

FAVEL: Fact Validation Ensemble Learning

Umair Qudus[✉], Franck Lionel Tatkeu Pekarou, Ana Alexandra Morim da Silva[✉],
Michael Röder[✉], and Axel-Cyrille Ngonga Ngomo[✉]

Data Science Group, Department of Computer Science, Paderborn University, Germany
{umair.qudus, ltphen, ana.silva, michael.roeder,
axel.ngonga}@uni-paderborn.de
<https://dice-research.org/>

Abstract. Validating assertions before adding them to a knowledge graph is an essential part of its creation and maintenance. Due to the sheer size of knowledge graphs, automatic fact-checking approaches have been developed. These approaches rely on reference knowledge to decide whether a given assertion is correct. Recent hybrid approaches achieve good results by including several knowledge sources. However, it is often impractical to provide a sheer quantity of textual knowledge or generate embedding models to leverage these hybrid approaches. We present FAVEL, an approach that uses algorithm selection and ensemble learning to amalgamate several existing fact-checking approaches that rely solely on a reference knowledge graph and, hence, use fewer resources than current hybrid approaches. For our evaluation, we create updated versions of two existing datasets and a new dataset dubbed FAVEL-DS. Our evaluation compares our approach to 15 fact-checking approaches—including the state-of-the-art approach HybridFC—on 3 datasets. Our results demonstrate that FAVEL outperforms all other approaches significantly by at least 0.04 in terms of the area under the ROC curve. Our source code, datasets, and evaluation results are open-source and can be found at <https://github.com/dice-group/favel>.

Keywords: fact checking · ensemble learning · transfer learning · knowledge management.

1 Introduction

Knowledge graphs play a vital role in the web ecosystem.¹ The popularity and quantity of knowledge graphs have surged in recent years [35]. However, their usage is bound to the assumption that each individual statement within a knowledge graph is correct. Since large knowledge graphs are generated automatically (e.g., DBpedia [2,28], YAGO [48], and WikiData [33]), validating assertions before adding them to such a graph is an essential part of its creation and maintenance. At the same time, the sheer size of these graphs led to the development of automatic fact-checking approaches. These approaches rely on reference knowledge to decide whether a given assertion is true or false.

Recently, hybrid approaches achieved good results by including different knowledge sources, i.e., large reference knowledge graphs, knowledge graph embedding models,

¹ <http://webdatacommons.org/structureddata/2021-12/stats/stats.html>

and textual corpora [41,40]. However, providing a large amount of textual knowledge for a particular field of interest is not always feasible. Likewise, the generation of embedding models for large reference knowledge graphs can incur significant costs in both runtime and computational resources [13]. At the same time, several fact-checking approaches exist, that only rely on a reference knowledge graph. While these single approaches on their own showed inferior performance when compared to hybrid approaches in recent evaluations [46,41,40], their combination was not investigated before. Our work fills this research gap.

To the best of our knowledge, our approach FAVEL is the first attempt to combine several knowledge-graph-based fact-checking approaches. Internally, it is based on an ensemble learning algorithm to combine the prediction results of the different fact-checking approaches. Ensemble learning is a powerful technique in machine learning, that holds significant promise for enhancing the overall predictive performance in comparison to single approaches. The core motivation behind adopting the ensemble method stems from the acknowledgment that diverse approaches, when combined, can collectively outperform individual models by mitigating weaknesses, such as bias-variance tradeoff, and leveraging their respective strengths. We combine this with an algorithm selection approach to automatically configure FAVEL for the given training data.

Our contributions in this paper are as follows:

- We present FAVEL, a fact-checking approach that relies on ensemble learning to combine several knowledge-graph-based fact-checking approaches. Our evaluation shows that FAVEL is able to outperform the single fact-checking approach it combines and the state of the art HybridFC.
- We propose a new dataset dubbed FAVEL-DS for the evaluation of fact-checking approaches. We created this dataset based on the DBpedia of March 2022.
- We further present BPDP 22 and FactBench Mix 22, which are updated versions of previously published datasets that we aligned to the same DBpedia version.

The remainder of this paper is structured as follows. In the following Section, we briefly explain preliminaries before we summarize the related work in Section 3. In Section 4, we explain our approach FAVEL. We evaluate our approach in Section 5. In Section 6, we present an ablation study of our approach. Finally, we conclude and discuss potential future work in Section 7.

2 Preliminaries

Our work focuses on knowledge graphs, in particular on knowledge graphs in the sense of the Resource Description Framework (RDF). We define them as follows:

Definition 1 (Knowledge graph). *Let E be the set of all RDF resource IRIs, B the set of all blank nodes, $P \subseteq E$ the set of all RDF predicates and L the set of all literals. A knowledge graph G is a set of RDF assertions, i.e.,*

$$G = \{(s, p, o) \mid s \in E \cup B, p \in P, o \in E \cup B \cup L\}, \quad (1)$$

where (s, p, o) is a triple in G comprising a subject s , a predicate p and an object o [9].

