

# Utilizing Language Model Probes for Knowledge Graph Repair

Hiba Arnaout  
Bosch Center for AI  
Renningen, Germany  
Max Planck Institute for Informatics  
Saarbrücken, Germany  
harnaout@mpi-inf.mpg.de

Trung-Kien Tran  
Bosch Center for AI  
Renningen, Germany  
trungkien.tran@de.bosch.com

Daria Stepanova  
Bosch Center for AI  
Renningen, Germany  
daria.stepanova@de.bosch.com

Mohamed H. Gad-Elrab  
Bosch Center for AI  
Renningen, Germany  
mohamed.gad-elrab@de.bosch.com

Simon Razniewski  
Max Planck Institute for Informatics  
Saarbrücken, Germany  
srazniew@mpi-inf.mpg.de

Gerhard Weikum  
Max Planck Institute for Informatics  
Saarbrücken, Germany  
weikum@mpi-inf.mpg.de

## ABSTRACT

Structured knowledge is an important backend in the Wikimedia ecosystem, and knowledge graphs (KGs) like Wikidata are an asset also in many other applications like web search and question answering. At web scale, it is unavoidable that KGs contain erroneous statements. While error detection and subsequent removal of incorrect facts have received attention in prior works, a better approach is to repair errors without losing information. This paper presents a novel method to repair incorrect statements in KGs by replacing incorrect subject-predicate-object (SPO) triples with likely correct ones, thus avoiding information loss. To this end, our method explores the power of LM probes for KG repair, and shows that context retrieval from the KG can significantly boost the probing. Specifically, we use the KG to augment LM probes so as to generate high-confidence values for the replacements of incorrect SPO triples. Experiments with Wikidata and DBpedia show that our method is viable and outperforms a prior baseline.

## CCS CONCEPTS

• Information systems;

## KEYWORDS

knowledge graphs, language models, Wikipedia

### ACM Reference Format:

Hiba Arnaout, Trung-Kien Tran, Daria Stepanova, Mohamed H. Gad-Elrab, Simon Razniewski, and Gerhard Weikum. 2022. Utilizing Language Model Probes for Knowledge Graph Repair. In *Companion Proceedings of the Web Conference 2022 (WWW '22 Companion)*, April 25–29, 2022, Virtual Event, Lyon, France. ACM, New York, NY, USA, 8 pages. <https://doi.org/10.1145/3487553.3524929>

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

*WWW '22 Companion, April 25–29, 2022, Virtual Event, Lyon, France*

© 2022 Copyright held by the owner/author(s).

ACM ISBN 978-1-4503-9130-6/22/04.

<https://doi.org/10.1145/3487553.3524929>

## 1 INTRODUCTION

**Motivation and Problem.** Structured Knowledge is important for many applications, including question answering and entity summarization. This resulted in a rise of interest in Knowledge Graph (KG) construction, querying, and maintenance. Prominent KGs like Wikidata [34], DBpedia [4], YAGO [31], or the Google Knowledge Graph [30], contain millions of entities and billions of facts about those entities. With such web-scale knowledge repositories, it is inevitable that KGs include some erroneous information.

For example, DBpedia contains<sup>1</sup> the following two triples (extracted from Wikipedia infobox as illustrated in Figure 1), about the scientist Jessica Meeuwig: (Jessica Meeuwig, field, Marine Science) and (Jessica Meeuwig, almaMater, Montreal). While the former triple is correct, the latter is erroneous, since, according to the KG schema, it violates the type-signature of the predicate `almaMater`, which as an object expects an educational organization rather than a city.

**State of the Art and its Limitations.** The problem of automatically fixing erroneous triples is challenging, especially in large-scale KGs. Existing work for automatically detecting errors by checking if KGs violate constraints of the schema [24, 33] or discovering vandalism in real time [14, 29] rarely fix these error after identifying them. While the problem of detecting errors has been considered widely in the area of knowledge representation and reasoning [7], the available repair methods mainly result in the undesired information loss caused by the data removal.

In collaborative KGs, like Wikidata, incorrect information is manually repaired by editors; this is a very time-consuming and labor-intensive task. Recently proposed neural-based methods for repairing incorrect triples in KGs do not require manual efforts, but they typically rely on large amounts of training data (e.g., correction history [25]), which is rarely available in practice for automatically constructed KGs.

**Approach and Contribution.** In this paper, we exploit the advances of language models (LMs) to repair pre-detected incorrect KG triples. Probing LMs for knowledge is not a completely novel idea [18], yet so far it has largely focused on artificial benchmarks. Investigating its potential for actual repairs, and formalizing useful probes, remains challenging. After converting incorrect triples into

<sup>1</sup>As of September, 2021.

Property	Value
<a href="#">dbp:abstract</a>	<ul style="list-style-type: none"> <li>Professor Jessica Meeuwig appointed as a Conservation Fellow of the Zoological Society of London (appointed 2012)</li> </ul>
<a href="#">dbp:field</a>	<ul style="list-style-type: none"> <li><a href="#">dbp:Marine_Science</a></li> </ul>
<a href="#">dbp:almaMater</a>	<ul style="list-style-type: none"> <li><a href="#">dbp:Montreal</a></li> </ul>

Jessica Meeuwig	
<b>Nationality</b>	Dual national: Canada and Australia
<b>Alma mater</b>	McGill University, <a href="#">Montreal</a> , Quebec, Canada
<b>Known for</b>	Campaigning for Marine Conservation
<b>Awards</b>	Conservation Fellow of the Zoological Society of London (appointed 2012)
<b>Scientific career</b>	
<b>Fields</b>	<a href="#">Marine Science</a>
<b>Institutions</b>	University of Western Australia

Figure 1: Structured Information about Jessica Meeuwig, extracted from her Wikipedia Infobox.

