IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 12 August 2022, accepted 26 October 2022, date of publication 7 November 2022,
date of current version 6 December 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3220241

RESEARCH ARTICLE

Completing Scientific Facts in Knowledge Graphs of Research Concepts

AGUSTÍN BORREGO[1], DANILO DESSÌ[2], INMA HERNÁNDEZ[1], FRANCESCO OSBORNE[3,4], DIEGO REFORGIATO RECUPERO[2], DAVID RUIZ[1], DAVIDE BUSCALDI[5], AND ENRICO MOTTA[3]

[1]Department of Computer Languages and Systems, University of Seville, 41004 Seville, Spain
[2]Department of Mathematics and Computer Science, University of Cagliari, 09124 Cagliari, Italy
[3]Knowledge Media Institute, The Open University, MK7 6AA Milton Keynes, U.K.
[4]Department of Business and Law, University of Milano Bicocca, 20126 Milan, Italy
[5]LIPN, CNRS (UMR 7030), Sorbonne Paris North University, 93430 Villetaneuse, France

Corresponding author: Diego Reforgiato Recupero (diego.reforgiato@unica.it)

This work was supported in part by the Spanish Ministry of Science, Innovation and Universities, under Grant PID2019-105471RB-I00; and in part by the Office for Economy and Knowledge of the Andalusian Regional Government under Grant P18-RT-1060 and Grant US-1380565.

ABSTRACT In the last few years, we have witnessed the emergence of several knowledge graphs that explicitly describe research knowledge with the aim of enabling intelligent systems for supporting and accelerating the scientific process. These resources typically characterize a set of entities in this space (e.g., tasks, methods, evaluation techniques, proteins, chemicals), their relations, and the relevant actors (e.g., researchers, organizations) and documents (e.g., articles, books). However, they are usually very partial representations of the actual research knowledge and may miss several relevant facts. In this paper, we introduce SciCheck, a new triple classification approach for completing scientific statements in knowledge graphs. SciCheck was evaluated against other state-of-the-art approaches on seven benchmarks, yielding excellent results. Finally, we provide a real-world use case and applied SciCheck to the Artificial Intelligence Knowledge Graph (AI-KG), a large-scale automatically-generated open knowledge graph including 1.2M statements extracted from the 333K most cited articles in the field of Artificial Intelligence, and generated a new version of this knowledge graph with 300K additional triples.

INDEX TERMS Knowledge graphs, science of science, knowledge graph completion, triple classification, machine learning, semantic web.

I. INTRODUCTION

The rise of Open Science and the steady growth of the number of research publications, datasets, and other materials on the web is changing the way research outcomes are shared and explored, and is posing new challenges and opportunities. This large mass of open research outcomes has the potential of supporting a new generation of intelligent systems for actively supporting, automatizing, and accelerating the scientific effort [1].

One of the main challenges in this space is to generate a semantically rich, interlinked, and machine readable description of the available research knowledge. This could enable more sophisticated techniques to analyze the scientific literature. As a consequence, more advanced services could be provided, e.g., forecasting research dynamics, generating scientific hypothesis, identifying key insights, informing funding decision, confirming claims in news, or automatically running experiments [2], [3], [4].

The Semantic Web community has been working for several years on semantically rich representations of research outcomes by creating bibliographic repositories in the Linked Data Cloud [5], annotating existing knowledge bases [6], [7], generating knowledge bases of biological data [8], advocating the Semantic Publishing paradigm [9], formalising research workflows [10], [11], implementing systems for managing nano-publications [12], [13], micropublications [14] and developing ontologies to describe scholarly data, e.g., BIBO,[1] CSO [15], or SPAR [16].[2]

The associate editor coordinating the review of this manuscript and approving it for publication was Ananya Sen Gupta.

[1]BIBO - http://bibliontology.com
[2]SPAR - http://www.sparontologies.net/

In the last few years, we saw the emergence of several knowledge graphs (KGs) explicitly representing research knowledge. These KGs typically describe a set of entities in this space (for example, tasks, methods, evaluation techniques, datasets, proteins, chemicals), their relations, and the relevant actors (e.g., authors, organizations) and documents (articles, books…) [17], [18]. Some of these graphs are crowdsourced (e.g., ORKG [4], UMLS [19], Nanopublications [13]), while others are automatically generated from the text and metadata of research articles (e.g., AI-KG [2], CSO [20], TKG [21]).

As many other KGs, those that describe research concepts suffer from *incompleteness*. They are typically very partial representations of the actual research knowledge and may lack several relevant facts, that were not identified by information extraction approaches or human experts. The issue of incompleteness in knowledge graphs is usually addressed by link prediction or triple classification techniques [22], [23], which have proved to yield good results in several domains [17]. These methods typically use KG Embedding models (e.g. TransE [24], RotatE [25], ComplEx [26]), path-based features [27], [28], or Graph Neural Networks [29]. However, existing methods for knowledge graph completion under-perform on KGs of research concepts, as detailed in Section IV. In particular, they suffer from low precision, which is not acceptable in the scientific domain.

To address the above issue, in this paper, we introduce SciCheck, a new approach for completing scientific facts in knowledge graphs of research concepts. SciCheck is built on top of the CAFE approach [27] and introduces several new features and heuristics for the scholarly domain.

We evaluated SciCheck on two new benchmarks extracted from AI-KG (AIKG-1M and AIKG-500) and five well-known general benchmarks for triple classification (FB13, WN11, WN18, WN18RR, and NELL). The evaluation shows that SciCheck significantly outperforms nine alternative approaches in terms of precision, which we consider key for reliably extending knowledge graphs of research concepts, while still obtaining good values of recall. All the resources used for evaluation are available online.[3]

As use case, we used SciCheck to enrich the Artificial Intelligence Knowledge Graph (AI-KG)[4] [2], a large-scale automatically-generated open KG including 14M RDF triples and 1.2M reified statements extracted from the 333K most cited articles in the field of AI. We also made available to the scientific community a new version of AI-KG (version 1.2) with 300K additional triples[5] that we generated with SciCheck.

In summary, the main contributions of our work are the following:

- We propose SciCheck, a new triple classification technique that uses a variety of features to complete KGs of research concepts with a high precision.
- We compare SciCheck with nine alternative KG completion methods on AIKG-1M, AIKG-500, FB13, WN11, WN18, WN18RR, and NELL, showing that it obtains excellent results.
- We release two new datasets for KG completion: *AIKG-1M*, including 1M triples from AI-KG, and *AIKG-500*, including 500 manually annotated statements.
- We provide a real-world use case and apply SciCheck on AI-KG and use it to generate a new version of AI-KG containing 300K additional triples.

The remainder of this paper is organized as follows. Section II describes the related work. Section III describes SciCheck in detail, and Section IV discusses the evaluation results. Section V describes AI-KG and how SciCheck was applied to it in order to extend it. Finally, Section VI concludes the paper and presents future directions of research.

II. RELATED WORK

The majority of related proposals in this field are nowadays based on embedding models, i.e., producing a translation from the entities and relations in the graph into vectors that preserve their semantics. In this area, experts usually distinguish between knowledge graph embeddings, and language models.

KG embeddings [23], [30] learn embedded representations of KGs entities and relations, performing different transformations in an embedding space [24], [25], [26], [31], [32], [33], [34]. The resulting embedding space is subsequently used to evaluate the likelihood of a candidate triple to be correct or incorrect, since entities that are supposed to be related by means of a certain relation are expected to be closer to each other in the embedding space. They have also been recently used for assessing research hypotheses, yielding promising results [3].

While they provide good results in general, all of the former proposals suffer from a performance drawback: due to the way in which the embedded representations are obtained, they need to be recomputed whenever new triples are added to the KG, which is a relatively frequent event [35]. Language models are based on word embeddings (such as Word2Vec [36] or BERT [37]), that represent the semantic information encoded in the text of nodes and relations, and are therefore less affected by the introduction of new triples. These models are able to deal with text ambiguity and produce contextualized embeddings.

Embedding-based approaches are able to exploit features from both the entities and relations in the graph, but they usually explore the immediate neighborhood of entities, disregarding longer paths in the graph that could also provide some interesting features. Therefore, other approaches are proposed to leverage these longer paths: *path-based*, and *graph neural network-based* approaches.

Path-based techniques exploit the highly relational nature of KGs to learn how to predict new relations between entities. Regarding this approach, Lao and Cohen [38] introduced the Path Ranking Algorithm (PRA), a two-step

[3]Evaluation data - https://zenodo.org/record/5764114
[4]AI-KG - http://w3id.org/aikg
[5]AI-KG 1.2 - https://zenodo.org/record/7276434

process to find which paths may be useful to predict a certain relation. An evolution of PRA named Subgraph Feature Extraction (SFE) by Gardner and Mitchell [39] achieves better performance than PRA and produces more expressive results. Mazumder et al. [40] propose a random walk-based approach using neighborhood-guided path finding, where semantic similarities between entities are computed by applying a Word2Vec-based embedding model on the names of the entities. Reinforcement learning has also been used to find valuable paths that can help to successfully complete a KG [41]. Shen et al. [28] propose combining the benefits of embeddings and path-based approaches, by computing embeddings of the entities and relations, and then combining these embeddings in the forms of paths. Unfortunately, due to the non-deterministic way in which these paths are computed, they may miss relevant information by mere chance. More recently, Borrego et al. [27] proposed CAFE, a deterministic approach to exploit the highly connected nature of KGs that does not rely on random paths.

