Evaluating the Calibration of Knowledge Graph Embeddings for Trustworthy Link Prediction

Tara Safavi[*]
University of Michigan
tsafavi@umich.edu

Danai Koutra
University of Michigan
dkoutra@umich.edu

Edgar Meij
Bloomberg
emeij@bloomberg.net

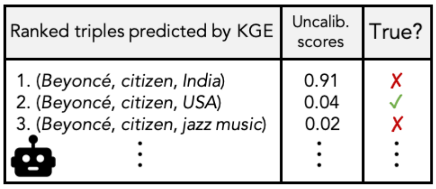
Abstract

Little is known about the trustworthiness of predictions made by knowledge graph embedding (KGE) models. In this paper we take initial steps toward this direction by investigating the *calibration* of KGE models, or the extent to which they output confidence scores that reflect the expected correctness of predicted knowledge graph triples. We first conduct an evaluation under the standard closed-world assumption (CWA), in which predicted triples not already in the knowledge graph are considered false, and show that existing calibration techniques are effective for KGE under this common but narrow assumption. Next, we introduce the more realistic but challenging open-world assumption (OWA), in which unobserved predictions are not considered true or false until ground-truth labels are obtained. Here, we show that existing calibration techniques are much less effective under the OWA than the CWA, and provide explanations for this discrepancy. Finally, to motivate the utility of calibration for KGE from a practitioner's perspective, we conduct a unique case study of human-AI collaboration, showing that calibrated predictions can improve human performance in a knowledge graph completion task.

1 Introduction

Knowledge graphs are essential resources in natural language processing tasks such as question answering and reading comprehension (Shen et al., 2019; Yang et al., 2019). Because they are by nature incomplete, extensive research efforts have been invested into completing them via different techniques (Ji et al., 2020; Belth et al., 2020).

One such technique is knowledge graph embedding (**KGE**), which involves learning latent representations of entities and relations to be used toward predicting new facts. KGE models are most commonly optimized for the **link prediction** task, which tests their ability to "learn to rank" plausible knowledge graph triples higher than implausible ones. While KGE accuracy as measured by ranking-based link prediction metrics has been steadily improving on benchmark datasets over the past decade (Ruffinelli et al., 2020), such evaluation setups can be misleading. As shown in Figure 1, ranking only considers the *ordering* of prediction scores, so models can perform well according to ranking metrics even if they assign high scores to incorrect or nonsensical triples (Wang et al., 2019).



Ranked triples predicted by KGE	Uncalib. scores	True?
1. (*Beyoncé, citizen, India*)	0.91	✗
2. (*Beyoncé, citizen, USA*)	0.04	✓
3. (*Beyoncé, citizen, jazz music*)	0.02	✗
⋮	⋮	⋮

Figure 1: An example of how optimizing for ranking does not necessarily lead to trustworthy prediction scores. Here, an uncalibrated KGE model would perform well according to ranking metrics because the true triple is ranked highly, even though it receives a much lower score than the incorrect top-ranked triple and a similar score to the nonsensical triple below it.

As such, the practical utility of KGE for real-world knowledge graph completion remains limited, especially given that other completion techniques such as relation extraction and manual curation have already been reliably deployed in commercial and scientific settings (Suchanek et al., 2007; Dong et al., 2014; Ammar et al., 2018). The outstanding issue that we address relates to *trustworthiness*: That is, to what degree can one trust the predictions made by KGE? We believe that trustworthiness is an important part of making KGE practical for knowledge graph completion.

In this paper we propose to investigate **confidence calibration** as a technique toward making KGE more trustworthy. Intuitively, calibration is a

[*] This work was done during an internship at Bloomberg.

Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, pages 8308–8321,
November 16–20, 2020. ©2020 Association for Computational Linguistics

post-processing step that adjusts KGE link prediction scores to be representative of actual correctness probabilities (Guo et al., 2017). Calibration has several benefits. From the *systems* perspective, natural language processing pipelines that include knowledge graphs can rely on calibrated confidence scores to determine which KGE predictions to trust. From a *practitioner*'s perspective, calibrated confidence scores act as decision support for accepting or verifying KGE predictions. Toward this direction we contribute the following:

Task We evaluate KGE calibration for link prediction, which is important for making KGE viable for deployment. While many knowledge graph embedding models exist, their calibration and general trustworthiness are under-explored (§ 2).

Complementary evaluations We first evaluate the calibration of established KGE models under the commonly-used closed-world assumption (**CWA**), in which triples not present in the knowledge graph are considered false (§ 4). We show that existing calibration techniques are highly effective for KGE under this assumption. Next, we introduce the more challenging open-world assumption (**OWA**), which reflects how *practitioners* would use KGE: Triples not present in the knowledge graph are assumed to be *unknown*, rather than false, until ground-truth labels are obtained (§ 5). We show that existing calibration techniques are less effective under the OWA than the CWA, and provide explanations for this discrepancy.

Case study Finally, as a proof of concept on the benefits of KGE calibration, we conduct a case study in which data annotators complete knowledge graph triples with the help of KGE predictions. We show that presenting calibrated confidence scores alongside predictions significantly improves human accuracy and efficiency in the task, motivating the utility of calibration for human-AI tasks.

2 Related work

While knowledge graph embeddings and calibration have both been extensively studied in separate communities—see (Ji et al., 2020; Ruffinelli et al., 2020) for reviews of KGE and (Guo et al., 2017) for an overview of calibration for machine learning—relatively little work on calibration *for* knowledge graph embeddings exists.

In the domain of relation extraction, a few works calibrate predicted knowledge graph triples as components of large-scale relation extraction systems. Dong et al. (2014) used Platt scaling (Platt et al., 1999) to calibrate the probabilities of factual triples in the proprietary Knowledge Vault dataset, and West et al. (2014) used Platt scaling in a search-based fact extraction system. However, we focus on link prediction with KGE models that learn only from the knowledge graph itself (§ 3.1).

We are aware of only two recent works that investigate calibration for KGE, both of which address the task of triple classification (Tabacof and Costabello, 2020; Pezeshkpour et al., 2020). By contrast, we focus on link prediction, which is a different—and much more common (Safavi and Koutra, 2020)—KGE evaluation task. We also contribute an evaluation under the open-world assumption, whereas Tabacof and Costabello (2020) evaluate under the closed-world assumption only. Finally, unique to our work, we conduct a human-AI case study to demonstrate the benefits of calibration from a practitioner's perspective.

3 Preliminaries

3.1 Knowledge graph embeddings

A knowledge graph G comprises a set of entities E, relations R, and (*head*, *relation*, *tail*) triples $(h, r, t) \in E \times R \times E$. A knowledge graph embedding (**KGE**) takes triples (h, r, t) as input and learns corresponding embeddings $(\mathbf{h}, \mathbf{r}, \mathbf{t})$ to maximize a scoring function $f : E \times R \times E \rightarrow \mathbb{R}$, such that more plausible triples receive higher scores.