LM probes, we augment the probes with salient context from the input KG. We show that LMs used naively alone [27] are not sufficient for this task. In a nutshell: given a KG, its accompanying schema, and an incorrect triple (i.e., assumed as a subject-predicate pair  $S$ - $P$  with an incorrect object  $O$ ) identified using a blackbox method (e.g., [33]), we construct a LM probe, where  $O$  is masked. We then compile salient information about  $S$  from the KG and extend the LM probe with it to enhance the context. Retrieved predictions are ranked by their confidence scores. They are then automatically validated using the KG schema to fit the appropriate type(s). Finally, the top correction is used for repairing the triple.

The contributions of this work are as follows:

- (1) We introduce a framework for repairing incorrect triples in KGs, without removing them, by exploiting the strengths of LM probes.
- (2) We define the notion of *salient* KG context to improve the quality of LM probes.
- (3) We evaluate our approach on large scale real-world KGs Wikidata and DBpedia by comparing it to the prior baselines for probing factual knowledge [18].

## 2 PRELIMINARIES

A Knowledge Base (KB) consists of a schema and an extension, in which the extension is a set of facts and usually called a knowledge graph while the schema provides a type taxonomy/ontology and type constraints. We describe these components as follows.

**Knowledge Graph (KG).** A KG is a finite set of *triples* of the form  $(S, P, O)$ , where  $S, O$  are *entities* and  $P$  is a *property* (aka *predicate*). For example, the triple (Jessica Meeuwig, nationality, Canadian) states that the entity Jessica Meeuwig has the relation nationality with the entity Canadian.

**KG Schema.** We use ontologies following the Web Ontology Language (OWL 2) standard<sup>2</sup> as the schemas for the KGs. In particular, we utilize the *domain axiom* `ObjectPropertyDomain` and the *range axiom* `ObjectPropertyRange` to specify the type signatures for the

subjects and objects of the properties, respectively. For example, the axiom `ObjectPropertyDomain(nationality, Person)` specifies that the subject  $S$  in each  $(S, nationality, O)$  is of the type `Person`. Additionally, we use the *subclass axiom* `SubClassOf` to specify that a type, e.g. `Director`, is a sub-type of another type, e.g. `Person`. We also use the *disjoint class axiom* `DisjointClasses` to state that several types are disjoint.

**Language Models.** A *Language Model* (LM) is a language representation model that has been trained to learn a distributed representation for words/symbols [5]. While an LM can be used in different tasks, in this work we leverage a pre-trained LM, e.g. based on the Transformer architecture, to predict missing words in sentences.

**Erroneous Triple.** An *erroneous triple*  $t = (S, P, O)$  is a triple in  $\mathcal{G}$ , identified as incorrect using a blackbox method or by crowd sourcing, where the value of  $O$  component is false (i.e. to be fixed).

**Research Problem.** Given a KG  $\mathcal{G}$ , with its schema, and an incorrect triple  $t = (S, P, O)$ , the goal of our work is to repair  $t$  by replacing  $O$  with a correct alternative.

## 3 METHODOLOGY

**Overview.** We propose a method to repair incorrect KG triples utilizing pre-trained language models (LMs) as a source of corrections. Most LMs are transformer-based neural networks with billions of parameters, usually trained on the full text of Wikipedia and other high-quality text corpora. They can latently represent factual knowledge [18], and have been proposed as a source for completing or predicting SPO statements, by a mechanism called LM probing [27]. For instance, when looking for the birth place of Alan Turing, the LM can be probed by the masked string (aka cloze question): “Alan Turing was born in the city of [MASK]” or just “Alan Turing was born in [MASK]”.

However, when used out of the box, LMs have substantial shortcomings, and thus cannot be naively used for KG repair [10, 28]:

- They struggle to make correct predictions for short probes (i.e., masked sentences with no or short context) [18].

<sup>2</sup><https://www.w3.org/TR/owl2-overview/>

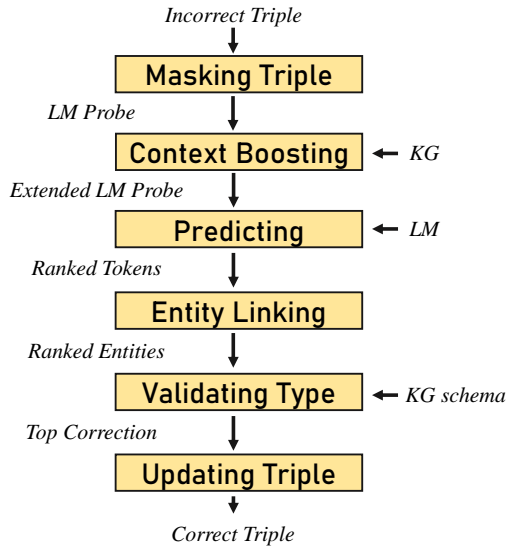


Figure 2: Methodology.

- The types and values of predictions are biased towards frequent types and values in the input corpus. For example, for English sounding names, suggested places of birth are very frequently New York, London, etc., especially for longer-tail entities that were not seen often in pre-training.
- The predicted tokens can upfront be of arbitrary type, so one has to look for the specific answer-type in the list of returned predictions. For instance, if one were to probe for Jessica Meeuwig’s profession using “Jessica Meeuwig is a [MASK].”, LMs return mixed set of types as a result (*biologist, soprano, feminist, canadian*). One needs to do further processing or filtering to retrieve valid answers, in this case *biologist*.

While similar to other works [10, 18, 27], we propose to utilize LM probes, our main novelty is that we focus specifically on repairing errors in KGs, and that our method judiciously expands the LM probes with *salient context* from the KG.