There are also a number of proposals that leverage the use of Graph Neural Networks (GNNs) to exploit not just a limited set of paths, but the entire structure of the graph. Some of them are based on traditional embedding models [42], [43]. The most recent proposals are based on Graph Attention Networks [44], [45], [46]. An extended survey on GNNs and their applications has been carried out by Zhou et al. [29]. The main drawback of this approach is the amount of computational resources it requires, making them unappealing to deal with real-world KGs, such as those about research concepts, which are our focus.

The particularities of research concepts make the former proposals generally unable to complete these KGs with a high precision. They usually contain a large number of ambiguous and synonym terms, due to a lack of standardization in the vocabulary used in different research works [47]. Also, they often contain highly categorical relations [48], i.e., relations in which the number of possible head entities is significantly higher than the number of possible tail entities. Therefore, some language models have been proposed based on different types of KG embeddings to deal specifically with this type of graphs [48], [49], [50]. Some recent techniques, such as exBERT [47] exploit contextualized language models rather than KG embeddings.

The novelty of our approach resides in not solely using KG embeddings, language models, or random paths, but on a combination of features that leverages the strengths of embeddings and deterministic path features, and does not suffer from the high hardware requirements of GNNs.

Specifically, SciCheck makes use of deterministic path-based and embedding-based features to solve the problem of triple classification in general-domain knowledge graphs, and more specifically, in scholarly KGs. In addition, according to our experimental results, SciCheck is also able to outperform the other proposals in terms of precision, which is essential to complete KGs of research concepts, while still achieving a fair recall.

III. SciCheck

SciCheck[6] is a novel approach for triple classification designed to complete scientific statements in a knowledge graph. It is built on top of the CAFE approach [27] by incorporting a new set of features and heuristics tailored to capture scientific knowledge. SciCheck takes an entire KG in the form of triples as input, and produces one neural-based classifier for each relation in the KG as output. Specifically, given a relationship r, SciCheck generates a model $f_r : (h, r, t) \rightarrow s$, that assigns a confidence score s in the range $[0, 1]$ to any arbitrary triple $< h, r, t >$ to solve a binary classification task (''is the triple correct or not?''). To feed the model, triples are converted into a numerical vector representation using ad-hoc features and contextual embedding representations. SciCheck can operate on any KG and focuses on optimizing precision, to ensure that the knowledge deemed correct is trustworthy.

In the following subsections, we describe all the relevant steps for the workflow of SciCheck. For the sake of illustration, we provide a visual summary of this workflow in Fig. 1. Additionally, Fig. 2 displays a small KG that will be used to provide specific examples for some steps.

A. LOADING THE KG

The first step of SciCheck takes as an input a set of triples from the target KG. Triples are transformed into a graph structure. Due to the generally large number of entities that comprise a KG and the high volume of read operations that are used in the following steps, the KG is stored in the form of adjacency hashmaps, which also preserve the types of the different relations.

B. GENERATING NEGATIVE EXAMPLES

Knowledge Graphs only contain positive knowledge, i.e., triples for which their heads and tails are known to be related by means of a relationship. However, in order to train a classifier, negative triples are also needed. To do this, SciCheck follows the same approach as many other related techniques [27], [28], [38], [51], [52], [53] and generates negative triples by corrupting a positive triple $< h, r, t >$ and replacing t with t', in such a way that $< h, r, t' >$ is not part of the original graph.

In order to produce more realistic negative triples, we randomly pick t' such that its type is in the range of the relation r [54]. This can either be done automatically by using entities which appear as tail of that relation in the set of positive triples, or by using ontological information if it is available.

C. CONVERTING TRIPLES INTO FEATURE VECTORS

After both positive and negative examples are included in the graph, all triples are converted into labeled feature vectors that are provided to the neural classifier for both training and testing. For this purpose, SciCheck uses an extensible set of neighbourhood-aware features specifically tailored to scholarly information, which represent the neighbourhoods

[6]https://github.com/agu-borrego/SciCheck

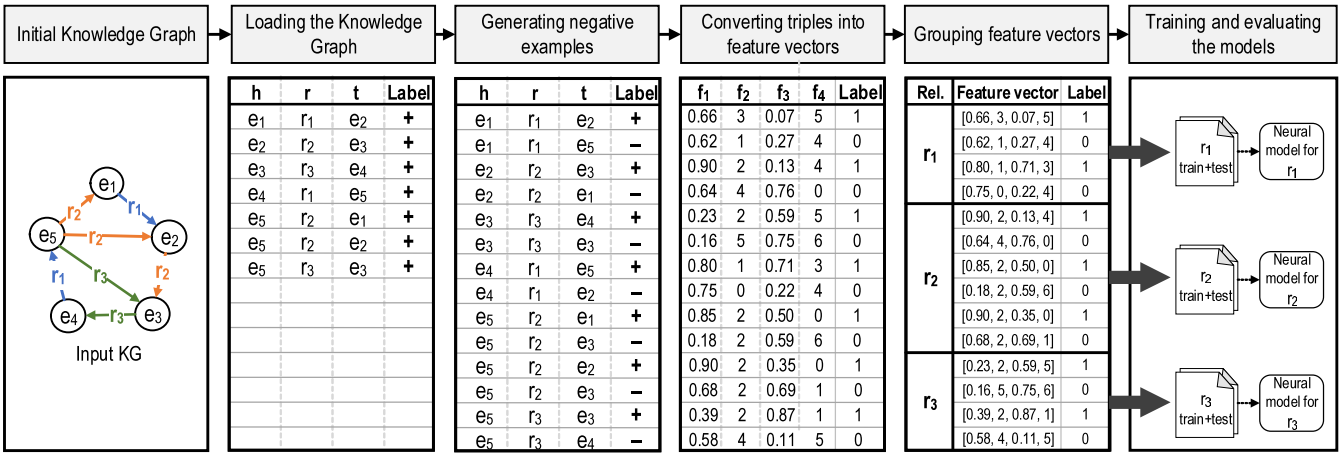


FIGURE 1. Workflow of SciCheck.

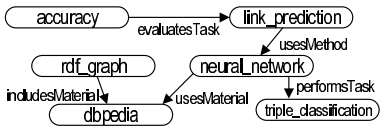


FIGURE 2. A small KG containing research concepts relevant to the Semantic Web domain.

of the two entities of a triple in a variety of ways. The neighbourhood of an entity is considered to be the set of all other entities that can be reached from it using an oriented path (i.e., the direction of links matters) in a certain number of hops. This number of hops is called the neighbourhood's "radius". Fig. 2 shows a KG that will be used as an example in the discussion of the features.

Each triple is evaluated by all features. The values associated to the triple for each feature form the triple feature vector.

Each feature can also depend on a number of parameters, such as a maximum neighbourhood radius. These features, and their rationales, are as follows:

- f_1*: Number of entities in the neighbourhood of radius r of the head and the tail of a triple.* For example, in Fig. 2, three entities can be reached in total using up to two hops from `link_prediction`, namely, `neural_network`, `dbpedia`, and `triple_classification` Note that the entity 'accuracy' is not reachable because the graph is oriented.
- f_2*: Index of N-path centrality [55] of the head and tail of a triple.* This feature assesses how well-connected an entity is to the rest of the graph in relative terms. It is defined as follows: for every vertex v of a graph $G = (V, E)$, the n-path centrality $C_k(v)$ is defined as the sum, over all possible source nodes s, of the probability that a message originating from s goes through v, assuming that the message traversals are only going along random simple paths of at most k edges. For example, in the KG shown in Fig. 2, the entity `dbpedia` has a higher N-path centrality than `accuracy`, since a random path from any entity in the graph is more likely to go through the former than the latter, considering the directionality of the graph.
- f_3*: Cardinality of the intersection of the neighbourhoods of radius r of the head and tail of the triple.* This feature measures the raw amount of common entities in the vicinities of the two entities in a triple. For example, using a radius $r = 1$ in the example shown in Fig. 2, the entities `rdf_graph` and `neural_network` have the common entity `dbpedia` in their neighbourhoods
- f_4*: Jaccard index of the neighbourhoods of radius r of the head and tail of the triple.* This feature provides a similar assessment as the previous one, but normalized in the interval [0, 1] The Jaccard index is defined as:

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|}$$

- f_5*: Adamic/Adar index [56] between the head and tail.* This index gives higher scores to entities whose neighbourhoods are smaller. It complements the previous two features, since a higher number of shared nearby entities is likely to be less significant if head and tail have a very large amount of connections. It is defined as the sum of the inverse logarithmic degree centrality of the neighbors shared by the two nodes:

$$A(x, y) = \sum_{u \in N(x) \cap N(y)} \frac{1}{log|N(u)|}$$

where $N(u)$ is the set of nodes adjacent to u

- f_6*: Paths of length r between the head and tail.* For example, in Fig. 2, the entities `link_prediction` and `dbpedia` are connected by a path of length 2, by means of the triples `< link_prediction, usesMethod, neural_network>` and `< neural_network, usesMaterial, dbpedia>`. Additionally, the relations that are present in those paths are also encoded using a r-hot vector.