Models In this paper we consider four KGE models: **TransE** (Bordes et al., 2013), **TransH** (Wang et al., 2014), **DistMult** (Yang et al., 2015), and **ComplEx** (Trouillon et al., 2016). Table 1 gives the scoring function of each model.

We choose these models because they are efficient and representative of two main classes of KGE architecture—translational (TransE, TransH) and bilinear (DistMult, ComplEx)—which allows us to interpret how different types of scoring functions affect calibration. Moreover, these earlier models tend to be used by NLP practitioners. For example, the language model in (Logan et al., 2019) uses TransE embeddings, and the machine reading system in (Yang et al., 2019) uses DistMult embeddings. From our knowledge of the literature, practitioners using KGE are more likely to use earlier, established models. Since our work targets real-world applications, we prioritize such models.

Table 1: Scoring functions of models used in our evaluation. **Bold letters** indicate vector embeddings. + indicates that the scoring function is translational, and × indicates that the scoring function is bilinear.

	Type	Scoring function f	Scoring function notes
TransE (Bordes et al., 2013)	$+$	$-\|\mathbf{h}+\mathbf{r}-\mathbf{t}\|$	We use the L_2 norm
TransH (Wang et al., 2014)	$+$	$-\|\mathbf{h}_\perp+\mathbf{r}-\mathbf{t}_\perp\|$	Projects $\mathbf{h}$, $\mathbf{t}$ onto relation-specific hyperplanes to get $\mathbf{h}_\perp$, $\mathbf{t}_\perp$
DistMult (Yang et al., 2015)	$\times$	$\mathbf{h}^\top \mathrm{diag}(\mathbf{r})\mathbf{t}$	$\mathrm{diag}(\cdot)$ turns a vector into a diagonal matrix
ComplEx (Trouillon et al., 2016)	$\times$	$\mathrm{Re}\left(\mathbf{h}^\top \mathrm{diag}(\mathbf{r})\bar{\mathbf{t}}\right)$	$\bar{\mathbf{t}}$: Complex conjugate of $\mathbf{t}$; Re: Real part of a complex number

3.2 Link prediction

The link prediction task, which is most commonly used to evaluate KGE (Safavi and Koutra, 2020), is conducted as follows. Given a test triple (h, r, t), we hold out one of its entities or its relation to form a query $(h, r, ?)$, $(?, r, t)$, or $(h, ?, t)$. The model then scores all tail entities $t_i \in E$, head entities $h_i \in E$, or relations $r_i \in R$ as answers to the respective query such that higher-ranked completions (h, r, t_i), (h_i, r, t), or (h, r_i, t) are more plausible. Prior to computing rankings, all true triples across train, validation, and test beyond the given test triple are filtered out (Bordes et al., 2013).

Under the closed-world assumption (**CWA**, § 4), models are evaluated by their ability to score the true test triples (h, r, t) as high as possible, because it is assumed that all triples not seen in the knowledge graph are incorrect. Under the open-world assumption (**OWA**, § 5), models simply score all predicted completions, and the predictions not seen in the knowledge graph are not considered true or false until ground-truth labels are obtained.

3.3 Confidence calibration

In the context of link prediction, calibration is the extent to which a KGE model outputs probabilistic confidence scores that reflect its expected accuracy in answering queries. For example, for 100 predicted triple completions scored at a confidence level of 0.99 by a perfectly calibrated model, we expect 99 of these predictions to be correct.

Calibration is a *post-processing* step. To calibrate a KGE model, separate calibration parameters are learned on a held-out validation set using the prediction scores of the uncalibrated model. These parameters do not affect the trained, fixed embeddings, but rather transform the model's scores.

Negative samples All calibration methods require negatives to appropriately adjust prediction scores for plausible and implausible triples. However, link prediction benchmarks (§ 4.2) do not contain negatives. Therefore, per positive instance, we assume that only the held-out entity or relation correctly answers the query, and take all other completions as negative samples.

This approach, which has been shown to work well in practice for training KGE models (Ruffinelli et al., 2020), treats link prediction as *multiclass*: The "class" for each query is its true, held-out entity or relation. Since this approach is less faithful to reality for queries that have many entities as correct answers, in this paper we evaluate calibration for the **relation prediction** task—that is, answering $(h, ?, t)$ queries—because there are usually fewer correct answers to relation queries than entity queries.[1] While the methods we describe are general, for brevity we focus on relation prediction in this rest of this section.

3.4 Calibration techniques

Let $(h, ?, t)$ be a relation query, $k = |R|$ be the number of relations in the graph, and

$$\mathbf{z} = [f(h, r_1, t), \ldots, f(h, r_k, t)]^\top \in \mathbb{R}^k \tag{1}$$

be a vector of uncalibrated KGE prediction scores across all relations $r_i \in R$, such that $z_i = f(h, r_i, t)$. Note that for head or tail queries $(?, r, t)$ or $(h, r, ?)$, $\mathbf{z}$ would instead contain prediction scores across all entities in E.

Our goal is to learn a function that transforms the uncalibrated score vector $\mathbf{z}$ into *calibrated probabilities* $\mathbf{z}' \in \mathbb{R}^k$. Post-calibration, the final predicted answer $\hat{r}$ to the query and corresponding confidence score $\hat{p}$ are taken as

$$\hat{r} = \arg\max\,[\mathbf{z}'] \text{ and } \hat{p} = \max\,[\mathbf{z}'], \tag{2}$$

where $\hat{p}$ reflects the expectation that $\hat{r}$ correctly answers the query, i.e., is the "class" of the query.

One-versus-all One approach to multiclass calibration is to set up k one-versus-all *binary* calibration problems, and combine the calibrated probabilities for each class afterward. The classic **Platt**

[1] For example, in our `FB15K-Wiki` dataset (§ 4.2), the mean number of relations between each unique pair of entities is 1.12, and the median is 1.

scaling technique (Platt et al., 1999), which was originally designed for binary classification, can be extended to the multiclass setting in this manner. For each class, scalar parameters a and b are learned such that the calibrated probability of the query belonging to the i-th class is given by $\hat{p}_i = \sigma_{\text{sig}}(az_i + b)$, where σ_{sig} denotes the logistic sigmoid. The parameters are optimized with negative log-likelihood (i.e., binary cross-entropy) loss, which is standard for obtaining probabilistic predictions (Niculescu-Mizil and Caruana, 2005). Afterward, all $\hat{p}_i$ are gathered into $\mathbf{z}' = [\hat{p}_1, \ldots, \hat{p}_k]$ and normalized to sum to 1.