As input, we assume to have an incorrect triple  $t = (S, P, O)$  from a KG  $\mathcal{G}$ , a KG schema, and a pre-trained language model  $\mathcal{LM}$ . First, our method masks the object of  $t$ . The LM probe is then augmented with the context from  $\mathcal{G}$  about the subject of the triple, prior to querying the  $\mathcal{LM}$ . After top predictions (i.e., tokens) are retrieved from the LM, we map them to entities in  $\mathcal{G}$ . The resulting candidate objects are scrutinized by type checks using the KG schema. Finally, the incorrect triple is replaced by the best validated correction. Figure 2 gives a visual overview of our method.

### 3.1 Probe Construction

We construct the LM probe for repairing the incorrect  $(S, P, O)$  triple in the following three steps:

- (1) The object  $O$  of the triple is masked (i.e.,  $(S, P, [\text{MASK}])$ ).
- (2) The subject  $S$  of the triple is converted into natural language using labels provided by the considered KG (e.g., DBpedia’s entity `dbp:Jessica_Meeuwig` has the label “Jessica Meeuwig”).

- (3) The predicate  $P$  is converted to natural language using i) textual parsing by splitting at capital letters (e.g., `almaMater` → “alma mater”), or ii) exploiting labels of the same relation from other KGs (e.g., DBpedia’s `almaMater` → Wikidata’s “educated at”). Predicate labels in Wikidata are rich and often include, on top of the main label, other alternatives. For instance, `P69`’s main label is “educated at”, and other labels include “education, alumni of, studied at”. In this case, we use the *main label*.

*Example 3.1.* Given  $\mathcal{G} = \text{DBpedia}$ <sup>3</sup> and the incorrect triple  $t = (\text{dbp:Jessica\_Meeuwig}, \text{almaMater}, \text{Montreal})$ , the object of the triple `Montreal` must be fixed. This results in the probe “Jessica Meeuwig educated at [MASK].”

### 3.2 Probe Expansion

We improve the constructed probe by expanding it with context from the KG. All the incorrect triples identified using the blackbox method are disregarded prior to this step. Therefore, we assume that most of the remaining statements are correct. Some subject entities have a large number of associated facts, though, and it is not viable to include all of them into an LM probe. Thus, we propose several ways of defining *salient* context as follows.

**Salient Types.** A natural source of background knowledge for generating context of LM probes are the types of a given subject. However, an entity can have many types (e.g., `Cher` has 125 types in DBpedia), where some of them are more salient (*Artist*) than others (*Natural Person*). We first eliminate types with too many instances (types shared by  $> 80\%$  of the entities) or too few ( $< 0.1\%$ ). From the remaining candidate types, we select the best ones in two steps:

- (1) Obtain peers: We collect  $n$  entities that are highly related to  $S$  by relying on various graph-based measures such as distance in the KG or link-sharing (i.e., the number of PO pairs which have  $S$  and its peer-entity in common).
- (2) Select frequent types: a list of types shared by peers is compiled with their relative frequency (e.g., 70% of peers are scientists, 10% are activists). Finally, the  $k$  most frequent types are considered for the LM probe, where  $k$  is a tuning parameter (set to 1 in experiments).

*Example 3.2.* We collect three peers ( $n=3$ ) for Jessica Meeuwig using graph-based measures, namely Stephen Calvert, James Syvitski and John Murray. They all share the same nationality and profession, (nationality-Canadian) and (academicDiscipline-Oceanography). The top shared type between the peers is retrieved ( $k=1$ ), in this case, *Scientist*, with relative frequency=100% (i.e., all of her peers are scientists).

**Salient Triples.** On top of the types, KGs offer facts about entities such as birthplace, profession etc. Again, the issue is a possibly overwhelming number of triples, including many uninteresting facts. To identify top  $m$  salient triples ( $m$  set to 2 in experiments), we are inspired by related work on entity summarization [2, 20] for scoring a given triple by the *informativeness* of its components.

$$\text{INF}(S, P, O) = \frac{\text{POP}(S) + \text{FRQ}(P) + \text{POP}(O)}{3} \quad (1)$$

<sup>3</sup>All the examples used in this paper have been selected as of September 2021.

	Non-salient context	Salient context
<b>Textual Summary</b>	Professor Jessica Meeuwig is the inaugural director of the Centre for Marine Futures at the University of Western Australia.	Professor Jessica Meeuwig is the inaugural director of the Centre for Marine Futures at the University of Western Australia.
<b>Types</b>	Jessica Meeuwig is a person.	Jessica Meeuwig is a scientist.
<b>Triples</b>	Jessica Meeuwig is winner of Zoological Society of London prize.	Jessica Meeuwig is a citizen of Canada.

**Table 1: Various kinds of context for the probe “<context>. Jessica Meeuwig educated at [MASK]”.**

where POP(S), FRQ(P), and POP(O) reflect the popularity of the subject, frequency of the predicate, and popularity of the object, respectively. We compute the popularity of S and O using an external source, namely, the number of page-views of their respective Wikipedia articles over  $w$  years ( $w = 5$  in examples and experiments). For predicates, the informativeness is computed as the frequency of P in  $\mathcal{G}$  (e.g., 4.3m triples about citizenships v. 314k about Twitter usernames in Wikidata). Moreover, we exclude predicates reflecting meta-information about web pages like `wikiPageWikiLink`, `wikiPageLength`, etc. Both popularity and frequency metrics are normalized by the maximum value possible (i.e., average number of views of the *top viewed* Wikipedia articles over the last  $w$  years and the most frequent predicate in  $\mathcal{G}$  respectively).