- *f_7: Cosine similarity of the word embeddings of the head and tail.* This feature measures the semantic similarity of the two entities in a triple, using any entity embeddings If we consider A and B to be the embeddings of the head and tail entities of the triple respectively, it is defined as:

$$\cos(\mathbf{A}, \mathbf{B}) = \frac{\sum_{i=1}^{n} \mathbf{A}_i \mathbf{B}_i}{\sqrt{\sum_{i=1}^{n} (\mathbf{A}_i)^2} \sqrt{\sum_{i=1}^{n} (\mathbf{B}_i)^2}}$$

- *f_8: Dot product of the word embeddings of the head and tail entities.* This feature complements the previous one by also taking into account the magnitudes of the embeddings of the entities If we consider A and B to be the embeddings of the head and tail entities of the triple respectively, it is defined as:

$$\mathbf{A} \cdot \mathbf{B} = \sum_{i=1}^{n} \mathbf{A}_i B_i$$

- *f_9: Types of the head and tail entities according to the ontology of the KG.* This feature encodes the known types of the entities according to the available ontology as two one-hot vectors. In Fig. 2, the entity `dbpedia` has type `Resource`, while `accuracy` is a `Metric`

Regarding the rationales of the features, f_1 and f_2 leverage the fact that large neighbourhoods are more prone to contain unrelated information, while smaller ones are usually more specific. This is especially true in the scholarly domain, since, as an example, the entity `neural_network` may be mentioned in a large amount of papers and proposals that are not directly related to each other.

The features that measure the similarities of two neighborhoods (f_3, f_4, and f_5) follow the intuition that correct triples have a higher amount of shared entities in their respective neighbourhoods than incorrect ones, as shown by previous research efforts [18], [27], [57].

Feature f_6 measures the number of paths between two entities because a correct triple will typically have a larger number of unique paths of a given maximum length between head and tail than an incorrect one. Furthermore, the information about which relations are comprised by those paths can be useful since the semantic meaning of a path changes depending on the relevant relations.

Features f_7 and f_8 incorporate information from the word embeddings of the two entities, which had been shown to be advantageous for triple classification [25], [31]. SciCheck uses by default the RoBERTa model [37] to generate the word embeddings, since is able to capture and represent semantic similarities across a wide range of domains.

Finally, feature f_9 leverages the ontological schema of the KG. This allows SciCheck to include information regarding the types of the two entities in a triple into the feature vector for that triple. Furthermore, SciCheck can automatically classify a triple as incorrect if the triple does not respect the domains and ranges of the relation as defined in the ontological schema. For example, in the KG shown in Fig. 2, the triple `< accuracy, evaluatesTask, rdf_graph>` would be considered incorrect without further evaluation, because the range of the relation `evaluatesTask` is `Task`, while `rdf_graph` is a `Material`.

SciCheck makes use of a much more comprehensive set of features than the original CAFE, which in turn allows a better characterization of entities and predicates. In particular, the features based on word embeddings enable SciCheck to exploit the implicit contextual information from the training papers that may not be encoded in the KG. Additionally, the inclusion of ontology-based features allows SciCheck to take advantage of the available high-level knowledge about any specific domain. These improvements are particularly crucial for assessing scientific claims, which tend to use a specific jargon and to rely on a well defined epistemological framework.

Furthermore, different types of relations in the graph may carry specific insight that should be captured separately. For this reason, SciCheck first computes all features in the input KG as-is, and then it computes them again in different versions of the KG where only relations of a single type are present. This is done for all the different relations in the KG. Additionally, in features that use the neighbourhoods of the head and tail entities such as f_1 or f_3, these two neighborhoods are calculated using all possible combinations of relations. Finally, SciCheck concatenates all the resulting features in the final feature vector.

The features which involve computing entity neighbourhoods or paths (from f_1 to f_6) use a maximum number of hops for their computations. Following the findings in [27], by default SciCheck computes them for a maximum number of hops num_{hops} of 1, 2, and 3. The resulting set of features using different radii are eventually all added to the final feature vector. Considering all the possible combinations with the number of different relations in the graph, which also affects the size of the feature vector as described previously, the number of total features is $num_{hops} \times 6 \times \#rels^2 + 3 \times \#rels$, where $\#rels$ is the number of distinct relations in the KG.

D. GROUPING FEATURE VECTORS

SciCheck creates one classifier per each relation, under the assumption that the specific information needed to correctly classify triples may vary depending on the specific relation. After all triples have been converted into feature vectors in the previous step, they are grouped by the relation present in the triple, and passed on to the relevant classifier.

E. TRAINING AND EVALUATING THE MODELS

SciCheck trains a neural network-based classifier model for each relation using the resulting feature vectors. We generate

multiple models, so that each classifier has a high specialization in addressing the target relation.

It is also advantageous to consider different neighbourhood radii that might carry information of different nature. For this reason, each of these classifiers is composed of several sub-models that consider only the features computed using a specific radius value on the sub-graph of a specific relation as in [27]. They are combined into a single classifier model by using an additional layer with a single neuron, which receives the outputs of all sub-models and combines them into a single output.

This step involves the use of a flexible neural classifier, which can be fine-tuned for the KG in question. The hyperparameters used in the evaluation are discussed in Section IV-A.

IV. EVALUATION

This section reports and discusses the evaluation of SciCheck. It also describes the evaluation data, including the new benchmarks that we created from the AI-KG Knowledge Graph (AIKG-1M and AIKG-500 are discussed in Section IV-A, and they are available at https://zenodo.org/record/5764114).

A. EVALUATION PROTOCOL

We evaluated the performance of SciCheck on seven benchmarks against nine alternative approaches. Five of the baselines are well-known embedding-based KG completion approaches: *TransE*, *TransD*, *TransH*, *SimplE*, and *ComplEx* [24], [26], [31], [53], [58]. To provide a common ground to train and test these techniques, we used the OpenKE [59] tool.

In order to assess the contributions of the different components of SciCheck, we also considered five alternative versions of our approach:

- *CAFE Baseline*, which uses solely the context-aware features for KG completion such as neighbourhood size, shared entities, connectivity, and so on from the original implementation [27].
- *CAFE + RoBERTa*, which extends CAFE by considering features based on the similarity of the embeddings of head and tail, using the RoBERTa model.
- *CAFE + SciBERT*, which extends CAFE by considering features based on the similarity of the embeddings of head and tail, using SciBERT, an alternative BERT-based text embedding model[7] specifically tailored to scientific documents.
- *CAFE + Ontology*, which extends CAFE by considering features that identify the types of head and tail according to the domain ontology (e.g., AI-KG ontology) and also filters triples whose entities are not consistent with the domain and range restrictions of the relation.
- *SciCheck*, the full version of our approach, which incorporates both features based on word embeddings (RoBERTa in the current implementation) and features based on the ontology, as described in Section III.

These methods were evaluated on the following benchmarks, whose characteristics are summarized in Table 1:

- *AIKG-1M*, a new dataset that we created from AI-KG. We used a de-reified version of AI-KG, in order to consider only triples which involve tasks, methods, materials, metrics, and other scientific entities. As a result, 1,075,652 triples were directly generated from scientific literature, without considering facts that were materialized using the domain semantics defined in the AI-KG ontology (e.g. transitivity). Triples were split into a training and a testing set with a split ratio of 80%-20%, respectively. To generate negative triples in the testing split, each positive triple was corrupted once by randomly replacing the tail entity with another one within the domain of the relation in the triple, i.e., if the range of the tail entity is a Task, then it is substituted by another entity whose type is Task. We also make sure that the randomly generated negative triple is not already present in the KG, to prevent creating false negatives whenever possible. As an example, the triple `< dbpedia, usesOtherEntity, sparql_query>` is correct, while the corrupted version `< dbpedia, usesOtherEntity, cost_function>` is considered incorrect, where `sparql_query` and `cost_function` are both of type `OtherEntity`. However, negative examples were not generated for the training split, as specific KG completion techniques usually have a preferred way to generate them automatically [60]. In total, the training split comprised 860,512 positive triples and the testing split includes 430,280 triples (50% positive and 50% negative).
- *AIKG-500*, a new dataset that we constructed by manually annotating triples in AI-KG about the Semantic Web. To construct it, we randomly selected 250 triples which had as their head one of the 24 sub-topics of the Semantic Web according to the CSO ontology [61] and were considered to be correct by at least 2 methods among *TransE*, *TransD*, *TransH*, *SimplE*, *ComplEx*, *and SciCheck*. Another 250 triples were randomly selected out of those deemed incorrect by at least 2 techniques. The resulting 500 triples were manually annotated by five domain experts, with an inter-reviewer agreement of 0.61 (according to Cohen's kappa), which is typically considered a substantial agreement. A majority vote approach was used to determine that 221 triples were correct and 279 were incorrect. Since this dataset was created for the purpose of providing a small but high-quality and manually-annotated testing split, in this evaluation we used AIKG-1M for the training split.

[7] https://huggingface.co/sentence-transformers/paraphrase-distilroberta-base-v2

TABLE 1. Overview of the benchmarks used for evaluation.