Another well-known calibration technique that fits in the one-versus-all framework is **isotonic regression** (Zadrozny and Elkan, 2002). For each class, a nondecreasing, piecewise constant function g is learned to minimize the sum of squares $[\mathbf{1}(r_i) - g(\sigma_{\text{sig}}(z_i))]^2$ across all queries, where $\mathbf{1}(r_i)$ is 1 if the class of the given query is r_i and 0 otherwise. The calibrated probability of the query belonging to the i-th class is taken as $\hat{p}_i = g(\sigma_{\text{sig}}(z_i))$. Again, these scores are gathered into $\mathbf{z}' = [\hat{p}_1, \ldots, \hat{p}_k]$ and normalized to sum to 1.

Multiclass An alternative approach is to use the softmax σ_{sm} to directly obtain probabilities over k classes, rather than normalizing independent logistic sigmoids. To this end, Guo et al. (2017) propose a variant of Platt scaling that learns weights $\mathbf{A} \in \mathbb{R}^{k \times k}$ and biases $\mathbf{b} \in \mathbb{R}^k$ to obtain calibrated confidences $\mathbf{z}' = \sigma_{\text{sm}}(\mathbf{Az} + \mathbf{b})$. $\mathbf{A}$ and $\mathbf{b}$ are optimized with cross-entropy loss.

The weight matrix $\mathbf{A}$ can either be learned with the full k^2 parameters (**matrix scaling**), or can be restricted to be diagonal (**vector scaling**). We compare both approaches in § 4.

4 Closed-world evaluation

We first evaluate KGE calibration under the **closed-world assumption** (CWA), in which we assume triples not observed in a given knowledge graph are false. This assumption, which is standard in KGE evaluation (Ruffinelli et al., 2020), helps narrow evaluation down to a well-defined task in which models are judged solely by their ability to fit known data. It is therefore important to first explore this (restrictive) assumption before moving to the more realistic but challenging OWA (§ 5).

Table 2: Datasets used in our closed-world evaluation.

	# entities	# relations	# triples
`WN18RR`	40,493	11	93,003
`FB15K-Wiki`	14,290	773	272,192

4.1 Task and metrics

As described in § 3.2, link prediction under the CWA is conducted by constructing queries from test triples and evaluating models' abilities to score these test triples as high as possible. We measure accuracy by the proportion of top-ranked predicted relations that correctly answer each query.[2]

We quantify a KGE model's level of calibration with expected calibration error (**ECE**) (Guo et al., 2017). ECE measures the degree to which a model's confidence scores match its link prediction accuracy in bins partitioning [0, 1]. Given M such bins of equal size, ECE is defined as $\sum_{m=1}^{M} \frac{|B_m|}{n} |\text{acc}(B_m) - \text{conf}(B_m)|$, where n is the number of test triples, B_m is the bin containing all predictions with confidence score in a given region of $[0, 1]$, acc(B_m) measures the average link prediction accuracy in bin B_m, and conf(B_m) measures the average confidence score in bin B_m. ECE is in [0, 1], and lower is better. For all reported ECE values, we use 10 bins.

4.2 Data

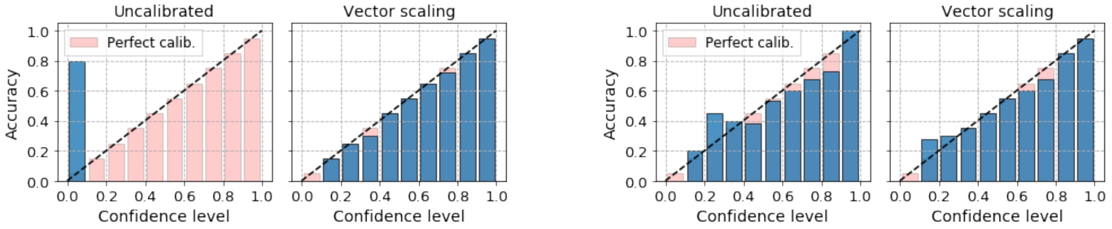
We use two link prediction benchmarks (Table 2): The `WN18RR` semantic relation network (Dettmers et al., 2018) and a version of the FB15K encyclopedic knowledge graph (Bordes et al., 2013). We refer to this dataset as `FB15K-Wiki` because we link it to Wikidata (Vrandečić and Krötzsch, 2014) to use as an external reference in § 5 for data annotation, discarding entities without entries in Wikidata. Following standard practice, we remove inverse relations from `FB15K-Wiki`, which artificially inflate link prediction accuracy (Dettmers et al., 2018). We randomly split both datasets into 80/10/10 train/validation/test triples to ensure a sufficient number of validation triples for calibration.

Note that there have been recent (concurrent) efforts to construct appropriate datasets for evaluating KGE calibration (Pezeshkpour et al., 2020; Safavi and Koutra, 2020). Analysis on these new datasets is an important direction for future work.

[2] Here we use top-1 accuracy because there are relatively few relations in knowledge graphs. However, any binary link prediction metric (i.e., hits@k) may be used.

Table 3: ECE (10 bins) and accuracy on `WN18RR` and `FB15K-Wiki`. ↑: Higher is better. ↓: Lower is better.

		WN18RR					FB15K-Wiki				
		Uncalib.	One-vs-all		Multiclass		Uncalib.	One-vs-all		Multiclass	
			Platt	Iso.	Vector	Matrix		Platt	Iso.	Vector	Matrix
ECE (↓)	TransE	0.624	0.054	0.040	**0.014**	0.022	0.795	0.071	**0.016**	0.026	0.084
	TransH	0.054	0.057	0.044	**0.018**	0.027	0.177	0.081	**0.024**	0.031	0.089
	DistMult	0.046	0.040	0.029	0.044	**0.014**	0.104	0.095	0.031	**0.018**	0.054
	ComplEx	0.028	0.041	0.034	0.035	**0.020**	0.055	0.102	0.037	**0.024**	0.112
Acc. (↑)	TransE	0.609	0.609	0.609	0.724	**0.739**	0.849	0.849	0.849	**0.857**	0.842
	TransH	0.625	0.625	0.625	0.735	**0.740**	0.850	0.850	0.850	**0.858**	0.839
	DistMult	0.570	0.570	0.570	0.723	**0.761**	0.819	0.819	0.819	0.862	**0.871**
	ComplEx	0.571	0.571	0.571	0.750	**0.781**	0.884	0.884	0.884	**0.908**	0.892



(a) TransE

(b) DistMult

Figure 2: Reliability diagrams on `FB15K-Wiki` with predictions grouped into 10 bins.

4.3 Results and discussion

We implement our methods in an extension of the OpenKE library.[3] To understand "off-the-shelf" calibration, we train models with the original loss functions and optimizers in the respective papers. Appendix A provides details on implementation and model selection.