Our approach aims to fulfill the task called automatic fact checking. We formally define this task as follows:

Definition 2 (Fact checking). *Given an assertion in the form of a triple, a reference knowledge graph, and/or a reference corpus, fact checking is the task of computing the likelihood that the given assertion is true or false [49].*

Definition 3 (Ensemble learning). *Let $\{h_1, h_2, \dots, h_n\}$ be n individual base learners, each producing an output $h_i(\mathbf{x})$ for a given input \mathbf{x} . The ensemble model, denoted as $H(\mathbf{x})$, is formed by combining the outputs of these base learners. The combination can be performed through various techniques, such as averaging, voting, or weighted averaging [42]. We define the ensemble prediction $H(\mathbf{x})$ as:*

$$H(\mathbf{x}) = m(h_1(\mathbf{x}), h_2(\mathbf{x}), \dots, h_T(\mathbf{x})), \quad (2)$$

where m represents the combination or aggregation function [42].

3 Related Work

A typical approach to fact checking involves searching for evidence for a given assertion in the provided reference data. We categorized the following four fact-checking approaches based on the reference data that they use: text-based, knowledge-graph-based, embedding-based, and hybrid approaches. Table 1 gives a brief overview, which we further explain in the following.

Text-based approaches transform the given assertion into one or several search queries that they use to derive textual evidence from a reference corpus [49,19]. De-Facto [19] and its extension FactCheck [49] are two examples of this category of algorithms. In contrast to these approaches, our work relies on a knowledge graph instead of a textual corpus as reference data.

Knowledge-graph-based approaches use a knowledge graph as reference knowledge to gather evidence. This category contains a variety of algorithms, ranging from those initially designed for graph link prediction, such as Adamic Adar [1], Degree Product [43], Jaccard [30], Katz [22], and Pathent [51]. Similarly, similarity measures like SimRank [21] have been used to compare the subject and object of the given assertion. Recently, more sophisticated knowledge-graph-based fact-checking approaches emerged, e.g., path-based approaches. Path-based approaches search for paths between the subject and object of the given assertion. Identified paths receive a score to express to which extent these paths can be used as evidence for the given assertion. KL [8] takes the specificity of a path, i.e., the degree of intermediate nodes on the path, into account. Knowledge Stream [45] makes use of a line graph to determine the similarity of properties in the knowledge graph. These similarities are used in a min cost max flow algorithm to score the connection between the subject and object. REL-KL [45] is based on a combination of KL’s specificity Knowledge Stream’s similarity measure. PRA [27] relies on path statistics that are gathered during random walks on the reference graph. Similarly, PredPath [44] and COPAAL [50] enumerate paths that connect subject and object and use features to assign scores to them. Rule-based approaches like AMIE3 [26]

Table 1: Existing fact-checking approaches.

Category	Approaches	Short Description
Text-based	DeFacto[19], FactCheck [49]	Transform a given assertion into search queries to retrieve textual evidence from a corpus.
Knowledge-graph-based	Adamic Adar [1], COPAAL [50], Degree Product [43], Jaccard [30], Knowledgestream [45], Katz [22], KL [8], PredPath [44], PRA [44], Pathent [51], REL-KL [27], SimRank [21]	Use a knowledge graph as reference knowledge and different statistical, path-, rule-, or pattern-based approaches to retrieve evidence from this graph.
	FAVEL (ours)	Combination of existing knowledge-graph-based approaches for improved performance.
Embedding-based	ESTHER [46], Dong et al. [11]	Utilize knowledge graph embedding models as reference data.
Hybrid	ExFakt [17], Tracy [18], Facy [29], ESTHER[46], HybridFC [41], TemporalFC [40])	Combine multiple types of reference data, such as a knowledge graph embedding model, textual evidence from a reference corpus, and paths in a reference knowledge graph.

and RuDiK [37] mine rules from the knowledge graph. An assertion is classified as true if there is a rule that supports the existence of the assertion. KV-Eval [24] extends this general approach further by adding rules that can reject the existence of an assertion. Pattern-based approaches mine patterns similar to rule-based approaches. These patterns can be more complex than rules. Examples are GFC [31] and OGFC [32]. Our approach fits into the category of knowledge-graph-based approaches and is designed to integrate any approach of this category. Our evaluation shows that the combination of approaches can lead to better results than the usage of single approaches.

Embedding-based fact-checking approaches use knowledge graph embedding models as reference data.² These approaches calculate the likelihood of the existence of the given assertion based on their reference embedding model. Such approaches have been used by da Silva et al. [46] and proposed by Dong et al. [11]. In contrast to these approaches, our approach does not rely on the expensive generation of a knowledge graph embedding model.

Hybrid fact-checking approaches use more than one of the aforementioned types of reference data. ExFaKT [17] and Tracy [18] combine rules mined on a reference knowledge graph and evidence from the web. FACTY [29] relies on textual evidence

² Some authors of knowledge graph embedding publications dubbed fact checking as a triple classification task.

and paths in a reference knowledge graph. ESTHER [46] is a path-based approach, which uses a knowledge graph embedding model to determine potential paths that could serve as evidence. HybridFC [41] makes use of all three categories of reference data. TemporalFC [40] is an extension of HybridFC and focuses on volatile assertions. TemporalFC uses temporal KG embeddings in addition to the other two sources. While hybrid approaches achieve good evaluation results, their need for different types of reference data makes their usage expensive and not always applicable. In contrast, our approach only makes use of a reference knowledge graph.