KG	Training triples	Test triples	Entities	Relations
AIKG-1M	860,512	430,280	820,708	20
AIKG-500	860,512	500	228	7
FB13	228,172	105,509	74,998	13
WN11	77,948	36,042	38,195	9
WN18	71,984	33,282	40,943	11
WN18RR	86,835	3,134	40,943	11
NELL	86,971	40,104	53,934	148

- *FB13* [34], a subset of FreeBase [51] that focuses on relevant people and their family relations, locations, professions, and other personal data.
- *WN11* [34], a subset of WordNet centered around different semantic relations between over 38K words.
- *WN18* [62], which expands WN11 with additional relations.
- *WN18RR* [63], which improves WN18 by removing reciprocal relations in the test set. This makes triple classification more challenging, since otherwise the model can predict that a triple `< a, hasChildren, b>` is true whenever the triple `< b, hasParent, a>` appears in the training set.
- *NELL* [39], a subset of the NELL KG [35] with information and relations about many different domains, e.g., actors which starred in movies, writers and their works, or athletes and their teams.

It is well-known [63] that these traditional benchmarks suffer from information leakage between the training and test sets, due to the presence of reciprocal relations. For this reason, we removed all reciprocal relations in all datasets except WN18, since we also include its previously discussed sanitized version, WN18RR.

To predict the correctness of a triple, we used feed-forward neural networks with 3 intermediate layers containing 128, 64 and 32 neurons, respectively. The output neuron uses a sigmoid function, returning a confidence score in the interval [0, 1]. The classifier was trained throughout 100 epochs, using a binary cross-entropy loss function.

Since SciCheck is a triple classifier, we evaluated its effectiveness by comparing the labels it predicted for the triples in the testing set against the ground truth. The results are thus reported in terms of precision and recall, which have been recently become standard metrics to evaluate KG completion, since they can be more informative than MRR and Hits@N in many practical settings [64], [65]. In this paper, we specifically focus on precision, since we have the concrete objective of extending AI-KG and this can only be reliably done using a method with a high precision.

B. RESULTS AND DISCUSSION

Table 2 and Table 3 report the precision and recall of the KG completion techniques on AIKG-1M. To determine whether a triple was correct or incorrect, we used a confidence threshold of 0.5 for SciCheck, as suggested in [27]. The thresholds of the other state-of-the-art techniques under evaluation and their results were obtained using the OpenKE [59] tool, allowing it to choose the optimal value for each one.

All CAFE variants outperform embedding-based techniques in precision, achieving notably higher values. Including features from the text embeddings provides also an important improvement over the base version of CAFE. Both SciCheck and the variants that improve the baseline using embedding-based features rank consistently among those with the highest precision for all relations, with the differences between them being very narrow.

The best performing method in terms of precision is the final version of SciCheck (0.74), followed by RoBERTa (0.73), which can obtain better precision for some less common relationships. Interestingly, using text embeddings trained specifically on academic abstracts (SciBERT) yields a slightly worse performance than using the generic RoBERTa model. This may suggest that more general embeddings may sometimes produce better performance on KGs of research concepts, but this needs to be investigated further.

The *Ontology* variation, which includes one-hot type vectors and domain/range checking for the relation, only slightly improves the baseline. This is most likely due to the type-constrained way in which the negative triples were generated, since it already guarantees that the domain and range types of the relation are preserved.

The recall of SciCheck is naturally lower than that of the embedding-based approaches, in a typical precision-recall trade-off. However, this is acceptable since the main goal is to expand scientific KG with correct triples, hence, a high precision is necessary. SciCheck has also a generally higher recall than all other CAFE variants. Consequently, the results suggest that SciCheck is the best performing technique for the task of reliably completing scientific KGs.

It is noteworthy that different relations can lead to very different performance. For instance, relations such as `narrower`, `supportsTask` and `supportsMethod` yield very good performance. Conversely, the methods did not perform as well on relations such as `evaluatesTask` and `evaluatesOtherEntity`. This may depend on the number of relevant examples or the fact that some relations are inherently harder to predict. The role of different relations in the context of completing scientific KG requires further analysis.

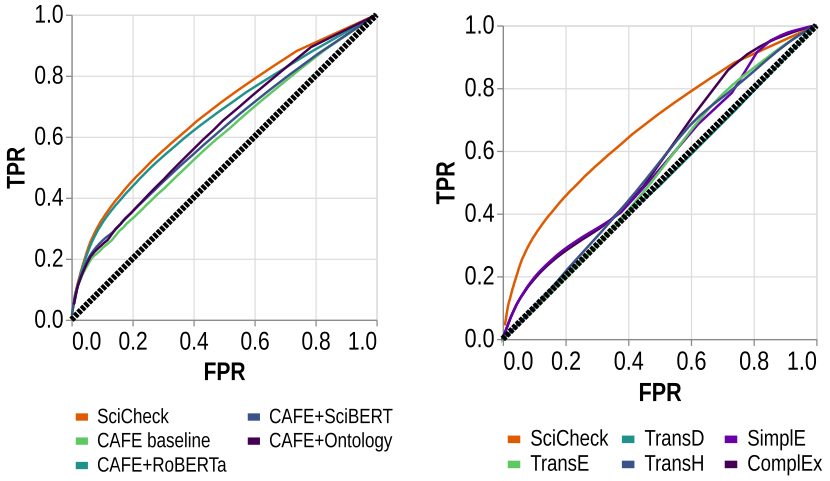
In order to study the performance of the different techniques for all possible threshold values, we also report their corresponding ROC curves in Fig. 3. This analysis confirms the previous findings: 1) SciCheck outperforms all the other methods, 2) text embedding features significantly improve the baseline, and 3) the ontological features slightly improve the baseline. In addition, Fig. 3(b) confirm that SciCheck outperforms the standard state-of-the art methods regardless of the threshold.

TABLE 2. Precision values for AIKG-1M. Highest precision for each relation is marked in **bold**.

Relation	# triples	SciCheck	Baseline	RoBERTa	SciBERT	Ontology	TransE	TransD	TransH	SimplE	ComplEx
usesMethod	460,723	**0.71**	0.67	0.70	0.70	0.70	0.38	0.49	0.36	0.56	0.56
usesOtherEntity	136,310	**0.72**	0.70	**0.72**	0.71	0.70	0.41	0.50	0.40	0.57	0.56
includesOtherEntity	113,678	**0.79**	0.77	**0.79**	0.78	0.76	0.43	0.49	0.41	0.55	0.55
narrower	107,811	**0.84**	0.80	0.83	0.81	0.76	0.49	0.49	0.48	0.57	0.56
usesMaterial	41,075	0.67	**0.68**	**0.68**	0.67	**0.68**	0.49	0.50	0.36	0.57	0.56
includesMethod	30,332	0.82	0.79	**0.83**	0.78	0.77	0.44	0.50	0.43	0.56	0.56
usesTask	22,341	**0.76**	0.73	0.75	0.68	0.73	0.49	0.49	0.48	0.55	0.54
evaluatesMethod	17,954	0.57	0.56	0.57	0.56	0.56	0.51	0.53	0.49	**0.60**	**0.60**
includesMaterial	10,190	0.70	0.67	**0.71**	0.67	0.65	0.49	0.49	0.40	0.57	0.56
usesMetric	7,749	0.67	0.64	**0.68**	0.60	0.66	0.41	0.50	0.37	0.56	0.56
includesTask	4,375	0.78	0.69	0.73	0.70	**0.81**	0.44	0.49	0.48	0.55	0.54
supportsTask	3,622	**0.87**	0.86	**0.87**	**0.87**	0.86	0.39	0.50	0.37	0.68	0.68
evaluatesOtherEntity	2,994	0.56	0.54	0.55	0.54	0.54	0.49	0.52	0.46	**0.60**	0.59
evaluatesTask	2,275	0.56	0.57	0.58	0.58	0.56	0.50	0.52	0.47	**0.59**	**0.59**
improvesMetric	1,860	0.84	0.81	0.84	0.81	**0.85**	0.54	0.52	0.53	0.67	0.67
supportsMethod	1,850	0.85	**0.86**	0.85	0.82	0.83	0.52	0.51	0.47	0.66	0.65
supportsOtherEntity	1,691	0.85	0.82	**0.87**	0.85	0.83	0.37	0.51	0.30	0.64	0.62
predictsOtherEntity	913	**0.84**	0.81	**0.84**	**0.84**	0.83	0.43	0.48	0.36	0.65	0.64
improvesMethod	814	0.67	0.68	**0.69**	0.68	**0.69**	0.39	0.52	0.44	0.67	0.66
improvesTask	639	**0.75**	0.72	**0.75**	0.73	0.71	0.43	0.48	0.42	0.66	0.66
Micro-average		**0.74**	0.70	0.73	0.72	0.71	0.42	0.49	0.39	0.56	0.56

TABLE 3. Recall values for AIKG-1M. Highest recall for each relation is marked in **bold**.