*Example 3.3.* To select *interesting facts* about Meeuwig, we collect the set of triples (Meeuwig, P, O), where P is a KG predicate and rank them using Equation 1. More specifically, we instantiate POP(S) and POP(O) to the average number of page-views<sup>4</sup> over the last 5 years, in their respective Wikipedia articles. Moreover, we compute FRQ(P) as the frequency of the predicates in DBpedia. For instance, the triple (Jessica Meeuwig, nationality, Canada) has the following score:  $(\text{views}(\text{Jessica Meeuwig})/\text{views}(\text{Top-pages}) + \text{freq}(\text{nationality})/\text{freq}(\text{wikiPageWikiLink}) + \text{views}(\text{Canada})/\text{views}(\text{Top-pages}))/3 = (443/43\text{m} + 150\text{k}/149\text{m} + 7.2\text{m}/43\text{m})/3 = 0.056$ . We have that the above triple is more informative than (Jessica Meeuwig, award, Zoological Society of London) with the informativeness score of  $(443/43\text{m} + 71\text{k}/149\text{m} + 21\text{k}/43\text{m})/3 = 0.0001$ .

**Textual Summary.** Similar to salient triples, another way for retrieving informative context is to query short textual summaries most Wikipedia-based KGs provide. For instance, Jessica Meeuwig’s textual summary in DBpedia is shown in Table 1 along with other context variations. We limit the size of this context to the first 50 tokens<sup>5</sup>. This is needed for prominent entities with long descriptions (e.g., 427 tokens for Samuel L. Jackson). Table 1 presents different contexts for retrieving the education institute of Jessica Meeuwig.

### 3.3 Predicting, Linking, and Validating

**Predicting.** At this point, we have constructed the expanded LM probe. The LM normally returns a list of single-token predictions,<sup>6</sup> in the form of surface tokens, ranked by confidence.

*Example 3.4.* The expanded LM probe, with the added context (underlined), for our running example is: “Jessica Meeuwig is a

<sup>4</sup><https://pageviews.toolforge.org/>

<sup>5</sup>We observe that the first two sentences are the most informative. This choice can be easily adjusted.

<sup>6</sup>Multi-token predictions are more challenging for BERT-like models, though this is an active field of research [15].

scientist. Jessica Meeuwig citizen of Canada. Jessica Meeuwig educated at [MASK]”. Using the pre-trained LM RoBERTa [21], we retrieve the following ranked list of tokens (with their confidence) among top-20:

- (1) *mcGill*: 0.17
- (2) *university*: 0.08
- (3) *harvard*: 0.07
- (6) *ucla*: 0.03
- (18) *canada*: 0.01
- (20) *manitoba*: 0.008

**Entity Linking.** In order to suggest a canonical entity as a correction, we map the top-ranked tokens to KG entities. Hence, non-entity mentions and noisy tokens are eliminated. For that, we use off-the-shelf entity linking component, namely, Wikipedia2Vec [38]<sup>7</sup>. The used API allows retrieving possible entities matching the name.

*Example 3.5.* The top ranked token *mcGill* is mapped to the KG entity McGill University.

While this approach performs well in most of the cases, it is worth noting that it is not optimal in the case of having several strong matching entities. We partially overcome this problem by the the automatic type-validation step. Adopting more suitable entity linking component is a subject for future investigation.

**Validating Type.** Finally, we ensure that the resulting object entity is of correct type, by checking it against the KG schema.

*Example 3.6.* In this case for  $P=\text{almaMater}$ , the type of  $O$  is `educationalInstitution`, thus the top repair McGill University is valid.

## 4 EXPERIMENTS


We assess the quality of our probes against the baseline where LMs are probed with short context [18]. We conduct study cases over 2 large KGs, with 5 different kinds of contexts, and report numerical and qualitative results.

### 4.1 Setup

**Datasets.** We use the following relevant datasets.

- *Wikidata* is a collaborative human-curated KG, containing billions of triples. Due to the increased risk of spreading falsified information, several methods for automatically detecting vandalism have been proposed [14, 29]. These methods, however, only detect wrong triples, but do not fix them. We propose to use our LM probes for *automatically* repairing detected vandalized triples.

<sup>7</sup>API available at <https://wikipedia2vec.github.io/wikipedia2vec/usage/>

LM Probe	Overall	Topics		Predicates				
		locations	organizations	locatedIn	nationality	employer	hometown	headquarter
								
brief context [18]	0.19	0.26	0	0.33	0.40	0	0.18	0.50
w/ types (random)	0.19	0.26	0	0.33	0.20	0	0.21	0.53
w/ types (salient)	0.19	0.28	0	0.44	0.40	0	0.21	0.53
w/ triples (random)	0.30	0.42	0.04	0.33	0.60	0.09	0.25	0.73
w/ triples (salient)	<b>0.40</b>	<b>0.55</b>	<b>0.13</b>	<b>0.78</b>	0.80	<b>0.27</b>	<b>0.46</b>	0.77
w/ textual summary	0.38	<b>0.56</b>	0.09	0.56	<b>1.0</b>	0.18	0.41	<b>0.80</b>
<i>Diff.</i>	<b>+0.21</b>	<b>+0.30</b>	<b>+0.13</b>	<b>+0.45</b>	<b>+0.60</b>	<b>+0.27</b>	<b>+0.28</b>	<b>+0.30</b>


LM Probe	Overall	Topics		Predicates				
		locations	professions	country	capital	shares border	occupation	religion
								
brief context [18]	0.26	0.27	0.30	0.44	0.38	0.20	0.23	0.50
w/ types (random)	0.42	0.40	0.20	0.51	0.56	0.40	0.13	0.50
w/ types (salient)	<b>0.53</b>	0.42	<b>0.64</b>	0.60	0.57	0.40	<b>0.61</b>	0.50
w/ triples (random)	0.39	0.40	0.38	0.61	0.44	0.20	0.34	0.42
w/ triples (salient)	0.45	0.47	0.45	0.63	<b>0.63</b>	0.40	0.42	0.58
w/ textual summary	<b>0.53</b>	<b>0.57</b>	0.57	<b>0.74</b>	0.56	<b>0.60</b>	0.53	<b>0.75</b>
<i>Diff.</i>	<b>+0.27</b>	<b>+0.30</b>	<b>+0.34</b>	<b>+0.30</b>	<b>+0.25</b>	<b>+0.40</b>	<b>+0.38</b>	<b>+0.25</b>

Table 2: Evaluation (p@1) of KG repairs using different LM probes.