Ensemble learning is used in a variety of areas to combine several existing approaches for enhancing the overall predictive performance of models [42], e.g., in the area of knowledge graphs, Speck et al. [47] proposes a combination of several entity recognition systems. To the best of our knowledge, FAVEL is the first attempt to combine knowledge-graph-based fact-checking approaches.

4 Approach

FAVEL is based on the idea of ensemble learning. Instead of using a single hybrid approach, we use multiple knowledge-graph-based fact-checking approaches in parallel and combine their results to achieve a final classification for the given assertion.

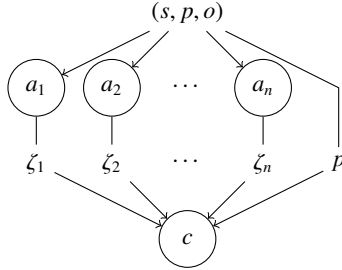


Fig. 1: Schematic overview of FAVEL.

Figure 1 gives an overview of our approach. FAVEL receives a given assertion in the form of a triple (s, p, o) as input. It forwards the assertion to the n knowledge-graph-based fact-checking approaches $\{a_1, \dots, a_n\}$ that FAVEL is configured to use. Since these approaches are treated as black boxes, FAVEL can make use of any fact-checking approach. However, within this paper, we restrict our work to knowledge-graph-based approaches since they all use the same reference data, and no additional data is needed. The veracity scores $\{\zeta_1, \dots, \zeta_n\}$ that the single approaches assign to the given assertion are collected. Together with the predicate p of the given assertion, these results are used as input for an ensemble classifier c . This classifier is trained to perform the final classification task. We formally define FAVEL as a fact checking function f as follows:

$$f(s, p, o) = c(a_1(s, p, o), \dots, a_n(s, p, o), p). \quad (3)$$

There are several ensemble learning algorithms available that can be used as c within our approach [42]. We tackle this algorithm-selection problem [23] with a meta-feature-free meta-learning technique proposed by Feurer et al. [14]. This approach trains several ensemble learners and successively removes bad-performing learners until either only one learner is left or the maximum training time is reached. In the latter case, the best-performing ensemble learner is returned from the list of remaining learners. This enables FAVEL to be trained and automatically configured specifically for a given dataset.

5 Evaluation

In this section, we evaluate our approach by comparing it with 12 knowledge-graph-based, 1 text-based, and 2 hybrid contemporary state-of-the-art fact-checking approaches. To begin, we outline the datasets integral to our evaluation, followed by a detailed exposition of our experimental configuration, and then present the results and discussion.

5.1 Datasets

All our experiments are based on the DBpedia [2,28] version March 2022.³ We make use of three benchmarking datasets. We reuse two established datasets and update them so that their content matches our DBpedia version. In addition, we create an additional dataset—FAVEL-DS. All datasets are provided as supplementary files to this submission. The created reference graph is too large to be uploaded, but it will be accessible online along with the source code, datasets, and evaluation results after the paper is accepted.

Dataset Updates FactBench [19] is a manually curated dataset based on DBpedia and Freebase with 10 predicates. The dataset is evenly distributed with each predicate having 150 correct statements. Gerber et al. propose the following six strategies to invalidate a correct assertion to create a false assertion [19]:

1. subject corruption with domain restriction,
2. object corruption with range restriction,
3. subject and object corruption with domain and range restrictions,
4. property corruption,
5. random subject, object and predicate corruption, and
6. temporal corruption.

For our experiments, we use the FactBench Mix dataset, which comprises a mixture of false assertions generated with strategies 1–5.

Birthplace-Deathplace (BPDP) is a dataset proposed by Syed et al. [49]. The dataset comprises 103 persons who have a birthplace and deathplace in two different countries. For each of these persons, the dataset comprises 2 correct assertions and 2 wrong assertions. The latter were created by swapping the correct birth and death places.

Both datasets have been created based on older versions of the DBpedia. Hence, we update the datasets to align their content with the previously chosen DBpedia version as follows:

³ <https://databus.dbpedia.org/dbpedia/mappings/mappingbased-objects/2022.03.01/>

Table 2: Post-processing statistics comprising the number of assertions of the train and test split and the number of distinct properties of BPDP 22, FactBench Mix 22 and FAVEL-DS. # True/# False assertions (Total count).

	BPDP 22	FactBench Mix 22	FAVEL-DS
Train	100/100 (200)	633/486 (1119)	380/385 (765)
Test	103/103 (206)	637/492 (1129)	163/164 (327)
Properties	2	9	11

1. We update the entity IRIs in the dataset to match the new DBpedia version. We replace assertions of FactBench that rely on Freebase with a DBpedia-based equivalent. To this end, we derive the DBpedia IRIs for the entities and properties of these assertions. Assertions for which a replacement cannot be created are removed.
2. We verify all assertions in the datasets by checking that all true assertions occur in the chosen DBpedia version and false assertions do not occur. Assertions for which this does not hold are removed.