Relation	# triples	SciCheck	Baseline	RoBERTa	SciBERT	Ontology	TransE	TransD	TransH	SimplE	ComplEx
usesMethod	460,723	0.27	0.22	0.28	0.21	0.19	0.20	0.69	0.16	0.71	**0.74**
usesOtherEntity	136,310	0.26	0.19	0.26	0.21	0.19	0.25	0.56	0.20	0.69	**0.73**
includesOtherEntity	113,678	0.26	0.16	0.25	0.20	0.16	0.27	0.67	0.22	0.75	**0.77**
narrower	107,811	0.39	0.17	0.32	0.20	0.28	0.67	0.51	0.59	0.70	**0.73**
usesMaterial	41,075	0.21	0.14	0.20	0.14	0.13	0.61	0.53	0.16	0.70	**0.72**
includesMethod	30,332	0.29	0.17	0.28	0.21	0.17	0.30	**0.75**	0.25	0.71	0.73
usesTask	22,341	0.28	0.15	0.28	0.20	0.15	0.65	0.65	0.61	0.77	**0.81**
evaluatesMethod	17,954	0.30	0.28	0.30	0.30	0.28	0.48	**0.76**	0.40	0.61	0.62
includesMaterial	10,190	0.25	0.16	0.24	0.16	0.17	0.60	0.67	0.20	0.73	**0.75**
usesMetric	7,749	0.18	0.10	0.17	0.12	0.10	0.25	0.63	0.18	**0.73**	**0.73**
includesTask	4,375	0.31	0.22	0.35	0.26	0.18	0.30	0.65	0.68	0.77	**0.80**
supportsTask	3,622	0.58	0.55	0.57	0.55	0.58	0.19	**1.00**	0.17	0.43	0.44
evaluatesOtherEntity	2,994	0.32	0.35	0.32	0.35	0.31	0.41	**0.77**	0.33	0.62	0.63
evaluatesTask	2,275	0.37	0.29	0.32	0.28	0.30	0.49	**0.86**	0.38	0.64	0.64
improvesMetric	1,860	0.41	0.41	0.42	0.40	0.37	0.67	**0.86**	0.49	0.44	0.45
supportsMethod	1,850	0.43	0.42	0.43	0.46	0.43	0.67	**0.77**	0.31	0.47	0.50
supportsOtherEntity	1,691	0.42	0.39	0.40	0.38	0.39	0.16	**0.63**	0.10	0.52	0.56
predictsOtherEntity	913	0.59	0.51	0.60	0.59	0.55	0.19	**0.66**	0.16	0.50	0.52
improvesMethod	814	0.33	0.29	0.34	0.32	0.33	0.19	**0.56**	0.34	0.44	0.48
improvesTask	639	0.70	0.65	0.70	0.69	0.65	0.29	**0.82**	0.25	0.49	0.48
Micro-average		0.28	0.20	0.28	0.21	0.20	0.31	0.65	0.24	0.71	**0.74**



(a) SciCheck variants

(b) SciCheck and embedding techniques

FIGURE 3. ROC curves of the different methods on AIKG-1M.

To check whether the differences between the methods were statistically significant, we used DeLong's test [66] to compare the areas under two curves. The p-values obtained when comparing the ROC curve of SciCheck with the alternative methods in Fig. 3(a) and Fig. 3(b) were all < 0.0001. This very high statistical confidence is due to the large number of observations, since the testing set of AIKG-1M includes more than 400,000 triples.

Table 4 shows the performance of the methods on AIKG-500, which are consistent with the previous findings. For the sake of brevity, here we do not report the results of all CAFE variants, which are in line with those obtained on AIKG-1M. Even in a smaller, manually annotated benchmark, SciCheck achieves a high precision, which confirms that it is suitable for completing scientific KGs.

Table 5 reports the performance of all the techniques on five standard benchmarks for triple classification. The results show that SciCheck is able to outperform other techniques in almost all cases, thus being an effective triple classification tool for KGs of many different natures. They also confirm that completing scientific KGs is indeed a challenging task that requires specialized techniques, as the general purpose embedding-based approaches yield worse results on benchmarks extracted from AI-KG in comparison to generic ones.

In order to assess the scalability of our solution, Table 6 reports the seconds used by SciCheck to process the previously discussed datasets. To ensure statistical significance, we measured the runtime for each benchmark 10 times, and we report the average and the standard deviation for each one.

Table 6 shows that the runtime ranges from a few seconds to over two hours according to the dataset. These differences are caused by mainly two factors. First, the amount of distinct entities corresponds directly to the number of RoBERTa embeddings that have to be computed, which are typically quite time-consuming. Hence, a larger number of entities has a negative impact on runtime. Second, and most importantly, the specific topology of every KG affects the size of the neighborhoods of the entities, and thus also affects the time it takes to compute features on them. The case of FB13 is particularly noteworthy since, in contrast with the other datasets, it contains many entities with a very high cardinality. This causes the sizes of the entity neighborhoods to grow exponentially in size, resulting in longer runtimes.

Finally, in order to establish a fair comparison with the existing embedding-based KG completion approaches, Table 7 reports their runtime in seconds compared to that of SciCheck for the AIKG-1M dataset. Embedding-based KG completion approaches were run using 1, 000 iterations, as it is commonly done by related studies [24], [26], [31], [53]. SciCheck took considerably less time to run on the large AIKG-1M dataset than its state-of-the-art counterparts. This suggests that SciCheck is more scalable and can realistically be used on large-scale scientific KGs.

V. USE CASE: AI-KG

A real-world use case for SciCheck involves the development and extension of AI-KG [2], a large scale knowledge graph about research entities from the AI domain. AI-KG was released in late 2020 and it includes about 14M RDF triples and 1.2M reified statements about 800K entities extracted from 333K articles in the field of AI. It describes 5 types of entities (tasks, methods, materials, metrics, others) linked by 27 relations (e.g., usesMaterial, evaluatesMethod, supportsTask). AI-KG statements characterize the relationships between two entities according to their description in a set of scientific articles, e.g., `< sentiment_analysis, usesMaterial, twitter_data>`.

TABLE 4. Precision and recall values for AIKG-500. The highest value for each metric is marked in **bold**.

		Precision						Recall					
Relation	# triples	SciCheck	TransE	TransD	TransH	SimplE	ComplEx	SciCheck	TransE	TransD	TransH	SimplE	ComplEx
includesMaterial	115	**0.69**	0.40	0.62	0.38	0.47	0.45	0.48	**1.00**	0.57	0.59	0.89	0.86
includesOtherEntity	93	**0.57**	0.50	0.55	0.38	0.49	0.45	**0.76**	0.24	0.74	0.42	0.74	0.61
usesMethod	88	**0.66**	0.39	0.46	0.42	0.53	0.59	**0.72**	0.44	0.53	0.53	0.50	0.61
usesMaterial	62	**0.86**	0.67	0.67	0.65	0.76	0.74	0.45	**0.95**	0.24	0.74	0.90	0.93
includesMethod	53	**0.61**	0.31	0.39	0.30	0.27	0.53	0.56	**0.72**	0.50	0.44	0.17	0.44
usesTask	46	**0.70**	0.48	0.55	0.48	0.57	0.51	0.73	**1.00**	0.77	**1.00**	0.95	0.86
usesOtherEntity	43	**0.78**	0.33	0.00	0.38	0.63	0.53	0.37	0.21	0.00	0.47	**0.53**	0.47
Micro-average		**0.68**	0.44	0.50	0.42	0.52	0.53	0.59	0.66	0.51	0.58	0.69	**0.70**

TABLE 5. Micro-average precision and recall on four general benchmarks. The highest value for each metric is marked in **bold**.

	Precision						Recall					
KG	SciCheck	TransE	TransD	TransH	SimplE	ComplEx	SciCheck	TransE	TransD	TransH	SimplE	ComplEx
FB13	**0.87**	0.60	0.64	0.67	0.31	0.41	**0.76**	0.66	0.67	0.67	0.10	0.16
WN11	**0.89**	0.45	0.48	0.47	0.57	0.44	**0.74**	0.53	0.58	0.60	0.33	0.22
WN18	**0.65**	0.40	0.39	0.19	0.51	0.49	0.81	0.56	0.60	0.12	**0.88**	0.87
WN18RR	**0.80**	0.31	0.36	0.35	0.62	0.73	**0.73**	0.51	0.53	0.44	0.36	0.40
NELL	0.66	0.47	0.50	0.45	0.68	**0.77**	**0.86**	0.53	0.62	0.62	0.31	0.34

TABLE 6. SciCheck runtimes in seconds for all datasets under evaluation (avg ± std, 10 runs).

KG	Runtime
AIKG-1M	2,758.79 ± 37.27
AIKG-500	1,794.94 ± 12.58
FB13	9,400.10 ± 63.04
WN11	34.30 ± 0.28
WN18	55.59 ± 0.45
WN18RR	26.00 ± 0.14
NELL	4.33 ± 0.09

TABLE 7. Runtime in seconds for SciCheck and embedding-based KG completion approaches on the AIKG-1M dataset.

Technique	Runtime
SciCheck	2,758.79
TransE	7,147.52
TransD	13,871.79
TransH	10,134.41
SimplE	6,592.20
ComplEx	11,767.73

It is important to note that in AI-KG a triple associated with a set of papers is considered true if the papers actually contain that claim. To analyze the general truth value of each claim is not currently possible. Therefore, triples in AI-KG are devised to be a means for representing specific claims by researchers.

For example, the entity *sentiment_analysis* only represents the concept or idea of sentiment analysis as it is described in the original corpus of papers, but it is not aimed to represent or include all available prototypes and implementations to predict sentiments and emotions available today. In fact, such a modeling would require to promote research entities from concepts to classes to describe specific ontological knowledge (e.g., by defining an ontology to describe how sentiment analysis prototypes can use datasets and machine learning approaches) which is out of the scope of AI-KG.

