations of the standard metrics or of the datasets used in this research field. For instance, the already mentioned work by [7] demonstrates that a carefully tuned implementation of `DistMult` can achieve state of the art performances, surpassing most of its own successors, raising questions on whether we are developing better LP models or we are just tuning better hyperparameters.

Limitations of standard metrics. The most similar work to ours is [22]: they observe that the currently employed metrics tend to be biased as they are computed using only "positive" test facts, originally belonging to the KG. For instance, if fact ⟨ *Barack Obama*, *place of birth*, *Honolulu* ⟩ is seen in the test set, our test questions will be ⟨ *?*, *place of birth*, *Honolulu* ⟩ and ⟨ *Barack Obama*, *place of birth*, *?* ⟩. This approach is highly biased as it just scores triples for which an answer is already known to exist. It is more akin to Question Answering than to Knowledge Graph Completion, because it never tests the plausibility of nonsensical facts, such as ⟨ *Honolulu*, *place of birth*, ? ⟩, or facts that have no answer, such as ⟨ *Barack Obama*, *place of death*, *?* ⟩ They then propose a new testing workload in which all possible couples of entities are tested for all relationships, in order to check whether any false or nonsensical triples manage to obtain high plausibility scores.

Limitations of standard datasets. Some of the current LP benchmarking workloads have been already put into discussion by a few previous works, to which we refer in our analysis. In general, these works do not aim at performing a systematic investigation of the benchmark properties; on the contrary, they just highlight a specific issue, often in the context of presenting a new model or implementation. To the best of our knowledge, [17] has been the first study to openly discuss the limitations of FB15K, demonstrating that it heavily suffers from test leakage: many relationships in this datasets are semantically identical or inverse to others, allowing even a very simple model based on observed features to outperform most embedding-based state of the art ones. The authors have then proceeded to extract a more challenging subset from FB15K, called FB15K-237, containing non-trivial facts only. Unfortunately, FB15K-237 has been only partially used by the research community, with prominent models such as HolE ([10]), ComplEx ([18]) and ANALOGY ([9]) ignoring it.

Starting from their analysis, [5] have further investigated test leakage in both FB15K and WN18. They have demonstrated that a simple rule-based system based on inverse relationships can reach state of the art performances in WN18; they have then applied a similar procedure as [17] on WN18 to generate its challenging subset WN18-RR.

Other tasks. For the sake of completeness we also observe that, when proposing new models for LP, many papers analyze their applicability and performances on related tasks too. For instance, the authors of [8] show the performances of their model in relational extraction from text using the NYT-FB dataset [14], where sentences from the New York Times Corpus are annotated with Stanford NER and linked to Freebase elements. Analyzing the properties of benchmarks for relation extraction tasks is out of the scope of our work.

5 CONCLUSIONS

We have analyzed the current LP benchmarks, observing that the training sets of their datasets display severely skewed distributions in both the degrees of entities and the mentions of relationships.

We have experimentally demonstrated that LP models are deeply affected by these unbalanced conditions; nonetheless, these effects go completely unnoticed by the current evaluation workloads, thus casting doubts on their relevance. We have also displayed that entities and relationships that are highly mentioned in *training* sets tend to be over-represented in *test* sets too, affecting the fairness of the evaluation workload. We have finally reported that, ignoring entities with high degree (and thus high performances), LP models show a counter-intuitive improvement in performances, potentially attributable to overfitting.

Overall, our results raise concerns on the effectiveness of these benchmarks. We demonstrate that relying on global metrics over heavily skewed distributions hinders our understanding of LP models; all in all, our results imply that at their current state these benchmarking practices may not be able to capture and fairly measure the capability of relation-learning models to effectively learn relations.

ACKNOWLEDGEMENTS

Work funded in part by Regione Lazio LR 13/08 Project “In Codice Ratio” (14832).