The updated datasets are called FactBench Mix 22 and BPDP 22, respectively. Table 2 shows the size of the updated datasets.

FAVEL-DS In addition, we create a new dataset dubbed FAVEL-DS based on the following 11 properties of the DBpedia ontology: `academicDiscipline`, `affiliation`, `award`, `birthPlace`, `chancellor`, `city`, `deathPlace`, `director`, `producer`, `productionCompany`, and `starring`. For each property, we randomly select 50 assertions from the knowledge graph as true assertions. For each of these assertions (s, p, o) , we generate a false assertion (s, p, o') by replacing the object. We randomly choose the new object o' from existing triples from the knowledge graph (s, p', o') with the additional conditions that

1. $p \neq p'$,
2. o' has to fulfill the RDF-S range condition of p [5], and
3. (s, p, o') does not exist in the graph.

Our strategy to create false assertions is similar to the strategies applied by Gerber et al. [19] and Syed et al. [49] to create the FactBench Properties and the BPDP datasets, respectively. These strategies are known to create difficult negative examples since subject and object of the false assertion are connected in the knowledge graph [49]. A post processing check showed that a small number of the sampled and generated assertions had to be removed due to a repetition of assertions. The statistics of the final FAVEL-DS dataset can be found in Table 2. Figure 2 depicts the distribution of property occurrences across 3 datasets, encompassing a total of 16 properties.

Reference Graph The true assertions for all three datasets origin from the Dbpedia. Hence, if we use a complete DBpedia dump or data that is derived from it the fact

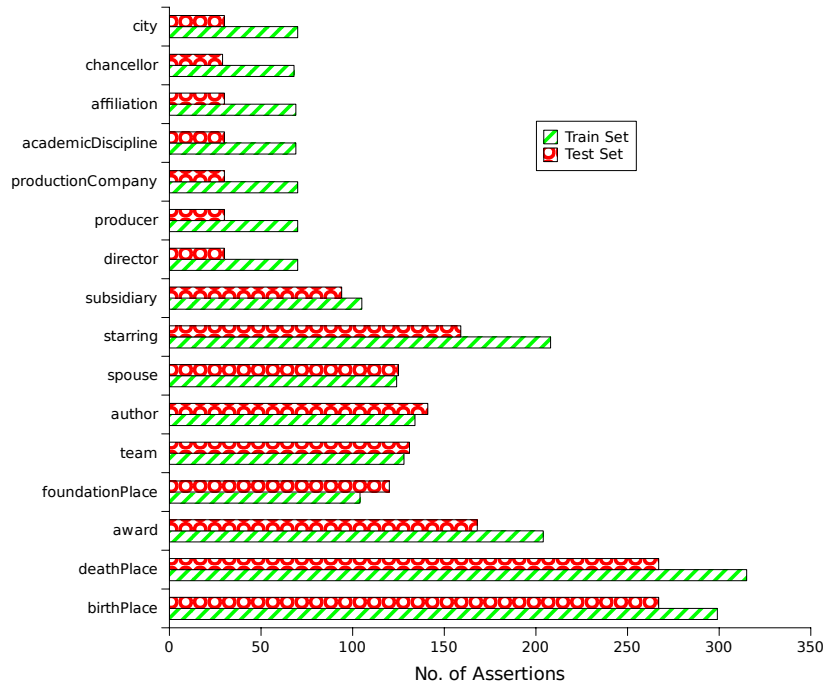


Fig. 2: Distribution of assertions with property occurrences across merged datasets.

checking task is reduced into a simple lookup whether a given assertion exists in the reference graph. To avoid this simplification, we remove all true assertions of the three datasets from the DBpedia to create the reference graph for our evaluation.

5.2 Setup

In our evaluation, we configure FAVEL to use the following 12 knowledge-graph-based approaches: Adamic Adar [1], Degree Product [43], Jaccard [30], Katz [22], KL [8], Knowledgestream [45], Pathent [51], PRA [27], PredPath [44], REL-KL [45], SimRank [21], and COPAAL [50]. We Auto-sklearn 2.0 [14] built on top of the Scikit-learn [6] as the meta-learner implementation within FAVEL. The AutoML library is configured to choose one of the following learners within a maximum runtime of 20 minutes:⁴ AdaBoost [15], Bernoulli Naive Bayes [34], Decision Tree, Extra Trees, Gaussian Naive Bayes, Gradient Boosting [16], K-Nearest Neighbors, Linear Discriminant Analysis [20], Linear Support Vector Classification [12], Support Vector Classification [7], Multi-Layer Perceptron, Multinomial Naive Bayes, passive-aggressive classifier, Quadratic Discriminant Analysis [20], Random Forest [4], and Stochastic Gradient

⁴ We use learners that are made available by Scikit-learn [39]. For learners that do not have a bibliographic reference, we refer the interested reader to the Scikit-learn documentation at https://scikit-learn.org/stable/user_guide.html.