For instance, a triple `< deep_model_cnn, usedBy Task, toxicity_detection>` from the paper [57] should be interpreted in the context of the same paper [57] i.e., *deep model cnn* is used for *toxicity detection* in [57] and, more broadly, some *deep model cnn* can be used for *toxicity detection*. Neither an interpretation like all *deep model cnn* are used for *toxicity detection* nor *deep model cnn* must be used for *toxicity detection* are correct according to the design and use of the current implementation of AI-KG.

AI-KG is adopted by several organizations for characterizing the AI domain and it has been used for supporting several research efforts, e.g., for extracting entities from scientific publications [67], describing competencies [52], and classifying scholarly articles [68]. AI-KG was generated by using Natural Language Processing (NLP) and Machine Learning (ML) methods for extracting entities and their relationships [69]. More specifically, AI-KG adopts a pipeline process that is applied on natural language scientific texts to (i) detect entities using a domain-specific extractor based on transformers [70] and a topic classifier developed on top of the CSO ontology [71]; (ii) identify relationships between entities by using open- and domain-specific ML and NLP tools [70], [72], [73], and (iii) define which facts make sense

according to an ontology representing the domain semantics. In addition, to determine whether a fact makes sense, the authors adopted a *support score* defined as the number of research papers where the fact was extracted from.

The reader can find more details about this methodology in [2], [69]. The current version of AI-KG consists of research entities belonging to one of the following classes:

- `Task`: A research challenge or a certain work to perform.
- `Method`: A research proposal or approach whose aim is to perform a certain task.
- `Material`: Resources that are employed for a certain research task, e.g., a dataset, an image, a text corpus.
- `Metric`: Entities that can be quantified and are used to measure the quality of a certain method.
- `OtherEntity`: A class used to group entities that cannot be classified in any of the previous ones.

The relations were created by clustering frequent verbs and asking human experts to define domain and range restrictions as well as transitiveness. Some examples of object properties are `evaluatesMethod`, `includesMaterial`, or `usesMethod`. The ontology of AI-KG is available online.[8]

Although the extracted facts compose a large-scale KG, the mining of such knowledge from natural language is an error-prone and challenging task and, therefore, it tends to have low coverage, i.e. well-known facts might not be materialized within the KG. As a result, AI-KG is sparse and incomplete. For example, the well-known fact `<neural_network, usesMaterial, rdf_graph>` cannot be found in the current AI-KG resource despite the fact that RDF graphs are the input of most of the existing neural network-based link prediction and triple classification algorithms.

For this reason, scientific KGs are calling for specific approaches for their completion [47]. However, state-of-the-art methods developed for general-domain KGs such as TransE, TransR, RotatE, and so on fail to predict triples with a good accuracy on AI-KG.

As reported in Section IV, these methods yield decent F1-measures, but suffer from a low precision (typically around 45-60%). Their adoption would thus introduce too many incorrect facts in the graph. The poor results of the existing techniques motivated this use case.

We applied SciCheck to AI-KG and, using a confidence threshold of 0.7, materialized 303,760 additional facts. Specifically, we used SciCheck to connect the most frequent 500 entities according to the relations defined in the AI-KG ontology. These include many significant facts there were missed by the information extraction pipeline, such as `< search_engine, includesMaterial, knowledge_base>`, `< f_measure, evaluatesMethod, neural_network>`, `<neural_network, uses Material, rdf_graph>`, or `< recommender_system, usesMethod, predictive_model>`.

[8]http://scholkg.kmi.open.ac.uk/aikg/ontology

The new version of AI-KG is available online at https://zenodo.org/record/7276434.

VI. CONCLUSION

In this paper, we introduced SciCheck, a new approach for completing scientific facts in knowledge graphs of research concepts. We evaluated SciCheck on two new benchmarks extracted from the Artificial Intelligence Knowledge Graph (AI-KG) [2], a large-scale KG of research concepts, (AIKG-1M and AIKG-500) and five well-known general benchmarks for link prediction (FB13, WN11, WN18, WN18RR, and NELL). The experiments show that SciCheck outperforms nine alternative approaches in terms of precision. Furthermore, we have shown a real-world use case and used SciCheck to complete AI-KG, producing a new version of it including more than 300K additional statements (a 28% increase).

As future work, we plan to study the application of KG completion techniques to hypothesis generation and extend SciCheck in this space. We also plan to consider weighted triples [50], [74] that could formalize the degree of certainty in specific statements. In addition, we intend to incorporate new features that could further improve recall. Finally, we look forward to applying our methodology to other scientific KGs, such as Open Research Knowledge Graph [4] and Nanopublications [13].

REFERENCES

[1] H. Kitano, "Artificial intelligence to win the Nobel prize and beyond: Creating the engine for scientific discovery," *AI Mag.*, vol. 37, no. 1, pp. 39–49, Apr. 2016.

[2] D. Dessí, F. Osborne, D. R. Recupero, D. Buscaldi, E. Motta, and H. Sack, "AI-KG: An automatically generated knowledge graph of artificial intelligence," in *Proc. ISWC*, vol. 12507. Cham, Switzerland: Springer, 2020, pp. 127–143, doi: 10.1007/978-3-030-62466-8_9.

[3] R. D. Haan, I. Tiddi, and W. Beek, "Discovering research hypotheses in social science using knowledge graph embeddings," in *Proc. Eur. Semantic Web Conf.* Cham, Switzerland: Springer, 2021, pp. 477–494.

[4] M. Y. Jaradeh, A. Oelen, K. E. Farfar, M. Prinz, J. D'Souza, G. Kismihók, M. Stocker, and S. Auer, "Open research knowledge graph: Next generation infrastructure for semantic scholarly knowledge," in *Proc. 10th Int. Conf. Knowl. Capture*, Sep. 2019, pp. 243–246.

[5] A. G. Nuzzolese, A. L. Gentile, V. Presutti, and A. Gangemi, "Semantic web conference ontology—A refactoring solution," in *Proc. Eur. Semantic Web Conf.* Cham, Switzerland: Springer, 2016, pp. 84–87.

[6] D. Ayala, A. Borrego, I. Hernández, and D. Ruiz, "A neural network for semantic labelling of structured information," *Expert Syst. Appl.*, vol. 143, Apr. 2020, Art. no. 113053, doi: 10.1016/j.eswa.2019.113053.

[7] F. Hoppe, D. Dessì, and H. Sack, "Deep learning meets knowledge graphs for scholarly data classification," in *Proc. Companion Proc. Web Conf.*, Apr. 2021, pp. 417–421.

[8] F. Belleau, M.-A. Nolin, N. Tourigny, P. Rigault, and J. Morissette, "Bio2RDF: Towards a mashup to build bioinformatics knowledge systems," *J. Biomed. Informat.*, vol. 41, no. 5, pp. 706–716, Oct. 2008.

[9] D. Shotton, "Semantic publishing: The coming revolution in scientific journal publishing," *Learned Publishing*, vol. 22, no. 2, pp. 85–94, Apr. 2009.

[10] A. Kelley and D. Garijo, "A framework for creating knowledge graphs of scientific software metadata," *Quant. Sci. Stud.*, vol. 2, no. 4, pp. 1423–1446, 2021.

[11] K. Wolstencroft, R. Haines, D. Fellows, A. Williams, D. Withers, S. Owen, S. Soiland-Reyes, I. Dunlop, A. Nenadic, P. Fisher, J. Bhagat, K. Belhajjame, F. Bacall, A. Hardisty, A. N. de la Hidalga, M. P. B. Vargas, S. Sufi, and C. Goble, "The Taverna workflow suite: Designing and executing workflows of web services on the desktop, web or in the cloud," *Nucleic Acids Res.*, vol. 41, no. W1, pp. W557–W561, Jul. 2013.