Calibration error Table 3 gives the ECE of all models before and after calibration using each technique in § 3.4. Confidence scores prior to calibration are scaled via the softmax. Across datasets, standard techniques **calibrate models within 1-2 percentage points of error** under the CWA. In most cases, the strongest methods are the multiclass (softmax) approaches. The only exception is matrix scaling on `FB15K-Wiki`, which overfits due to the large number of classes in the dataset (773 in `FB15K-Wiki` versus only 11 in `WN18RR`, Table 2). Evidently, taking the softmax over k classes leads to more discriminative probabilities than setting up k separate one-versus-all calibration problems and performing post-hoc normalization.

We also observe that off-the-shelf calibration error is correlated with model type, as the bilinear models (DistMult, ComplEx) consistently have lower ECE than the translational models (TransE, TransH). To illustrate these differences, Figure 2 gives reliability diagrams for TransE and DistMult before and after calibration. Reliability diagrams (Guo et al., 2017) bin predictions by confidence level into equally-sized regions of [0, 1] and show the relationship between average confidence level and accuracy in each bin, similar to ECE (§ 4.1). Without calibration, TransE is underconfident because it scores all predictions nearly the same, whereas DistMult is better calibrated. We observe a similar pattern for TransH and ComplEx.

One potential explanation for this difference is that multiplicative scoring functions lead to more discriminative scores due to the composition of dot products, which amplify embedding values. In fact, TransE is the only model that does not apply any dot product-based transformation to embeddings, leading to the worst off-the-shelf calibration. Another explanation relates to losses: All methods except ComplEx are trained with margin ranking loss, which optimizes the ordering of predictions rather than the values of prediction scores. By contrast, ComplEx is trained with binary cross-entropy loss, the same loss that we use to calibrate models in the validation stage.

Link prediction accuracy Table 3 also compares link prediction accuracy before and after calibration. In most cases vector and matrix scaling

[3] https://github.com/thunlp/OpenKE/

improve accuracy, which is reminiscent of previous work showing that training KGE with softmax cross-entropy improves link prediction performance (Kadlec et al., 2017; Safavi and Koutra, 2020). We conclude that for relation prediction under the CWA, **vector scaling provides the best trade-off** between calibration, accuracy, and efficiency, as it consistently improves accuracy and calibration with only $O(k)$ extra parameters.

5 Open-world evaluation

We now address the more realistic **open-world assumption** (OWA), in which predictions not present in the knowledge graph are considered *unknown*, rather than false, until ground-truth labels are obtained. While the OWA is beneficial because it helps us assess KGE calibration under more realistic conditions, it is also challenging because it significantly changes the requirements for evaluation. Specifically, now we need a label for every triple considered, whereas with the CWA we only needed labels for a small group of positives.

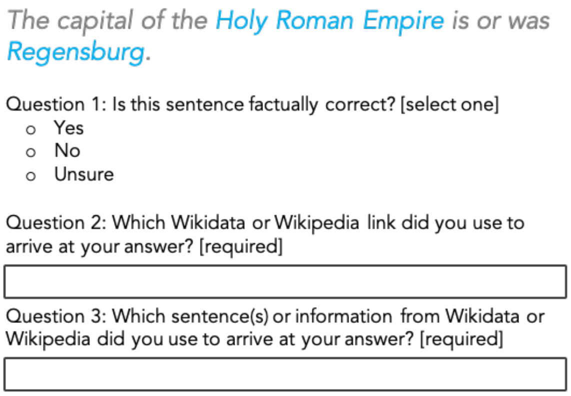
We emphasize that this is the reason the OWA is rarely used to evaluate KGE. Narrowing down the large space of unknowns to a manageable smaller set and labeling these triples can be difficult and costly. We thus contribute first steps toward evaluation strategies under the OWA.

5.1 Task and metrics

Similar to the link prediction task in § 4.1, we construct $(h, ?, t)$ queries from (h, r, t) knowledge graph triples. A KGE model then scores relations $r_i \in R$ to answer these queries. However, here we only consider completions $(h, r_i, t) \notin G$, those for which the *truth values are not known ahead of time*, which reflects how practitioners would use KGE to complete knowledge graphs in deployment settings. We use FB15K-Wiki as our dataset for this task because it is linked to Wikidata; we provide links to entities' Wikidata pages in our crowdsourced label collection process (§ 5.2).

Generating OWA predictions For each $(h, ?, t)$ query, we take the top-ranked $(h, \hat{r}, t)$ prediction made by a KGE model, and filter these predictions to unknowns $(h, \hat{r}, t) \notin G$.

To simulate how a practitioner might narrow down a large set of unknowns to a few promising candidates under resource constraints (i.e., the cost of collecting labels), we take only the predictions made with confidence level ≥ 0.80. In other words, we choose to obtain *many* judgments of a *few* high-confidence predictions rather than *few* judgments of *many* lower-confidence predictions. This helps us robustly compute agreement, maximize the probability of positives, and control quality.



The capital of the Holy Roman Empire is or was Regensburg.

Question 1: Is this sentence factually correct? [select one]
○ Yes
○ No
○ Unsure

Question 2: Which Wikidata or Wikipedia link did you use to arrive at your answer? [required]

Question 3: Which sentence(s) or information from Wikidata or Wikipedia did you use to arrive at your answer? [required]

Figure 3: Open-world annotation interface.

We run this generation process for each KGE model from § 3.1 trained on FB15K-Wiki before and after calibration. We use vector scaling as our calibrator because it yields the best results on FB15K-Wiki under the CWA (§ 4.3).

For evaluation, we use the same accuracy and calibration metrics as in § 4.1. However, since there is no "test set" in the open world, we must obtain ground-truth labels on predictions, discussed next.

5.2 Data annotation

We collect judgments of the unknown $(h, \hat{r}, t) \notin G$ predictions over FB15K-Wiki using the Figure 8 crowdsourcing platform.[4] In the task, crowd workers answer whether each prediction is factually correct (Figure 3). Triples are presented as sentences, converted via pre-defined relation templates, with links to the Wikidata entries of the head and tail entities. Appendix B gives sentence template examples, as well as more details on data preprocessing and the data annotation instructions.

Participants We limit the annotation task to the highest-trusted group of contributors on Figure 8, and require references from Wikidata or Wikipedia for answers. We also pre-label 20% of all triples and require participants to pass a 5-question "quiz" before starting the task and maintain 90% accuracy on the remaining gold questions. We gather judgments for 1,152 triples, and collect five judgments per triple, taking the majority label as ground-truth. The inter-annotator agreement using Fleiss'

[4] https://www.figure-eight.com/

Table 5: Examples of OWA predictions before and after calibration.