Descent. We compare the performance of FVEL to the performance of the single knowledge-graph-based approaches, the text-based approach FactCheck [49], as well as the hybrid approaches ESTHER [46] and HybridFC [41]. The latter represents the current state of the art in this area of research. Da Silva et al. [46] suggest that a system’s performance can be improved if it is further combined with ESTHER. Hence, we evaluate a version of FVEL in which it uses the input of ESTHER as 13th system. Like Qudus et al. [41], we use the English Wikipedia for approaches that rely on a reference corpus. For hybrid approaches that use a knowledge graph embedding model, we use TransE [3] to generate a model. However, the results for HybridFC with the generated TransE model were not convincing as they were very different from the results reported by Qudus et al. in their publication. Hence, we rerun HybridFC with a ConEx embedding model [10], which leads to the results reported in the next section.

Evaluation Metric As suggested in the literature, we utilize the area under the receiver operator characteristic curve (AUC-ROC) to measure the performance of all systems [25,50,49]. We compute this score using the knowledge-base curation branch of the GERBIL framework [38,36].

5.3 Results

Table 3 shows the results of the single approaches⁵, the text-based FactCheck, ESTHER, HybridFC, and the two versions of FVEL. A Wilcoxon-signed-rank test comparing the veracity scores of FVEL with those of HybridFC and PredPath shows that FVEL’s scores are significantly different to the others.⁶ Regardless of whether ESTHER is added to the list of available fact-checking systems, FVEL uses a Decision Tree for BPDP 22, Gradient Boosting for FactBench Mix 22, and Random Forest for FVEL-DS.

During the evaluation, the knowledge-graph-based approaches showed a low runtime. For example, COPAAL needed 0.49 seconds per assertion on average.⁷ In comparison, HybridFC consumed more resources. While the knowledge-graph-based module of HybridFC relies on COPAAL and, hence, has the same low runtime, the embedding-based and the text-based modules lead to a higher resource consumption. For the embedding-based module, an embedding model has to be generated. Within our evaluation, the generation of the ConEx embedding model took 33 hours on a server with 2 GPUs.⁸ Apart from the resource-demanding model generation, the embedding-based module of HybridFC has a very low runtime per assertion since it mainly comprises a lookup of embedding vectors. The text-based module of HybridFC needed 15 seconds per assertion

⁵ The results of the single approaches can also be found on GERBIL at <https://gerbil-kbc.aksw.org/gerbil/experiment?id=202401120014>, <https://gerbil-kbc.aksw.org/gerbil/experiment?id=202401120015>, and <https://gerbil-kbc.aksw.org/gerbil/experiment?id=202401120026>.

⁶ We use $\alpha = 0.05$.

⁷ The per assertion runtimes were measured on a system with an AMD EPYC 7742 64-Core Processor 64 and 1 TB RAM.

⁸ The runtime of the model generation was measured on a system with an AMD EPYC 7713 64-Core Processor, 1 TB RAM and 2 NVIDIA GeForce RTX A5000 with 24GB VRAM, each.

Table 3: AUC-ROC scores achieved by the different approaches. The best results are marked bold while the results of the best KG-based approach are underlined. T stands for text-based and H for hybrid approaches.

	Approach	BPDP 22	FactBench Mix 22	FABEL-DS
KG-based approaches	Adamic Adar [1]	0.5000	0.6671	0.5579
	COPAAL [50]	0.5014	0.5797	0.5303
	Degree Product [43]	0.4965	0.4765	0.6096
	Jaccard [30]	0.5000	0.6729	0.5567
	Knowledgestream [45]	0.5114	0.7017	0.4859
	Katz [22]	0.4967	0.7882	0.5886
	KL [8]	0.4995	0.7097	0.4533
	PredPath [44]	<u>0.7136</u>	<u>0.8719</u>	<u>0.7399</u>
	PRA [27]	0.6845	0.8321	0.6942
	Pathent [51]	0.4956	0.7852	0.5872
	REL-KL [45]	0.5131	0.7294	0.4812
	SimRank [21]	0.4948	0.7213	0.5543
T	FactCheck [49]	0.4911	0.6501	0.5846
H	ESTHER [46]	0.4997	0.5855	0.5209
	HybridFC [41]	0.6811	0.8801	0.7098
Ours	FABEL	0.7539	0.9250	0.7718
	FABEL + ESTHER	0.7539	0.9239	0.7694

on average. The time is mainly needed to retrieve documents and extract the textual evidence from them.

5.4 Discussion

Our evaluation results give several insights. First, on all three datasets, FABEL shows a significantly better performance than the best-performing single knowledge-graph-based approach PredPath and the state-of-the-art approach HybridFC. Although not exactly the same, the results of HybridFC for BPDP 22 and FactBench Mix 22 are close to the results that Qudus et al. report in their evaluation on the original BPDP and FactBench Mix datasets [41].

The runtime comparison shows that the state-of-the-art approach HybridFC needs more resources concerning

1. the preparation of the system and
2. the classification of a single assertion.