REFERENCES

[1] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives. Dbpedia: A nucleus for a web of open data. In *The semantic web*, pages 722–735. Springer, 2007.
[2] I. Balažević, C. Allen, and T. M. Hospedales. Tucker: Tensor factorization for knowledge graph completion. *arXiv preprint arXiv:1901.09590*, 2019.
[3] K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250. ACM, 2008.
[4] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, and O. Yakhnenko. Translating embeddings for modeling multi-relational data. In *Advances in neural information processing systems*, pages 2787–2795. NIPS, 2013.
[5] T. Dettmers, P. Minervini, P. Stenetorp, and S. Riedel. Convolutional 2d knowledge graph embeddings. In *AAAI Conference on Artificial Intelligence*, 2018.
[6] X. Han, S. Cao, X. Lv, Y. Lin, Z. Liu, M. Sun, and J. Li. Openke: An open toolkit for knowledge embedding. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 139–144, 2018.
[7] R. Kadlec, O. Bajgar, and J. Kleindienst. Knowledge base completion: Baselines strike back. *arXiv preprint arXiv:1705.10744*, 2017.
[8] Y. Lin, Z. Liu, M. Sun, Y. Liu, and X. Zhu. Learning entity and relation embeddings for knowledge graph completion. In *Proceedings of the 29th AAAI Conference on Artificial Intelligence*, volume 15, pages 2181–2187. AAAI, 2015.
[9] H. Liu, Y. Wu, and Y. Yang. Analogical inference for multi-relational embeddings. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 2168–2178. JMLR. org, 2017.
[10] M. Nickel, L. Rosasco, and T. Poggio. Holographic embeddings of knowledge graphs. In *Thirtieth Aaai conference on artificial intelligence*, 2016.
[11] H. Paulheim. Knowledge graph refinement: A survey of approaches and evaluation methods. *Semantic web*, 8(3):489–508, 2017.
[12] R. Qian. Understand your world with bing, 2013. Blogpost in Bing Blogs.
[13] R. Richardson, A. F. Smeaton, and J. N. Murphy. Using wordnet as a knowledge base for measuring semantic similarity between words. Technical report, In Proceedings of AICS Conference, 1994.
[14] S. Riedel, L. Yao, and A. McCallum. Modeling relations and their mentions without labeled text. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 148–163. Springer, 2010.
[15] A. Singhal. Introducing the knowledge graph: things, not strings, 2012. Blogpost in the Official Google Blog.
[16] F. M. Suchanek, G. Kasneci, and G. Weikum. Yago: a core of semantic knowledge. In *Proceedings of the 16th international conference on World Wide Web*, pages 697–706. ACM, 2007.
[17] K. Toutanova and D. Chen. Observed versus latent features for knowledge base and text inference. In *Proceedings of the 3rd Workshop on Continuous Vector Space Models and their Compositionality*, pages 57–66, 2015.
[18] T. Trouillon, J. Welbl, S. Riedel, É. Gaussier, and G. Bouchard. Complex embeddings for simple link prediction. In *International Conference on Machine Learning*, pages 2071–2080, 2016.
[19] J. von Kistowski, J. A. Arnold, K. Huppler, K.-D. Lange, J. L. Henning, and P. Cao. How to build a benchmark. 02 2015.
[20] D. Vrandečić and M. Krötzsch. Wikidata: a free collaborative knowledge base. 2014.
[21] Q. Wang, Z. Mao, B. Wang, and L. Guo. Knowledge graph embedding: A survey of approaches and applications. *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, 29(12):2724–2743, 2017.
[22] Y. Wang, D. Ruffinelli, S. Broscheit, and R. Gemulla. On evaluating embedding models for knowledge base completion. *arXiv preprint arXiv:1810.07180*, 2018.
[23] B. Yang, W.-t. Yih, X. He, J. Gao, and L. Deng. Embedding entities and relations for learning and inference in knowledge bases. *arXiv preprint arXiv:1412.6575*, 2014.