	Head h	Predicted relation $\hat{r}$	Tail t	Model	Conf. $\hat{p}$	True?
Uncalib.	Bloomfield Hills	/location/administrative_division/second_level_division_of	United States of America	ComplEx	0.985	✗
	Spanish	/language/human_language/countries_spoken_in	Spain	ComplEx	0.946	✓
	New Hampshire	/location/location/containedby	Hampshire	DistMult	0.860	✗
	Billie Holiday	/music/artist/origin	New York City	ComplEx	0.844	✓
	egg	/food/ingredient/compatible_with_dietary_restrictions	veganism	ComplEx	0.844	✗
Vector	Asia	/locations/continents/countries_within	Kazakhstan	TransH	0.999	✓
	Shigeru Miyamoto	/architecture/architectural_style/architects	Mario & Sonic at the Olympic Games	DistMult	0.958	✗
	Gujarati	/language/human_language/countries_spoken_in	Uganda	TransE	0.871	✓
	Finnish	/location/location/containedby	Europe	TransH	0.843	✗
	James Wong Jim	/people/person/nationality	Hong Kong	ComplEx	0.832	✓

Table 4: ECE and link prediction accuracy by model in the open-world setting, before and after calibration. The translational models do not make any predictions at a confidence level over 0.80 before calibration.

	ECE (↓)		Accuracy (↑)	
	Uncalib.	Vector	Uncalib.	Vector
TransE	-	0.234	-	0.594
TransH	-	0.307	-	0.521
DistMult	0.618	0.344	0.308	0.509
ComplEx	0.540	0.291	0.293	0.581
Aggregate	0.548	0.296	0.295	0.549

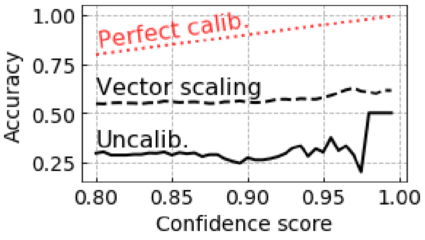


Figure 4: Reliability before and after calibration, aggregated across all four models.

kappa (Fleiss, 1971) is 0.7489 out of 1.

5.3 Results and discussion

Table 4 compares calibration error and link prediction accuracy before and after applying vector scaling. As shown in the table, the translational models do not make any uncalibrated predictions above a confidence level of 0.80 due to underconfidence, as dicussed in § 4.3. The bilinear models, DistMult and ComplEx, are much less calibrated off-the-shelf than under the CWA (c.f. Table 3).

Even after vector scaling, which reduces ECE significantly for both models and scales the scores of the translational models appropriately, **all models are overconfident**, collectively reaching around 50-60% accuracy at the 80-100% confidence level (Table 4 and Figure 4). This is consistent with observations of KGE overconfidence made by Pezeshkpour et al. (2020) for the task of triple classification, as well as observations on the general overconfidence of neural networks for vision and language processing (Guo et al., 2017).

We also do not observe any correlation between a model's level of exposure to a particular relation type and its calibration on that relation type. For example, all models achieve relatively low ECE ($< 4\%$) on the relation */language/human_language/countries_spoken_in*, which appears in only 0.148% of all triples in `FB15K-Wiki`. By contrast, for the relation */location/location/containedby*, which appears in 2.30% of all `FB15K-Wiki` triples (15× more frequent), all models are poorly calibrated both before and after vector scaling (ECE $> 10\%$). We discuss these results and behaviors in more detail next.

Challenges of the OWA Accurately calibrating KGE models (and evaluating calibration thereof) is challenging under the OWA for several reasons. First, in the CWA, all queries are known to have at least **one correct answer ahead of time**, whereas in the OWA we have no such guarantee. This highlights one of the fundamental challenges of the OWA, which is that of selecting predictions from a vast space of unknowns to maximize the probability of positives. It is likely that different strategies for selecting unknowns would lead to different observed levels of calibration.

In the OWA there is also a **mismatch between negatives** in the calibration and evaluation stages. Recall that in the calibration stage, we take completions not seen in the graph as negative samples (§ 3.3), which is essentially a closed-world assumption. By contrast, at evaluation time we make an open-world assumption. Higher-quality validation negatives may alleviate this problem; indeed, recent works have raised this issue and constructed new datasets toward this direction, albeit for the task of triple classification (Pezeshkpour et al., 2020; Safavi and Koutra, 2020).

Finally, our observation about the varying levels of calibration per relation suggests that **some relations are simply more difficult to** calibrate because of the knowledge required to accurately model them. Most popular "vanilla" KGE models do not explicitly make use of external knowledge that can help refine prediction confidences, such as entity types, compositional rules, or text.

Table 5 provides examples of high-scoring predictions made before and after calibration with corresponding labels. While most predictions are grammatically correct, it is perhaps not reasonable to expect KGE to capture certain types of semantics, logic, or commonsense using just the structure of the graph alone, for example that architects can design buildings but not video games (Table 5).

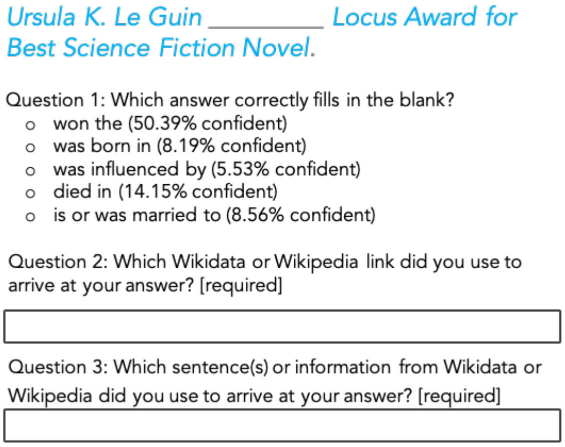
Link prediction accuracy As shown in Table 4, calibration with vector scaling on `FB15K-Wiki` improves OWA link prediction accuracy by 20-28 percentage points, which is significantly higher than under the CWA (c.f. Table 3), in which it improved accuracy by 1-5 percentage points on `FB15K-Wiki`. We conclude that from a practitioner's perspective, vector scaling is a **practical technique for making predictions more accurate and trustworthy** even if it does not perfectly calibrate models.

6 Case study

Finally, we conduct a case study of human-AI knowledge graph completion as a proof of concept on the benefits of KGE calibration for practitioners. In this experiment, given "fill-in-the-blank" sentences corresponding to incomplete knowledge graph triples, the task is to choose from multiple-choice answer lists generated by KGE to complete the sentences. We show that, compared to annotators *not* provided with confidence scores for this task, annotators provided with calibrated confidence scores for answer choices **more accurately and efficiently** complete triples.