HybridFC relies on a knowledge graph embedding model. The generation of this model is costly and for many models, the usage of modern GPUs is mandatory [10,13]. In practice, this resource demand could be reduced by relying on existing, pre-computed embedding models.⁹ However, the average runtime of HybridFC to classify an assertion

⁹ Note that this is not possible within our evaluation setup since the reference knowledge graph had to be adapted to the evaluation datasets as described in Section 5.1.

is also higher than for the knowledge-graph-based approaches. This makes FAVEL an alternative that uses fewer resources while it achieves a significantly better performance within our evaluation.

A second insight is that the configurations of FAVEL that AutoML chooses are different for all three datasets. At the same time, we didn’t see a configuration of FAVEL that is very good over all three datasets. A deeper per-property analysis of the results reveals that FAVEL performs well on most properties but shows a low performance for the `city` property in the FAVEL-DS dataset. Hence, we can conclude that for different data, different ensemble strategies are better than others. This raises the question of which features of the data influence these strategies and whether there is a way to predict the best strategy for a given triple. However, we leave answering these questions to future work.

With respect to our newly created dataset FAVEL-DS, a comparison of the performance of nearly all systems on BPDP 22 and FAVEL-DS shows that both datasets seem to have a nearly equal difficulty compared to FactBench Mix 22. At the same time, FAVEL-DS is bigger than BPDP 22 and covers 11 instead of only 2 properties. Hence, we argue that FAVEL-DS is a good contribution for the future evaluation of new fact-checking approaches.

For a more comprehensive analysis of FAVEL, we utilize a concrete example drawn directly from our selected dataset and the output of all competing approaches. In the example presented in Listing 1.1, we compare the results of all approaches and FAVEL for two assertions. According to the ground truth, the left assertion is true while the right assertion is false. The example shows that some systems give higher scores to the wrong assertion (Adamic Adar, Degree Product, Katz, KL, KL-REL, Knowledgestream) while others give nearly the same scores for both (Jaccard, PredPath, Simrank, PRA). In the example, only Pathent and COPAAL give a higher score to the correct assertion. However, these two systems didn’t achieve the highest scores over the complete datasets. This small example emphasizes the need of a meta learner that decides based on the given assertion which of the single approaches may give a reliable result. This is exactly the approach of FAVEL, which results in a higher score for the correct assertion in the example.

6 Ablation study

Our previous experiments suggest that FAVEL outperforms single KG-based, text-based, or hybrid approaches. To assess the impact of performance resulting from individual KG-based approaches on FAVEL, we conduct a series of experiments wherein we systematically remove individual approaches from the ensemble setting of FAVEL and rerun the experiments. We also perform another experiment on each property of all the datasets.

Listing 1.1: Example (correct and wrong assertions, from FAVEL-DS dataset.)

PREFIX	dbr:	< http://dbpedia.org/resource/ >
PREFIX	dbo:	< http://dbpedia.org/ontology/ >

<p>Correct assertion: dbr:Detouring_America dbo:productionCompany dbr:Warner_Bros._Cartoons</p> <p>Ground truth score: 1.0</p> <p>Approach-Score-Range Adamic Adar: 0.15 [0-1] Degree Product: 2455 (scale 0-100k) Jaccard: 0.002 [0-1] Katz: 0.154 [0-1] KL: 0.0724 [0-1] KL-REL: 0.115 [0-1] KS: 0.071 [0-1] Pathent: 3819 (scale 0-100k) PredPath: 1.0 [0/1] Simrank: 0.00046 [0-1] PRA: 0.0 [0/1] COPAAL: 0.6 [0-1] ----- FaVEL Score: 0.89 [0-1]</p>	<p>Wrong assertion: dbr:George_de_Hevesy dbo:deathPlace dbr:Budapest</p> <p>Ground truth score: 0.0</p> <p>Approach-Score-Range Adamic Adar: 0.57 [0-1] Degree Product: 78451 (scale 0-100k) Jaccard: 0.0009 [0-1] Katz: 0.954 [0-1] KL: 0.107 [0-1] KL-REL: 0.89 [0-1] KS: 0.14 [0-1] Pathent: 3441 (scale 0-100k) PredPath: 1.0 [0/1] Simrank: 0.0002 [0-1] PRA: 0.0 [0/1] COPAAL: 0.338 [0-1] ----- FaVEL Score: 0.18 [0-1]</p>
--	---

This experiment aims to evaluate FaVEL on a per-property basis. In this experiment, we independently merge the training and testing sets of all the datasets introduced earlier, group them by property, and separate them into multiple datasets with training and testing subsets for each property. The distribution of the datasets on the per-property basis can be found in Figure 2. We conduct these experiments using the same setup described in subsection 5.2, employing knowledge-graph-based fact-checking approaches. For the first set of experiments, we set the number of iterations to 10 and compute the results’ minimum, maximum, and mean. These 10 iterations take on average 200 minutes for each experiment on our server with the specifications described in the previous section. Table 5 and 4 show the results of our ablation study experiments.