For that, we randomly sample 500 triples from the Wikidata Vandalism dataset.<sup>8</sup> Examples include (Andrew Jackson [Q11817], occupation [P106], Liar [Q1049271]) and (Beyoncé [Q36153], gender [P21], Alien [Q103569]).

• *DBpedia* is a KG automatically constructed from Wikipedia infoboxes. Due to automatic construction, DBpedia still contains erroneous triples, (similar to those in Figure 1), in which the object is of wrong type, i.e., for (S, P, O), the type of O is disjoint with the range of P. We randomly sample 500 triples where the type of the object violates the schema. One example is (The System, instrument, David Frank) where the band member has been mistaken for one of the instruments.<sup>9</sup> The contradiction can be automatically detected by retrieving the types allowed for predicate instrument and comparing them with the object’s type. In this case the types Instrument and Person are disjoint. Another example is (WRVN, sisterStation, Hamilton New York), where sisterStation expects an entity of type Broadcaster but received a City instead.

**Baselines.** We compare our method to the one, in which probes with no context haven been used [18] (e.g., “Kuching is the capital of [MASK].”). For a fair comparison, we also apply our linking and validating steps to the baseline. Moreover, we consider the following configurations of our method which differ in terms of the generated context.

- (1) Random Types:  $k$  random types about S.
- (2) Types:  $k$  relevant types about S.
- (3) Random Triples:  $m$  random triples about S.

- (4) Triples:  $m$  salient triples about S.

- (5) Textual Summary: first 50 tokens of the summary of S.

We initialize  $k$  to 1 and  $m$  to 2.

**Language Model.** The main goal of our work is to evaluate the impact of probes on the quality of the computed repairs using a fixed LM in its zero-shot setting. For this, we pick one of the most prominent LMs, RoBERTa [21]: a pre-trained model on English language (Wikipedia, 11k books, 63 millions crawled news articles, etc.) using a masked language modeling (MLM) objective. In preliminary experiments, we consider BERT as a second LM. Results are comparable with a slight advantage for RoBERTa over the baseline. **Evaluation Metric.** For the task of KG repair, the top repair matters the most. For this reason, we use the standard precision@1 (p@1) as our evaluation metric.

## 4.2 KG Repair using LM Probes

We run our model to repair the 1k incorrect triples described in Section 4.1. In particular, we exploit the proposed 5 LM probes (additionally to the initial short probe) and assess the top-1 repair. We evaluate a total of 6k repaired triples, i.e, 2 KGs \* 500 incorrect triples \* 1 LM \* 6 probes \* top-1 prediction, and report the overall average p@1 in Table 2 for both DBpedia and Wikidata. It is clear that adding *salient* context outperforms the baseline, by 21 percent for DBpedia and 27 for Wikidata. Probes augmented with *salient facts about the subject* are most useful context for DBpedia, while probes with *salient types & textual summaries* are the most effective for Wikidata. We illustrate various examples in Tables 3 and 5.

<sup>8</sup><https://www.wsdm-cup-2017.org/vandalism-detection.html>

<sup>9</sup>[https://en.wikipedia.org/wiki/The\\_System\\_\(band\)](https://en.wikipedia.org/wiki/The_System_(band))

<b>Incorrect triple:</b> (C.L._Bryant, almaMater, Shreveport)		
<b>Probe</b>		
Brief [18]	probe repair	C.L. Bryant graduated from [MASK]. <token: Harvard, entity: Harvard_University>
Triples ( <i>rand</i> )	probe repair	C.L. Bryant parents L. C. & Elnola Bryant. C.L. Bryant same as Q5006160. C.L. Bryant graduated from [MASK]. <token: UCLA, entity: University_of_California_Los_Angeles>
Triples ( <i>sal</i> )	probe repair	C.L. Bryant is a Baptists. C.L. Bryant place of birth Shreveport, Louisiana. C.L. Bryant graduated from [MASK]. <token: LSU, entity: Louisiana_State_University> ✓
<b>Incorrect triple:</b> (Hanns_Johst, militaryBranch, World_War_II)		
Brief [18]	probe repair	Hanns Johst member of [MASK]. <token: Greenpeace, entity: Greenpeace>
Types ( <i>rand</i> )	probe repair	Hanns Johst is an official. Hanns Johst member of [MASK]. <token: FIFA, entity: FIFA>
Types ( <i>sal</i> )	probe repair	Hanns Johst is a military person. Hanns Johst member of [MASK]. <token: NATO, entity: NATO>
Triples ( <i>rand</i> )	probe repair	Hanns Johst page revision id "705907257". Johst died in Bavaria. Hanns Johst member of [MASK]. <token: IEEE, entity: Institute of Electrical and Electronics Engineers>
Triples ( <i>sal</i> )	probe repair	Hanns Johst allegiance German Empire. Johst winner of SS-Ehrenring. Hanns Johst member of [MASK]. <token: SS, entity: Schutzstaffel> ✓
<b>Incorrect triple:</b> (Epigram, headquarter, University_of_Bristol_Union)		
Brief [18]	probe repair	Epigram headquartered in [MASK]. <token: Town, entity: Town>
Types ( <i>rand</i> )	probe repair	Epigram is a periodical literature. Epigram headquartered in [MASK]. <token: Kolkata, entity: Kolkata>
Types ( <i>sal</i> )	probe repair	Epigram is a newspaper. Epigram headquartered in [MASK]. <token: Dhaka, entity: Dhaka>
Triples ( <i>rand</i> )	probe repair	Epigram foundation 1988. Epigram website http://www.epigram.org.uk. Epigram headquartered in [MASK]. <token: London, entity: London>
Triples ( <i>sal</i> )	probe repair	Epigram subject University of Bristol Union. Epigram edited by Connolly and Coward. Epigram headquartered in [MASK]. <token: Bristol, entity: Bristol> ✓
Text. Summ.	probe repair	Epigram, newspaper of the Uni. of Bristol, established by Landale, who studied politics at Bristol. Epigram headquartered in [MASK]. <token: Bristol, entity: Bristol> ✓