6.1 Data

We construct a knowledge graph consisting of 23,887 entities, 13 relations, and 86,376 triples from Wikidata. We collect triples in which the head entity is categorized as a writer on Wikidata, and 13 people-centric relations (e.g., *born in*, *married to*). We extract our dataset directly from Wikidata to guarantee that all answers are resolvable using a single public-domain source of information. We choose writing as a domain because it is less "common knowledge" than, e.g., pop culture.



Ursula K. Le Guin _________ Locus Award for Best Science Fiction Novel.

Question 1: Which answer correctly fills in the blank?

- ○ won the (50.39% confident)
- ○ was born in (8.19% confident)
- ○ was influenced by (5.53% confident)
- ○ died in (14.15% confident)
- ○ is or was married to (8.56% confident)

Question 2: Which Wikidata or Wikipedia link did you use to arrive at your answer? [required]

Question 3: Which sentence(s) or information from Wikidata or Wikipedia did you use to arrive at your answer? [required]

Figure 5: Example completion task from our case study. The confidence scores shown in parentheses for Question 1 are presented to the confidence group only.

Task input After training and calibrating each KGE model from § 3.1 over the Wikidata graph, we use our best-calibrated model (ComplEx + Platt scaling, ECE < 0.01 under the CWA) to predict relations. Per triple, we take the top-five predicted relations $\{\hat{r}\}_{i=1}^{5}$ and their calibrated confidence scores $\{\hat{p}\}_{i=1}^{5}$. We filter these predictions to a sample of 678 triples by choosing only instances whose ground-truth relation r is in the top-five predictions $\{\hat{r}\}_{i=1}^{5}$, balancing class proportions, and discarding questions with answers that are easy to guess. Appendix C.1 provides more details.

6.2 Task setup

The task is to complete triples by choosing the correct relation among the top-five KGE-predicted relations $\{\hat{r}\}_{i=1}^{5}$, presented in natural language.

We conduct an A/B test whereby we vary how the confidence scores $\{\hat{p}\}_{i=1}^{5}$ for answer choices are presented to participants. We provide the **no-confidence (control) group** with multiple-choice answers in their natural language form without any accompanying confidence estimates, whereas the **confidence (treatment) group** is provided a calibrated confidence score along with each answer candidate in parentheses (Figure 5). We also provide the confidence group with an extra paragraph of instructions explaining that the confidence scores are generated by a computer system; Appendix C.2 provides the full task instructions.

To mitigate position bias, we randomize the pre-

Table 6: Case study results. ↑: Higher is better. ↓: Lower is better. **Bold**: Significant at $p < 0.05$. Underline: Significant at $p < 0.01$. *: p-value not applicable. Detailed explanations are given in § 6.3.

	Accuracy ↑			Sec. per
	Overall	Per triple	Per person	triple ↓
No-conf.	0.8977	0.8969	0.9120	36.88
Conf.	0.9175*	**0.9220**	**0.9478**	**31.91**
Abs. diff.	+0.0198	+0.0251	+0.0358	-4.97
Rel. diff.	+2.21%	+2.79%	+3.93%	-13.48%

sentation order of answer choices so that the answers are not necessarily ranked in order of confidence. The answer candidates are presented in the *same* randomized order for both groups.

Participants We recruit 226 participants for the no-confidence group and 202 participants for the confidence group from Figure 8. Participants are required to pass a 10-question "quiz" and maintain 50% minimum accuracy across all pre-labeled gold questions. We limit each participant to up to 20 judgments, and collect three judgments per triple.

6.3 Results and discussion

Table 6 summarizes the results of our case study. For the accuracy results, statistical significance is determined with the Wilcoxon rank-sum test (Wilcoxon, 1992) due to non-normality. For the efficiency results, statistical significance is determined with an independent two-sample t-test.

Accuracy The proportion of correct judgments in the no-confidence group was 0.8977 compared to 0.9175 in the confidence group, an improvement of 1.98 percentage points. In terms of the average judgment accuracy per triple, or the number of correct judgments divided by the number of judgments per triple, the no-confidence and confidence averages were 0.8969 and 0.9220 respectively, a significant difference ($p < 10^{-3}$). The average judgment accuracy per participant also differed significantly ($p < 10^{-6}$), again in favor of the confidence group.

Finally, model accuracy was 0.6268, meaning that for 62.68% (425/678) of triples seen by participants in the confidence group, the answer choice with the highest confidence score was the correct answer. Given that the confidence group's accuracy was much higher (0.9175 versus 0.6268), we can conclude that the participants in this group did not blindly trust the confidence scores.

Efficiency For this comparison we remove outliers with average judgment times more than two standard deviations away from the group mean. The mean time per judgment was 36.88 seconds in the no-confidence group (194 participants) versus 31.91 seconds in the confidence group (179 participants), a significant difference ($p = 0.010$). Note that we required sources and textual references for all answers across both groups (Questions 2 and 3 in the example in Figure 5). However, even with these quality control measures, the confidence group was significantly faster.

In conclusion, the results of our case study indicate that human-AI knowledge graph completion is more accurate and efficient with calibrated confidence scores generated by KGE. These findings suggest that calibrated probabilities are indeed trustworthy to practitioners, motivating the utility of calibration for human-AI tasks.

7 Conclusion

We investigate calibration as a technique for improving the trustworthiness of link prediction with KGE, and uniquely contribute both closed-world *and* open-world evaluations; the latter is rarely studied for KGE, even though it is more faithful to how practitioners would use KGE for completion. We show that there is significant room for improvement in calibrating KGE under the OWA, and motivate the importance of this direction with our case study of human-AI knowledge graph completion. As knowledge graphs are increasingly used as gold standard data sources in artificial intelligence systems, our work is a first step toward making KGE predictions more trustworthy.

Acknowledgments

We thank Caleb Belth for his feedback. This material is partially supported by the National Science Foundation under Grant No. IIS 1845491, Army Young Investigator Award No. W911NF1810397, and an NSF Graduate Research Fellowship.

References

Waleed Ammar, Dirk Groeneveld, Chandra Bhagavatula, Iz Beltagy, Miles Crawford, Doug Downey, Jason Dunkelberger, Ahmed Elgohary, Sergey Feldman, Vu Ha, et al. 2018. Construction of the literature graph in semantic scholar. In *Proceedings of NAACL-HLT*, pages 84–91.

Caleb Belth, Xinyi Zheng, Jilles Vreeken, and Danai Koutra. 2020. What is normal, what is strange, and what is missing in a knowledge graph: Unified characterization via inductive summarization. In *The Web Conference*, pages 1115–1126. ACM / IW3C2.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *Advances in neural information processing systems*, pages 2787–2795.

Tim Dettmers, Pasquale Minervini, Pontus Stenetorp, and Sebastian Riedel. 2018. Convolutional 2d knowledge graph embeddings. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Xin Dong, Evgeniy Gabrilovich, Geremy Heitz, Wilko Horn, Ni Lao, Kevin Murphy, Thomas Strohmann, Shaohua Sun, and Wei Zhang. 2014. Knowledge vault: A web-scale approach to probabilistic knowledge fusion. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 601–610.