Table 4 presents the performance of each system on a per-property basis. We can observe variations in performance across different properties among the various approaches. For example, Adamic Adar exhibits poor performance across all properties except for the city property, where it outperforms all other approaches with an AUC-ROC score of 0.81. PredPath, on the other hand, achieves top performance alongside FaVEL on the deathPlace, author, spouse, and director properties, with scores of 0.77, 1.0, 0.93, and 0.83, respectively. Additionally, PredPath surpasses FaVEL and all other approaches on the foundationPlace and affiliation properties, achieving scores of 0.93 and 0.90, respectively. However, it performs the worst among all other approaches on the academicDiscipline property. The discrepancies in performance across various properties among different approaches underscore our assumption that each approach possesses distinct advantages that FaVEL leverages. Moreover, this underscores the rationale behind our future work, where we aim to train the ensemble learner of FaVEL on a per-property basis and develop an algorithm selection mechanism for each property. This mechanism will enable FaVEL to determine which approach is most pertinent for a given property. By doing so, we aim to further enhance FaVEL’s performance.

Table 4: Results from benchmarking FAVEL and all other approaches on a per-properties basis; the abbreviations are: AA/Adamic Adar, DP/Degree Product, KL/Knowledge Linker, and KS/KnowledgeStream. The best results are marked bold.

	FAVEL	AA	KL	KL-Rel	KS	Pathent	PRA	PredPath	Simrank	DP	Jaccard	Katz
birthPlace	0.80	0.63	0.56	0.55	0.53	0.68	0.77	0.75	0.57	0.48	0.65	0.67
deathPlace	0.77	0.68	0.63	0.64	0.64	0.72	0.77	0.77	0.65	0.59	0.63	0.75
award	0.79	0.52	0.52	0.56	0.52	0.59	0.69	0.78	0.68	0.52	0.52	0.63
foundationPlace	0.83	0.71	0.63	0.86	0.71	0.82	0.82	0.93	0.77	0.61	0.70	0.93
team	0.92	0.69	0.72	0.68	0.69	0.70	0.74	0.85	0.73	0.44	0.69	0.74
author	1.00	0.62	0.77	0.69	0.71	0.82	0.93	1.00	0.82	0.26	0.67	0.75
spouse	0.93	0.61	0.67	0.59	0.58	0.78	0.91	0.93	0.83	0.34	0.65	0.75
starring	0.84	0.52	0.76	0.75	0.75	0.81	0.70	0.82	0.67	0.51	0.53	0.79
subsidiary	0.89	0.78	0.82	0.87	0.82	0.87	0.86	0.83	0.81	0.62	0.82	0.83
director	0.83	0.46	0.50	0.54	0.48	0.76	0.76	0.83	0.58	0.57	0.46	0.65
producer	0.87	0.59	0.47	0.52	0.50	0.54	0.83	0.79	0.50	0.72	0.59	0.59
productionCompany	0.86	0.50	0.63	0.50	0.58	0.62	0.730	0.83	0.71	0.56	0.55	0.64
academicDiscipline	0.79	0.60	0.64	0.61	0.56	0.60	0.73	0.53	0.74	0.57	0.60	0.60
affiliation	0.89	0.31	0.50	0.57	0.64	0.61	0.6	0.90	0.45	0.55	0.34	0.57
chancellor	0.73	0.56	0.27	0.35	0.39	0.43	0.50	0.67	0.58	0.57	0.53	0.47
city	0.59	0.81	0.64	0.75	0.78	0.71	0.63	0.60	0.55	0.63	0.67	0.70

Table 5: Results of an ablation study conducted over 10 iterations of FAVEL. The ‘Difference’ column indicates the variance between FAVEL’s overall scores in Table 3 and these average scores. w/o stands for without.

	Approach	AUC-ROC scores (10 iterations)			
		Min. score	Max. score	Avg. score	Difference
FAVEL	w/o Adamic Adar [1]	0.7705	0.7753	0.7748	0.0030
	w/o COPAAL [50]	0.7627	0.7692	0.7656	-0.0062
	w/o Degree Product [43]	0.7344	0.7344	0.7344	-0.0374
	w/o Jaccard [30]	0.7697	0.7697	0.7696	-0.0022
	w/o Knowledgestream [45]	0.7781	0.7781	0.7781	0.0063
	w/o Katz [22]	0.7622	0.7775	0.7645	-0.0073
	w/o KL [8]	0.7943	0.7943	0.7943	0.0225
	w/o PredPath [44]	0.7364	0.7364	0.7364	-0.0354
	w/o PRA [27]	0.7492	0.7525	0.7495	-0.0223
	w/o Pathent [51]	0.7474	0.7474	0.7474	-0.0244
	w/o REL-KL [45]	0.7768	0.7768	0.7768	0.0050
	w/o SimRank [21]	0.7418	0.7418	0.7418	-0.0300

Table 5 demonstrates that individual systems in our ablation study have only a small impact on the overall system performance (ranging from -0.0374 on Degree Product to 0.0225 on KL). However, significant variations exist on a per-property basis. For instance, removing Adamic Adar from FAVEL would increase the overall performance by only 0.003. However, it would also forfeit FAVEL’s effectiveness in handling the *city* property, where Adamic Adar performs exceptionally well, as indicated in Table 4.