[12] P. Groth, A. Gibson, and J. Velterop, "The anatomy of a nanopublication," *Inf. Services Use*, vol. 30, nos. 1–2, pp. 51–56, Sep. 2010.
[13] T. Kuhn, C. Chichester, M. Krauthammer, N. Queralt-Rosinach, R. Verborgh, G. Giannakopoulos, A.-C. Ngonga Ngomo, R. Viglianti, and M. Dumontier, "Decentralized provenance-aware publishing with nanopublications," *PeerJ Comput. Sci.*, vol. 2, p. e78, Aug. 2016.
[14] J. Schneider, P. Ciccarese, T. Clark, and R. D. Boyce, "Using the micropublications ontology and the open annotation data model to represent evidence within a drug-drug interaction knowledge base," in *Proc. LISC@ISWC*, 2014, pp. 1–11.
[15] A. A. Salatino, T. Thanapalasingam, A. Mannocci, F. Osborne, and E. Motta, "The computer science ontology: A large-scale taxonomy of research areas," in *Proc. Int. Semantic Web Conf.* Cham, Switzerland: Springer, 2018, pp. 187–205.
[16] S. Peroni and D. Shotton, "The spar ontologies," in *Proc. Int. Semantic Web Conf.* Cham, Switzerland: Springer, 2018, pp. 119–136.
[17] A. Hogan, "Knowledge graphs," *ACM Comput. Surv.*, vol. 54, no. 4, pp. 1–37, 2021.
[18] S. Vahdati, N. Arndt, S. Auer, and C. Lange, "Openresearch: Collaborative management of scholarly communication metadata," in *Proc. Eur. Knowl. Acquisition Workshop*. Cham, Switzerland: Springer, 2016, pp. 778–793.
[19] O. Bodenreider, "The unified medical language system (UMLS): Integrating biomedical terminology," *Nucleic Acids Res.*, vol. 32, pp. 267D–270, Jan. 2004.
[20] A. Salatino, F. Osborne, and E. Motta, "Researchflow: Understanding the knowledge flow between academia and industry," in *Proc. Int. Conf. Knowl. Eng. Knowl. Manag.*, 2020, pp. 219–236.
[21] A. Rossanez, J. C. dos Reis, and R. da Silva Torres, "Representing scientific literature evolution via temporal knowledge graphs," in *Proc. MEPDaW@ ISWC*, 2020, pp. 33–42.
[22] D. Ayala, A. Borrego, I. Hernández, C. R. Rivero, and D. Ruiz, "AYNEC: All you need for evaluating completion techniques in knowledge graphs," in *Proc. ESWC*, vol. 11503. Cham, Switzerland: Springer, 2019, pp. 397–411, doi: 10.1007/978-3-030-21348-0_26.
[23] Y. Dai, S. Wang, N. N. Xiong, and W. Guo, "A survey on knowledge graph embedding: Approaches, applications and benchmarks," *Electronics*, vol. 9, no. 5, p. 750, May 2020.
[24] A. Bordes, N. Usunier, A. García-Durán, J. Weston, and O. Yakhnenko, "Translating embeddings for modeling multi-relational data," in *Proc. NIPS*, 2013, pp. 2787–2795.
[25] M. Nickel, V. Tresp, and H.-P. Kriegel, "Factorizing YAGO: Scalable machine learning for linked data," in *Proc. 21st Int. Conf. World Wide Web*, Apr. 2012, pp. 271–280.
[26] T. Trouillon, J. Welbl, S. Riedel, E. Gonz Gaussier, and G. Bouchard, "Complex embeddings for simple link prediction," in *Proc. 33rd Int. Conf. Int. Conf. Mach. Learn. (ICML)*, vol. 48, Jun. 2016, pp. 2071–2080.
[27] A. Borrego, D. Ayala, I. Hernández, C. R. Rivero, and D. Ruiz, "CAFE: Knowledge graph completion using neighborhood-aware features," *Eng. Appl. Artif. Intell.*, vol. 103, Aug. 2021, Art. no. 104302.
[28] Y. Shen, D. Wen, Y. Li, N. Du, H.-T. Zheng, and M. Yang, "Path-based attribute-aware representation learning for relation prediction," in *Proc. SIAM Int. Conf. Data Mining*. Philadelphia, PA, USA: SIAM, 2019, pp. 639–647.
[29] J. Zhou, G. Cui, Z. Zhang, C. Yang, Z. Liu, L. Wang, C. Li, and M. Sun, "Graph neural networks: A review of methods and applications," in 2018, *arXiv:1812.08434*.
[30] Q. Wang, Z. Mao, B. Wang, and L. Guo, "Knowledge graph embedding: A survey of approaches and applications," *IEEE Trans. Knowl. Data Eng.*, vol. 29, no. 12, pp. 2724–2743, Dec. 2017.
[31] S. M. Kazemi and D. Poole, "SimpLE embedding for link prediction in knowledge graphs," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 1–12.
[32] Y. Lin, Z. Liu, M. Sun, Y. Liu, and X. Zhu, "Learning entity and relation embeddings for knowledge graph completion," in *Proc. AAAI*, vol. 15, 2015, pp. 1–7.
[33] H. Liu, Y. Wu, and Y. Yang, "Analogical inference for multi-relational embeddings," in *Proc. ICML*, vol. 70, 2017, pp. 1–11.
[34] R. Socher, D. Chen, C. D. Manning, and A. Ng, "Reasoning with neural tensor networks for knowledge base completion," in *Adv. NIPS*, 2013, pp. 926–934.
[35] T. Mitchell, W. Cohen, E. Hruschka, P. Talukdar, B. Yang, J. Betteridge, A. Carlson, B. Dalvi, M. Gardner, B. Kisiel, and J. Krishnamurthy, "Never-ending learning," *Commun. ACM*, vol. 61, no. 5, pp. 103–115, 2018.
[36] T. Mikolov, L. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 2, Dec. 2013, pp. 3111–3119. [Online]. Available: http://dl.acm.org/citation.cfm?id=2999792.2999959
[37] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "RoBERTa: A robustly optimized BERT pretraining approach," 2019, *arXiv:1907.11692*.
[38] N. Lao and W. W. Cohen, "Relational retrieval using a combination of path-constrained random walks," *Mach. Learn.*, vol. 81, no. 1, pp. 53–67, Oct. 2010.
[39] M. Gardner and T. Mitchell, "Efficient and expressive knowledge base completion using subgraph feature extraction," in *Proc. EMNLP*, 2015, pp. 1488–1498.
[40] S. Mazumder and B. Liu, "Context-aware path ranking for knowledge base completion," in *Proc. 26th Int. Joint Conf. Artif. Intell.*, Aug. 2017, pp. 1195–1201.
[41] W. Xiong, T. Hoang, and W. Y. Wang, "DeepPath: A reinforcement learning method for knowledge graph reasoning," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2017, pp. 564–573.
[42] T. Hamaguchi, H. Oiwa, M. Shimbo, and Y. Matsumoto, "Knowledge transfer for out-of-knowledge-base entities: A graph neural network approach," in *Proc. 26th Int. Joint Conf. Artif. Intell.*, Aug. 2017, pp. 1802–1808, doi: 10.24963/ijcai.2017/250.
[43] C. Shang, Y. Tang, J. Huang, J. Bi, X. He, and B. Zhou, "End-to-end structure-aware convolutional networks for knowledge base completion," in *Proc. AAAI*, vol. 33, 2019, pp. 3060–3067. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/4164
[44] D. Nathani, J. Chauhan, C. Sharma, and M. Kaul, "Learning attention-based embeddings for relation prediction in knowledge graphs," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, 2019, pp. 4710–4723.
[45] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," in *Proc. Int. Conf. Learn. Represent.*, 2018, pp. 1–12.
[46] Z. Wang, Z. Ren, C. He, P. Zhang, and Y. Hu, "Robust embedding with multi-level structures for link prediction," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, Aug. 2019, pp. 5240–5246.
[47] M. Y. Jaradeh, K. Singh, M. Stocker, and S. Auer, "Triple classification for scholarly knowledge graph completion," in *Proc. 11th Knowl. Capture Conf.*, Dec. 2021, pp. 225–232, doi: 10.1145/3460210.3493582.
[48] M. Nayyeri, G. M. Cil, S. Vahdati, F. Osborne, M. Rahman, S. Angioni, A. Salatino, D. R. Recupero, N. Vassilyeva, E. Motta, and J. Lehmann, "Trans4E: Link prediction on scholarly knowledge graphs," *Neurocomputing*, vol. 461, pp. 530–542, Oct. 2021.
[49] M. Nayyeri, S. Vahdati, X. Zhou, H. S. Yazdi, and J. Lehmann, "Embedding-based recommendations on scholarly knowledge graphs," in *Proc. Eur. Semantic Web Conf.* Cham, Switzerland: Springer, 2020, pp. 255–270.
[50] M. Nayyeri, G. M. Cil, S. Vahdati, F. Osborne, A. Kravchenko, S. Angioni, A. Salatino, D. R. Recupero, E. Motta, and J. Lehmann, "Link prediction of weighted triples for knowledge graph completion within the scholarly domain," *IEEE Access*, vol. 9, pp. 116002–116014, 2021.
[51] K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor, "Freebase: A collaboratively created graph database for structuring human knowledge," in *Proc. ACM SIGMOD Int. Conf. Manage. Data (SIGMOD)*, 2008, pp. 1247–1250.
[52] N. Heist and P. Haase, "Flexible and extensible competency management with knowledge graphs," in *Proc. ISWC (Posters/Demos/Industry)*, vol. 2980, 2021, pp. 1–2. [Online]. Available: http://ceur-ws.org/Vol-2980/paper412.pdf
[53] G. Ji, S. He, L. Xu, K. Liu, and J. Zhao, "Knowledge graph embedding via dynamic mapping matrix," in *Proc. 53rd Annu. Meeting Assoc. Comput. Linguistics 7th Int. Joint Conf. Natural Lang. Process. (Long Papers)*, vol. 1, 2015, pp. 687–696.
[54] I. Bansal, S. Tiwari, and C. R. Rivero, "The impact of negative triple generation strategies and anomalies on knowledge graph completion," in *Proc. 29th ACM Int. Conf. Inf. Knowl. Manage.*, Oct. 2020, pp. 45–54.
[55] T. Alahakoon, R. Tripathi, N. Kourtellis, R. Simha, and A. Iamnitchi, "K-path centrality: A new centrality measure in social networks," in *Proc. 4th Workshop Social Netw. Syst. (SNS)*, 2011, pp. 1–6.
[56] L. A. Adamic and E. Adar, "Friends and neighbors on the web," *Soc. Netw.*, vol. 25, no. 3, pp. 211–230, 2003.
[57] D. Dessì, D. R. Recupero, and H. Sack, "An assessment of deep learning models and word embeddings for toxicity detection within online textual comments," *Electronics*, vol. 10, no. 7, p. 779, Mar. 2021.