**Table 3: Sample repairs in DBpedia with different LM probes (*rand* =random, *sal*=salient).**

To give more insights, we report the performance of probes over different topics and predicates. We define two major recurring themes in each dataset (in other words, different object-families). For every topic, we define the most frequent predicates (see Table 4). For example, the topic locations include predicates where the LM is probed for a place, including cities, countries, etc. The topic organizations covers predicates where the prediction is an organization of some sort, including universities, companies, etc.

The best results for DBpedia are achieved for the topic of *locations* and for Wikidata for the topic of *professions*. We attribute this to the relatively small search space for such predicates, and the ability to be identified using only one token (e.g., Germany, London, Lawyer, Actor). The more challenging topic is *organizations*, for which the baseline as well as two of our probes fail to make correct predictions. We notice, however, that our textual summary and probes with salient triples outperform other probes with organizations such as universities and companies (see *employer* in Table 2 and examples of Hanns Johst & C. L. Bryant in Table 3). Additionally to topics, we consider various individual predicates. One particular predicate that stood out for the DBpedia triples is *nationality*, where our probes outperform the baseline by 60 and 40 percent using the textual summary probe and probes with salient triples respectively.

For Wikidata, the relation with the most impressive improvement over the baseline is *shares border* (see Chile’s example in Table 5).

## 5 DISCUSSION

**LM Biases.** One major challenge when using LMs for our target as well as other tasks is concerned with their bias towards training data. We observed that probes with predicates about places of birth and hometowns with entities from the U.S. (especially politicians) in 28% of the cases result in Chicago being the top prediction. Our method deals with such biases, with the help of context-augmented probing, reducing its appearance as the top-prediction in this case to 7%. The same holds for predicates about organizations such as almaMater where the probes often return MIT and Harvard as the top prediction (88% of the queries with *short* context, and 19% when the salient context is added).

**Corner Cases.** We observe that some incorrect triples are practically *unrepairable* via LM probes. These can be grouped into the following three categories:

- *The triple is factually correct but breaks the type constraint.* For instance, consider the triple (Deutschland Ein Sommermärchen, starring, Germany national football team) in DBpedia originated from its Wikipedia infobox.<sup>10</sup> It is obvious for a human

<sup>10</sup>[https://en.wikipedia.org/wiki/Deutschland\\_Ein\\_Sommerm%C3%A4rchen](https://en.wikipedia.org/wiki/Deutschland_Ein_Sommerm%C3%A4rchen)

Topic	KG	Predicates
locations	DBpedia	state, sourceCountry, nearestCity, locationCountry, country, locatedIn, locationCity, hometown, city, county
organizations	DBpedia	almaMater, employer, network, manufacturer, parentOrganisation, governingBody, distributingLabel, company, recordLabel
locations	Wikidata	country, place of birth, country of citizenship, continent, capital, shares border with, country of origin, location
professions	Wikidata	position held, field of work, occupation, genre

Table 4: Predicates used in evaluation over subsets of triples.

Vandalized triple: (Leonardo da Vinci, occupation, gay)		
Probe		
Brief [18]	probe repair	Leonardo da Vinci is a [MASK]. <token: Hero, entity: Hero>
Triples (rand)	probe repair	Leonardo da Vinci CCAB ID 000045477. Da Vinci has works in the collection Victoria & Albert Museum. Da Vinci is a [MASK]. <token: DJ, entity: Disc_jockey>
Triples (sal)	probe repair	Leonardo da Vinci notable work Mona Lisa. Da Vinci genre religious painting. Da Vinci is a [MASK]. <token: Painter, entity: Painter> ✓
Text. Summ.	probe repair	Leonardo da Vinci is Italian Renaissance polymath (1452-1519). Da Vinci is a [MASK]. <token: Painter, entity: Painter> ✓
Vandalized triple: (Oscar Wilde, place of birth, Berlin)		
Brief [18]	probe repair	Oscar Wilde was born in [MASK]. <token: Chicago, entity: Chicago>
Types (rand)	probe repair	Oscar Wilde is a human. Oscar Wilde was born in [MASK]. <token: Chicago, entity: Chicago>
Types (sal)	probe repair	Oscar Wilde is a writer. Oscar Wilde was born in [MASK]. <token: London, entity: London>
Triples (sal)	probe repair	Oscar Wilde ethnic group Irish people. Oscar Wilde works in comedy. Oscar Wilde was born in [MASK]. <token: Dublin, entity: Dublin> ✓
Text. Summ.	probe repair	Oscar Wilde is Irish writer and poet (1854-1900). Oscar Wilde was born in [MASK]. <token: Dublin, entity: Dublin> ✓
Vandalized triple: (Chile, shares border with, England)		
Brief [18]	probe repair	Chile shares borders with [MASK]. <token: Brazil, entity: Brazil>
Types (rand)	probe repair	Chile is a sovereign state. Chile shares borders with [MASK]. <token: Argentina, entity: Argentina> ✓
Types (sal)	probe repair	Chile is a country. Chile shares borders with [MASK]. <token: Argentina, entity: Argentina> ✓
Triples (rand)	probe repair	Chile railway traffic side left. Chile has diplomatic relations with Indonesia. Chile shares borders with [MASK]. <token: Indonesia, entity: Indonesia>
Triples (sal)	probe repair	Chile is part of Latin America. Chile's official language is Spanish. Chile shares borders with [MASK]. <token: Peru, entity: Peru> ✓
Text. Summ.	probe repair	Chile is sovereign state in South America. Chile shares borders with [MASK]. <token: Argentina, entity: Argentina> ✓