Joseph L Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76(5):378.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. 2017. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1321–1330. JMLR. org.

Shaoxiong Ji, Shirui Pan, Erik Cambria, Pekka Marttinen, and Philip S Yu. 2020. A survey on knowledge graphs: Representation, acquisition and applications. *arXiv preprint arXiv:2002.00388*.

Rudolf Kadlec, Ondrej Bajgar, and Jan Kleindienst. 2017. Knowledge base completion: Baselines strike back. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 69–74.

Dong C Liu and Jorge Nocedal. 1989. On the limited memory bfgs method for large scale optimization. *Mathematical programming*, 45(1-3):503–528.

Robert Logan, Nelson F Liu, Matthew E Peters, Matt Gardner, and Sameer Singh. 2019. Barack's wife hillary: Using knowledge graphs for fact-aware language modeling. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5962–5971.

Alexandru Niculescu-Mizil and Rich Caruana. 2005. Predicting good probabilities with supervised learning. In *Proceedings of the 22nd international conference on Machine learning*, pages 625–632. ACM.

Pouya Pezeshkpour, Yifan Tian, and Sameer Singh. 2020. Revisiting evaluation of knowledge base completion models. In *Automated Knowledge Base Construction (AKBC)*.

John Platt et al. 1999. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3):61–74.

Daniel Ruffinelli, Samuel Broscheit, and Rainer Gemulla. 2020. You {can} teach an old dog new tricks! on training knowledge graph embeddings. In *International Conference on Learning Representations*.

Tara Safavi and Danai Koutra. 2020. Codex: A comprehensive knowledge graph completion benchmark. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*.

Tao Shen, Xiubo Geng, QIN Tao, Daya Guo, Duyu Tang, Nan Duan, Guodong Long, and Daxin Jiang. 2019. Multi-task learning for conversational question answering over a large-scale knowledge base. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2442–2451.

Fabian M Suchanek, Gjergji Kasneci, and Gerhard Weikum. 2007. Yago: a core of semantic knowledge. In *Proceedings of the 16th international conference on World Wide Web*, pages 697–706.

Pedro Tabacof and Luca Costabello. 2020. Probability calibration for knowledge graph embedding models. In *International Conference on Learning Representations*.

Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard. 2016. Complex embeddings for simple link prediction. In *International Conference on Machine Learning*, pages 2071–2080.

Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85.

Yanjie Wang, Daniel Ruffinelli, Rainer Gemulla, Samuel Broscheit, and Christian Meilicke. 2019. On evaluating embedding models for knowledge base completion. In *Proceedings of the 4th Workshop on Representation Learning for NLP (RepL4NLP-2019)*, pages 104–112.

Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. 2014. Knowledge graph embedding by translating on hyperplanes. In *Twenty-Eighth AAAI conference on artificial intelligence*.

Robert West, Evgeniy Gabrilovich, Kevin Murphy, Shaohua Sun, Rahul Gupta, and Dekang Lin. 2014. Knowledge base completion via search-based question answering. In *Proceedings of the 23rd international conference on World wide web*, pages 515–526.

Frank Wilcoxon. 1992. Individual comparisons by ranking methods. In *Breakthroughs in statistics*, pages 196–202. Springer.

An Yang, Quan Wang, Jing Liu, Kai Liu, Yajuan Lyu, Hua Wu, Qiaoqiao She, and Sujian Li. 2019. Enhancing pre-trained language representations with rich knowledge for machine reading comprehension. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2346–2357.

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 *International Conference on Learning Representations*.

Bianca Zadrozny and Charles Elkan. 2002. Transforming classifier scores into accurate multiclass probability estimates. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 694–699. ACM.

Table 7: Example sentence templates for relations in FB15K-Wiki. h: Head entity label. t: Tail entity label.

Relation	Template	Reverse template
/architecture/structure/architect	h was designed by the architect t.	t was the architect of h.
/cvg/computer_videogame/sequel	h is a video game with sequel t.	t is the sequel of the video game h.
/fight/crime_type/victims_of_this_crime_type	h is a crime that happened to t.	t was a victim of h.
/film/writer/film	h was a writer on the film t.	The film t's writers included h.
/food/diet_follower/follows_diet	h follows a diet of t.	t is a diet followed by h.
/government/government_agency/jurisdiction	h is a governmental agency with jurisdiction over t.	t is under the jursidiction of h.
/medicine/risk_factor/diseases	h has the risk of causing t.	t can be caused by h.
/people/person/nationality	h has or had t nationality.	t is the nationality of h.
/time/holiday/featured_in_religions	h is a holiday featured in the religion of t.	t is a religion that celebrates h.

A Implementation details

To select models, we grid search over the number of training epochs in {200, 300, 500}, the batch size in {100, 200, 500}, and the embedding dimension in {50, 100}. For training, we use random uniform negative sampling to speed up the training process. We search over the number of negative relations sampled per positive triple in {1, 5}.

We follow the original papers' choices of loss functions and optimizers. For loss functions, we use margin ranking for TransE, TransH, and DistMult and binary cross-entropy for ComplEx, and grid search over the margin hyperparameter in {1, 5, 10} for margin ranking. For optimizers, we use SGD for TransE and TransH, and Adagrad for DistMult and ComplEx, with a learning rate of 0.01.

We use the scikit-learn implementations of one-versus-all Platt scaling and isotonic regression[5], and implement vector and matrix scaling in Tensorflow with L-BFGS (Liu and Nocedal, 1989) limited to 2,000 iterations following the reference implementation provided by Guo et al. (2017).[6]

B Open-world evaluation

B.1 Data

To construct the set of triples for annotation, we discard relations pertaining to Netflix (e.g., */media_common/netflix_genre/titles*) to avoid disagreement due to crowd workers' countries of origin, since Netflix title availability varies widely by country. We convert all triples to sentences with a set of pre-defined relation templates. Because all relations can be reversed—e.g., (*Beyoncé*, *citizenOf*, *USA*) and (*USA*, *hasCitizen*, *Beyoncé*) express the same fact—we create two sentence templates for each relation and take the sentence that expresses the more plausible and/or grammatical statement per triple. Table 7 gives examples of sentence templates for relations in FB15K-Wiki.

B.2 Task instructions

This section gives the data annotation task instructions. Note that we conduct two separate annotation tasks: One with links to entities' Wikidata pages, and one with links to entities' IMDb pages for */film* relations only (Wikidata is linked to both Freebase and IMDb). The instructions are exactly the same between the two versions of the task, except that each instance of "Wikidata and/or Wikipedia" is replaced with "IMDb" in the latter.

Overview The goal of this task is to determine whether a given sentence is true or false.