[58] Z. Wang, J. Zhang, J. Feng, and Z. Chen, "Knowledge graph embedding by translating on hyperplanes," in *Proc. AAAI*, vol. 14, 2014, pp. 1112–1119.
[59] X. Han, S. Cao, X. Lv, Y. Lin, Z. Liu, M. Sun, and J. Li, "OpenKE: An open toolkit for knowledge embedding," in *Proc. Conf. Empirical Methods Natural Lang. Process., Syst. Demonstrations*, 2018, pp. 139–144.
[60] A. Borrego, D. Ayala, I. Hernández, C. R. Rivero, and D. Ruiz, "Generating rules to filter candidate triples for their correctness checking by knowledge graph completion techniques," in *Proc. 10th Int. Conf. Knowl. Capture*, Sep. 2019, pp. 115–122.
[61] A. A. Salatino, T. Thanapalasingam, A. Mannocci, A. Birukou, F. Osborne, and E. Motta, "The computer science ontology: A comprehensive automatically-generated taxonomy of research areas," *Data Intell.*, vol. 2, no. 3, pp. 379–416, Jul. 2020.
[62] A. Bordes, X. Glorot, J. Weston, and Y. Bengio, "A semantic matching energy function for learning with multi-relational data - application to word-sense disambiguation," *Mach. Learn.*, vol. 94, no. 2, pp. 233–259, 2014.
[63] T. Dettmers, P. Minervini, P. Stenetorp, and S. Riedel, "Convolutional 2D knowledge graph embeddings," in *Proc. AAAI*, 2018, pp. 1811–1818.
[64] P. Pezeshkpour, Y. Tian, and S. Singh, "Revisiting evaluation of knowledge base completion models," in *Proc. AKBC*, 2020, pp. 2052–2058.
[65] M. Speranskaya, M. Schmitt, and B. Roth, "Ranking vs. classifying: Measuring knowledge base completion quality," in *Proc. AKBC*, 2020.
[66] E. R. DeLong, D. M. DeLong, and D. L. Clarke-Pearson, "Comparing the areas under two or more correlated receiver operating characteristic curves: A nonparametric approach," *Biometrics*, vol. 44, pp. 837–845, Sep. 1988.
[67] X. Li and M. Daoutis, "Unsupervised key-phrase extraction and clustering for classification scheme in scientific publications," 2021, *arXiv:2101.09990*.
[68] F. Hoppe, D. Dessí, and H. Sack, "Understanding class representations: An intrinsic evaluation of zero-shot text classification," in *Proc. Workshop Deep Learn. Knowl. Graphs (DL4KG@ ISWC2021)*, vol. 3034, 2021, pp. 1–10.
[69] D. Dessì, F. Osborne, D. Reforgiato Recupero, D. Buscaldi, and E. Motta, "Generating knowledge graphs by employing natural language processing and machine learning techniques within the scholarly domain," *Future Gener. Comput. Syst.*, vol. 116, pp. 253–264, Mar. 2021.
[70] D. Wadden, U. Wennberg, Y. Luan, and H. Hajishirzi, "Entity, relation, and event extraction with contextualized span representations," in *Proc. Conf. Empirical Methods Natural Lang. Process. 9th Int. Joint Conf. Natural Lang. Process. (EMNLP-IJCNLP)*, 2019, pp. 5783–5788, doi: 10.18653/v1/D19-1585.
[71] A. A. Salatino, F. Osborne, T. Thanapalasingam, and E. Motta, "The CSO classifier: Ontology-driven detection of research topics in scholarly articles," in *Digital Libraries for Open Knowledge*, A. Doucet, A. Isaac, K. Golub, T. Aalberg, and A. Jatowt, Eds. Cham, Switzerland: Springer, 2019, pp. 296–311.
[72] G. Angeli, M. J. J. Premkumar, and C. D. Manning, "Leveraging linguistic structure for open domain information extraction," in *Proc. ACL*, vol. 1, 2015, pp. 344–354.
[73] K. Toutanova, D. Klein, C. D. Manning, and Y. Singer, "Feature-rich part-of-speech tagging with a cyclic dependency network," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics Hum. Lang. Technol. (NAACL)*, 2003, pp. 252–259.
[74] X. Chen, M. Chen, W. Shi, Y. Sun, and C. Zaniolo, "Embedding uncertain knowledge graphs," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 3363–3370.

AGUSTÍN BORREGO is currently pursuing the Ph.D. degree with the University of Seville. From 2018 to 2019, he was a Research Assistant at the DEAL Research Group, University of Seville. Since 2019, he has been a Ph.D. student at USE and a Visiting Student at The Open University, U.K. His current research interests involve knowledge graphs, including their completion and refinement.

DANILO DESSÌ received the master's and doctoral degrees from the University of Cagliari, Italy. His Ph.D. thesis was supervised by Prof. Diego Reforgiato Recupero. He has been a Researcher at the University of Cagliari, since August 2021. Previously, he was a Postdoctoral Researcher/a Senior Researcher at FIZ Karlsruhe–Leibniz Institute for Information Infrastructure and Karlsruhe Institute of Technology (KIT) with Prof. Dr. Harald Sack. He has been a Visiting Researcher at the following centers all around the world: Philips Research (Eindhoven, 2016), Center for Data Science NYU (New York City, 2017), Knowledge Media Institute–The Open University (Milton Keynes, 2018), and Laboratoire d'informatique de Paris Nord–University of Paris 13 (Paris, 2019). He is the coauthor of AI-KG and developer of the pipeline for its generation. His current research interests include artificial intelligence, knowledge graphs, science of science, deep learning, and natural language processing.

INMA HERNÁNDEZ is an Associate Professor at the University of Seville and a Founding Member at the Data Engineering Applications Laboratory. Her current research involves data engineering and knowledge graphs. She has authored many peer-reviewed publications on these topics in top conferences and journals.

She is a very active reviewer and a member of several program committees in major conferences. She is currently a Principal Investigator on a number of projects funded by the Spanish National Research and Development Program. Since 2020, she has been the coordinator of the master on software engineering: cloud, data and IT management at the Postgraduate School of the University of Sevilla.

FRANCESCO OSBORNE is a Research Fellow at the Knowledge Media Institute, The Open University, U.K., where he leads the Scholarly Data Mining Team. He is also an Assistant Professor at the University of Milano Bicocca. He collaborates with major publishers, universities, and companies in the space of innovation for producing a variety of innovative services for supporting researchers, editors, and research polities makers. He has released many well-adopted resources such as the computer science ontology and the artificial intelligence knowledge graph. His research interests include artificial intelligence, information extraction, knowledge graphs, science of science, and semantic web. He has authored more than 100 peer-reviewed publications in top journals and conferences of these fields.

DIEGO REFORGIATO RECUPERO received the Ph.D. degree in computer science from the University of Naples Federico II, Italy, in 2004. He has been a Full Professor at the Department of Mathematics and Computer Science, University of Cagliari, Italy, since February 2022. From 2005 to 2008, he was a Postdoctoral Researcher at the University of Maryland, College Park, USA. He co-founded six companies with the ICT sector. He is actively involved in European projects and research (with one of his companies he won more than 40 FP7 and H2020 projects). His current research interests include sentiment analysis, semantic web, natural language processing, human–robot interaction, financial technology, and smart grid. He is the author of more than 190 conference and journal papers in these research fields, with more than 2400 citations. He has won different awards in his career (such as the Marie Curie International Reintegration Grant, the Marie Curie Innovative Training Network, the Best Researcher Award from the University of Catania, the Computer World Horizon Award, the Telecom Working Capital, the Startup Weekend, and the Best Paper Award).

DAVID RUIZ is a Full Professor of software engineering at the University of Seville. He leads the Data Engineering Applications Laboratory, University of Seville, focusing his research on data engineering, knowledge graphs, and data integration. He has recently started two new related lines of research, focused on the application of machine learning techniques for the automated retrieval and processing of aviation data, and for the genomic analysis of multi-resistant bacteria. Since 2014, he has been the Deputy Director of the School of Computer Science, University of Seville, where he has contributed to the creation of a dual bachelor's degree in computer science and mathematics, and two new postgraduate master's courses.

DAVIDE BUSCALDI is an Associate Professor at LIPN, Sorbonne Paris North University and a part-time Assistant Professor at the Ecole Polytechnique, where he is teaching machine learning and data science courses, principally. He has directed or co-directed two Ph.D. theses and is currently directing three more theses in NLP and Machine Learning. He collaborates in various national and European projects. He is the author of more than 110 peer-reviewed conference and journal papers. His main research interests include natural language processing and text mining, in particular the application of modern NLP techniques to text annotation and relation extraction.

ENRICO MOTTA received the Laurea degree in computer science from the University of Pisa, Italy, and the Ph.D. degree in artificial intelligence from The Open University. He is currently a Professor of knowledge technologies at The Open University, U.K. From 2000 to 2007, he was the Former Director at the Knowledge Media Institute (KMi), The Open University. Over the years, he has been Leding KMi's contribution to numerous high-profile projects, receiving over 10.4M in external funding since 2000, from a variety of institutional funding bodies and commercial organizations. His research spans a variety of aspects at the intersection of large-scale data integration and modeling, semantic and language technologies, intelligent systems, and human–computer interaction.

Open Access funding provided by 'Università degli Studi di Cagliari' within the CRUI CARE Agreement