Table 5: Sample repairs in Wikidata with different LM probes (rand =random, sal=salient).

annotator that the *members* of the football team starred in the respective documentary, rather than the team as an entity.

- *Controversial topics.* Especially in collaborative KGs, controversial facts are constantly updated and might be identified as vandalism. In preliminary crowdsourcing studies using our probes, the repair for the triple “Jerusalem is the capital of [MASK].” received very low agreement.
- *No valid correction exists.* In some cases, no repair at all exists for a questionable triple. For instance, John Duff’s infobox<sup>11</sup> states that his resting place is *unknown*, which has been identified as an erroneous triple in DBpedia.

<sup>11</sup>[https://en.wikipedia.org/wiki/John\\_Duff\\_\(counterfeiter\)](https://en.wikipedia.org/wiki/John_Duff_(counterfeiter))

Overall, our method presents promising results and demonstrates that LM probes with the proposed context variations can be effectively exploited for supporting human KG curators in fixing erroneous facts automatically. While this work focuses on the English language, future work should examine on other languages.

## 6 RELATED WORK

**Wikidata Quality.** Wikidata’s quality is maintained by a combination of collaborative inspection and automated quality checks. A standard quality assurance mechanism in the Wikimedia ecosystem is the public nature, and manual inspection of content, which works especially for popular topics. Due to Wikidata’s size, these are complemented by a range of automated approaches, notably the

ORES<sup>12</sup> automated quality scoring and constraint checks. Error and vandalism detection has received considerable research attention as well (e.g., via the Wikidata Vandalism Detection challenge [13]).

**KG Completion.** To tackle the problem of missing information, several popular approaches have been proposed, including knowledge graph embeddings (KGE) [35], rule learning [37] and their combination [22]. While KGE methods mainly rely on embedding relations and entities of a KG into continuous vector spaces, rule learning methods are widely used for pattern discovery in KGs (e.g., AMIE [17]). To date, the goal of these methods is to predict new links in the KG rather than repair erroneous ones.

**KG Repair.** Cleaning and repairing KGs typically concerns removing incorrect information, and in some rare cases replacing it with correct information. To this end, the closest work to ours is [25], in which the authors propose to repair Wikidata using its edit history. More precisely, the paper proposes a deep learning model that exploits edits that removed inconsistent triples in the past to infer similar corrections in the present. Unlike our method, this approach *requires* a collaborative KG as input. In the absence of a rich and diverse edit history, the method cannot be applied. Moreover, this method considers triple removal as a possible repair, in contrast to our proposal. An approach that relies on KGE for cleaning heterogeneous dirty data has been proposed in [9]. The method detects errors based on the assumption that they occur due to inaccurate value assignments, and deals with the detection using a clustering model that classifies each triple as clean or erroneous. From the semantic reasoning perspective, KG repair approaches have been proposed in the area of Description Logic (DL) [11, 32]. Other methods include defining a set of appropriate actions for each inconsistency [12], reaching out to human annotators to correct erroneous information [1, 3, 6, 8], and using entity labels as textual clues for repairing violating triples [19].

**LM as a Source of Knowledge.** Recently, there has been an increasing interest in making use of LMs as sources of factual knowledge [16, 18, 27], by executing masked probes such as “Tim Roth was born in [MASK]”. Some works propose to unify KGE and LM methods for better knowledge representation [36]. Others proposed the addition of context through an information retrieval system [23, 26] and studied the position of tokens in the probes through shuffling and deleting certain types of words. Our work takes inspiration from these efforts to be used for repairing erroneous triples.

## 7 CONCLUSION

We presented a new method for judiciously expanding LM probes in order to repair incorrect triples in KGs. We showed that simple LM probes often provide low-accuracy results and, therefore, proposed different methods that utilize both salient information of the KG and the KG’s schema to improve the predictions. Experiments, with erroneous triples from popular large-scale DBpedia and Wikidata KGs, show that carefully selected context from the KG can significantly improve the probing results.

## REFERENCES

[1] A. Arioua and A. Bonifati. 2018. User-guided Repairing of Inconsistent KBs. In *EDBT*.