Instructions Given a sentence that states a potentially true fact about the world, for example

> *Elizabeth Alexandra Mary Windsor is the queen of the Commonwealth.*

Read the sentence carefully and answer whether the sentence is factually correct by choosing one of *Yes*, *No*, or *Unsure*. To arrive at your answer, you **must use English-language Wikidata and/or Wikipedia**, even if you know the answer ahead of time. Each sentence already contains links to potentially relevant Wikidata pages; however, if you do not find an answer in the Wikidata page, you must check related Wikipedia pages. You **may not use any external data sources beyond English-language Wikidata or Wikipedia**. After you select your answer (Question 1), give the primary English-language Wikidata or Wikipedia URL (Question 2) and the text snippet or reasoning you used to arrive at your answer (Question 3).

Rules and Tips

- Read each sentence carefully and check both Wikidata and Wikipedia before choosing your answer.

[5] https://scikit-learn.org/stable/modules/generated/sklearn.calibration.CalibratedClassifierCV.html

[6] https://github.com/gpleiss/temperature_scaling

- **Question 1**: If a sentence is not grammatically correct, treat it as false. If a sentence is grammatically correct but you cannot find any information on Wikidata or Wikipedia supporting or disproving its claim, or you cannot reason about whether its claim is true or false, choose Unsure.
- **Question 2**: You must copy-paste the primary Wikidata or Wikipedia link that you used to arrive at your answer. **Only copy-paste the single link that contains the most complete answer to the question.** You may use the provided Wikidata links, but you **may also need to check related Wikipedia pages if you do not find what you are looking for. You may not use any external data sources beyond English-language Wikidata or Wikipedia.**
- **Question 3**: You may copy-paste relevant textual snippets from Wikidata or Wikipedia. If there is no relevant text to copy-paste, you may write a brief explanation of how you arrived at your answer.

Examples We give two examples presented to crowd workers in the task instructions.

1. *Nawaz Sharif is or was a leader of Pakistan.*
   - Is this sentence factually correct?
     - Yes
   - Which Wikidata or Wikipedia link did you use to arrive at your answer?
     - https://en.wikipedia.org/wiki/Nawaz_Sharif
   - Which sentence(s) or information from Wikidata or Wikipedia did you use to arrive at your answer?
     - "Mian Muhammad Nawaz Sharif is a Pakistani businessman and politician who served as the prime minister of Pakistan for three non-consecutive terms" - from the Wikipedia page of Nawaz Sharif
2. *The capital of France is or was Avignon.*
   - Is this sentence factually correct?
     - No
   - Which Wikidata or Wikipedia link did you use to arrive at your answer?
     - https://en.wikipedia.org/wiki/List_of_capitals_of_France
   - Which sentence(s) or information from Wikidata or Wikipedia did you use to arrive at your answer?
     - Avignon is not listed as a capital of France on the Wikipedia page about the capitals of France.

C Case study

C.1 Data

To convert all triples into natural language for the task, we map each relation in the dataset to a phrase: P19 (*was born in*), P20 (*died in*), P21 (*is of gender*), P26 (*is or was married to*), P101 (*works or worked in the field of*), P103 (*speaks or spoke the native language of*), P106 (*works or worked as a*), P119 (*is buried in*), P136 (*created works in the genre of*), P140 (*follows or followed the religion*), P166 (*was awarded the*), P551 (*lives or lived in*), and P737 (*was influenced by*).

We train each model on all triples that we collected from Wikidata, but limit the task input to a subset of triples for which the correct answer is unambiguous but also not easy to guess. To this end, we discard triples with relations that can be guessed via type matching: *Gender* (the tail entity is always *male* or *female* in our dataset), *award received* (the tail entity usually contains the word "award", "prize", etc.), and *place of burial* (the tail entity usually contains the word "cemetery"). We also discard triples with relations that can be interpreted as synonyms of one another (*occupation* and *genre*, e.g., "fiction writer"), and triples with the relation *field of work* for which the tail entity is synonymous with "writer" or "author", since all people in the dataset are categorized as authors or writers on Wikidata. Finally, we remove triples for which there is more than one correct answer in the top-five predicted relations.

C.2 Task instructions

This section gives the task instructions of the case study. Underline indicates that the enclosed text was presented to the confidence group only.

Overview The goal of this task is to complete a sentence so that it states a true fact about the world.

Instructions Given a partially complete sentence, fill in the blank with exactly one of the provided answer choices so that the sentence states a true fact about the world. To arrive at your answer, you **must use the provided Wikidata links**

in each sentence. You may not use any external data sources beyond the provided Wikidata links in each sentence. Please note that we have used a computer system to generate "confidence values" for each answer choice in order to help you with the task. These values signify our system's belief about which answer is most likely to be correct. After you select your answer (Question 1), give the single Wikidata URL (Question 2) and the text snippet or reasoning you used to arrive at your answer (Question 3). You must provide all answers in English.

Rules and Tips

- **Question 1**: Choose the answer that makes the sentence grammatically correct and factual according to Wikidata. Every sentence has at least one correct answer. If you believe a sentence has multiple equally correct answers, choose any of them.

- **Question 2**: You must copy-paste the single, entire Wikidata link that you used to arrive at your answer. The link that you copy-paste **must contain the correct answer that fills in the blank in the sentence**. You must use the Wikidata links provided in each sentence. You may not use any external data sources beyond the provided Wikidata links.

- **Question 3**: You may copy-paste relevant textual snippets from Wikidata. If there is no relevant text to copy-paste, you must write a brief explanation of how you arrived at your answer. You must provide all answers in English.

Examples We give two examples presented to crowd workers in the task instructions.

1. *Anna Akhmatova* ________________ *Leo Tolstoy.*

   (a) was or is married to (40% confident)
   (b) **was influenced by** (45% confident)
   (c) was the academic advisor of (5% confident)
   (d) was the child of (5% confident)
   (e) was the parent of (5% confident)

   - Which Wikidata link did you use to arrive at your answer?
     - https://www.wikidata.org/wiki/Q80440
   - Which sentence(s) or information from Wikidata did you use to arrive at your answer?
     - Wikidata says that Anna Akhmatova was influenced by Leo Tolstoy.

2. *Ursula K. Le Guin* ________________ *Hugo Award for Best Short Story.*

   (a) **was awarded the (40% confident)**
   (b) was influenced by (0% confident)
   (c) created the (50% confident)
   (d) was or is married to (10% confident)
   (e) lives in (0% confident)

   - Which Wikidata link did you use to arrive at your answer?
     - https://www.wikidata.org/wiki/Q181659
   - Which sentence(s) or information from Wikidata did you use to arrive at your answer?
     - Wikidata says that Ursula K Le Guin won the Hugo Award for Best Short Story.