<sup>12</sup><https://www.mediawiki.org/wiki/ORES>

- [2] H. Arnaout and S. Elbassouni. 2018. Effective Searching of RDF Knowledge Graphs. *JWS* (2018).
- [3] A. Assadi, T. Milo, and S. Novgorodov. 2018. Cleaning Data with Constraints and Experts. In *WebDB*.
- [4] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives. 2007. DBpedia: A Nucleus for a Web of Open Data. In *ISWC*.
- [5] Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Janvin. 2003. A Neural Probabilistic Language Model. *J. Mach. Learn. Res.* (2003).
- [6] M. Bergman, T. Milo, S. Novgorodov, and W. Tan. 2015. QOCO: A Query Oriented Data Cleaning System with Oracles. In *PVLDB*.
- [7] M. Biennvenu. 2020. A Short Survey on Inconsistency Handling in Ontology-Mediated Query Answering. *KI* (2020).
- [8] M. Biennvenu, C. Bourgaux, and F. Goasdoué. 2016. Query-driven repairing of inconsistent DL-lite KBs. In *IJCAI*.
- [9] Ge C., Y. Gao, H. Weng, C. Zhang, X. Miao, and B. Zheng. 2020. KGClean: An Embedding Powered Knowledge Graph Cleaning Framework. *CoRR* (2020).
- [10] Boxi Cao, Hongyu Lin, Xianpei Han, Le Sun, Lingyong Yan, Meng Liao, Tong Xue, and Jin Xu. 2021. Knowledgeable or Educated Guess? Revisiting LMs as KBs. *ACL* (2021).
- [11] M. Chortis and G. Flouris. 2015. A Diagnosis and Repair Framework for DL-LiteA knowledge bases. In *ESWC*.
- [12] S. Flesca, S. Greco, and E. Zumpano. 2004. Active Integrity Constraints. In *PPDP*.
- [13] Stefan Heindorf, Martin Potthast, Gregor Engels, and Benno Stein. 2017. Overview of the Wikidata vandalism detection task at WSDM cup 2017. *WSDM* (2017).
- [14] S. Heindorf, M. Potthast, B. Stein, and G. Engels. 2016. Vandalism Detection in Wikidata. In *CIKM*.
- [15] Zhengbao Jiang, Antonios Anastasopoulos, Jun Araki, Haibo Ding, and Graham Neubig. 2020. X-FACTR: Multilingual Factual Knowledge Retrieval from Pretrained Language Models. In *EMNLP*.
- [16] Z. Jiang, F. Xu, J. Araki, and G. Neubig. 2020. How Can We Know What Language Models Know? *TACL* (2020).
- [17] J. Lajus, L. Galárraga, and F. Suchanek. 2020. Fast and Exact Rule Mining with AMIE 3. In *ESWC*.
- [18] N. Lee, B. Z. Li, S. Wang, W. Yih, H. Ma, and M. Khasba. 2020. Language Models as Fact Checkers?. In *ACL*.
- [19] P. Lertvittayakumjorn, N. Kertkeidkachorn, and R. Ichise. 2017. Correcting Range Violation Errors in DBpedia. In *ISWC*.
- [20] Q. Liu, G. Cheng, K. Gunaratna, and Y. Qu. 2021. Entity summarization: state of the art and future challenges. *JWS* (2021).
- [21] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach.
- [22] C. Meilicke, M. Fink, Y. Wang, D. Ruffinelli, R. Gemulla, and H. Stuckenschmidt. 2018. Fine-Grained Evaluation of Rule- and Embedding-Based Systems for Knowledge Graph Completion. In *ISWC*.
- [23] J. O’Connor and J. Andreas. 2021. What Context Features Can Transformer Language Models Use?. In *ACL*.
- [24] Heiko Paulheim and Aldo Gangemi. 2015. Serving DBpedia with DOLCE – More than Just Adding a Cherry on Top. In *ISWC*.
- [25] T. Pellissier Tanon and F. Suchanek. 2021. Neural KB Repairs. In *ESWC*.
- [26] F. Petroni, P. S. H. Lewis, A. Piktus, T. Rocktäschel, Y. Wu, A. H. Miller, and S. Riedel. 2020. How Context Affects LMs’ Factual Predictions. In *AKBC*.
- [27] F. Petroni, T. Rocktäschel, S. Riedel, P. Lewis, A. Bakhtin, Y. Wu, and A. Miller. 2019. Language Models as Knowledge Bases?. In *EMNLP-IJCNLP*.
- [28] Simon Razniewski, Andrew Yates, Nora Kassner, and Gerhard Weikum. 2021. Language Models As or For Knowledge Bases. *DL4KG* (2021).
- [29] A. Sarabadani, A. Halfaker, and D. Taraborelli. 2017. Building Automated Vandalism Detection Tools for Wikidata. In *WWW Companion*.
- [30] A. Singhal. 2012. Introducing the Knowledge Graph: things, not strings. <https://www.blog.google/products/search/introducing-knowledge-graph-things-not>.
- [31] F. Suchanek, G. Kasneci, and G. Weikum. 2007. YAGO: a Core of Semantic Knowledge. In *WWW*.
- [32] S. Tahrat, S. Benbernou, and M. Ouziri. 2021. Cleaning Inconsistent Data in Temporal DL-Lite Under Best Repair Semantics.
- [33] T. Tran, M. Gad-Elrab, D. Stepanova, E. Kharlamov, and J. Strötgen. 2020. Fast Computation of Explanations for Inconsistency in Large-Scale KGs. In *WWW*.
- [34] D. Vrandečić and M. Krötzsch. 2014. Wikidata: a Free Collaborative KB. *CACM* (2014).
- [35] Q. Wang, Z. Mao, B. Wang, and L. Guo. 2017. Knowledge Graph Embedding: a Survey of Approaches and Applications. *IEEE TKDE* (2017).
- [36] X. Wang, T. Gao, Z. Zhu, Z. Zhang, Z. Liu, J. Li, and J. Tang. 2021. KEPLER: A Unified Model for Knowledge Embedding and Pre-trained Language Representation. *TACL* (2021).
- [37] C. Xiaojun, J. Shengbin, and X. Yang. 2020. A review: knowledge reasoning over knowledge graph. *Expert Syst Appl* (2020).
- [38] I. Yamada, A. Asai, J. Sakuma, H. Shindo, H. Takeda, Y. Takefuji, and Y. Matsumoto. 2020. Wikipedia2Vec: An Efficient Toolkit for Learning and Visualizing the Embeddings of Words and Entities from Wikipedia. In *EMNLP*.