7 Conclusion and Future Work

Within this paper, we present FAVEL—an approach that utilizes ensemble learning to combine knowledge-graph-based fact-checking approaches. Our evaluation depicts that FAVEL significantly outperforms the state-of-the-art approach HybridFC on three datasets. Notably, FAVEL requires less reference knowledge, as it does not require an additional textual corpus or a knowledge graph embedding model as supplementary reference data. Additionally, it relies on approaches with lower runtimes compared to HybridFC.

We also propose a new dataset FAVEL-DS that is bound to an explicit DBpedia version and can be used by the community in future experiments. We further updated the existing datasets BDP and FactBench Mix creating BDP 22 and FactBench Mix 22, two datasets which are explicitly bound to the same DBpedia version.

Our future work encompasses three main objectives. Firstly, we aim to integrate additional fact-checking approaches into our system. Secondly, we plan to focus on identifying features that can predict the optimal configuration of FAVEL for a given assertion. Finally, we intend to explore the implementation of a per-property basis ensemble learner for FAVEL, along with developing an algorithm selection mechanism to determine the best approach for each property.

Acknowledgments

This work is part of a project that has received funding from the European Union’s Horizon 2020 research and innovation programme (Marie Skłodowska-Curie, No. 860801), the German Federal Ministry of Education and Research (BMBF) within the project NEBULA (13N16364), the Ministry of Culture and Science of North Rhine-Westphalia (MKW NRW) within the project SAIL (NW21-059D).

References

1. Adamic, L.A., Adar, E.: Friends and neighbors on the web. *Social Networks* **25**(3) (2003)
2. Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., Ives, Z.: DBpedia: A Nucleus for a Web of Open Data. In: Aberer, K., Choi, K.S., Noy, N., Allemang, D., Lee, K.I., Nixon, L., Golbeck, J., Mika, P., Maynard, D., Mizoguchi, R., Schreiber, G., Cudré-Mauroux, P. (eds.) *The Semantic Web*, pp. 722–735. Springer Berlin Heidelberg, Berlin, Heidelberg (2007). https://doi.org/10.1007/978-3-540-76298-0_52, https://link.springer.com/chapter/10.1007/978-3-540-76298-0_52
3. Bordes, A., Usunier, N., Garcia-Durán, A., Weston, J., Yakhnenko, O.: Translating embeddings for modeling multi-relational data. In: *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*. p. 2787–2795. NIPS’13, Curran Associates Inc., Red Hook, NY, USA (2013)
4. Breiman, L.: Random Forests. *Machine Learning* **45**, 5–32 (October 2001). <https://doi.org/10.1023/A:1010933404324>, <https://doi.org/10.1023/A:1010933404324>
5. Brickley, D., Guha, R., McBride, B.: RDF Schema 1.1. W3C Recommendation, W3C (February 2014), <http://www.w3.org/TR/2014/REC-rdf-schema-20140225/>
6. Buitinck, L., Louppe, G., Blondel, M., Pedregosa, F., Mueller, A., Grisel, O., Niculae, V., Prettenhofer, P., Gramfort, A., Grobler, J., Layton, R., VanderPlas, J., Joly, A., Holt, B., Varoquaux, G.: API design for machine learning software: experiences from the scikit-learn project. In: *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*. pp. 108–122 (2013)
7. Chang, C.C., Lin, C.J.: Libsvm: A library for support vector machines. *ACM Trans. Intell. Syst. Technol.* **2**(3) (may 2011). <https://doi.org/10.1145/1961189.1961199>, <https://doi.org/10.1145/1961189.1961199>
8. Ciampaglia, G.L., Shiralkar, P., Rocha, L.M., Bollen, J., Menczer, F., Flammini, A.: Computational fact checking from knowledge networks. *PloS one* **10**(6), e0128193 (2015)
9. Cyganiak, R., Wood, D., Lanthaler, M.: RDF 1.1 Concepts and Abstract Syntax. W3C Recommendation, W3C (February 2014), <http://www.w3.org/TR/2014/REC-rdf11-concepts-20140225/>
10. Demir, C., Ngomo, A.C.N.: Convolutional complex knowledge graph embeddings. In: *European Semantic Web Conference*. pp. 409–424. Springer (2021)
11. Dong, T., Wang, Z., Li, J., Bauckhage, C., Cremers, A.B.: Triple classification using regions and fine-grained entity typing. *Proceedings of the AAAI Conference on Artificial Intelligence* **33**(01), 77–85 (Jul 2019). <https://doi.org/10.1609/aaai.v33i01.330177>, <https://ojs.aaai.org/index.php/AAAI/article/view/3771>
12. Fan, R.E., Chang, K.W., Hsieh, C.J., Wang, X.R., Lin, C.J.: Liblinear: A library for large linear classification. *J. Mach. Learn. Res.* **9**, 1871–1874 (jun 2008)

13. Ferrari, I., Frisoni, G., Italiani, P., Moro, G., Sartori, C.: Comprehensive analysis of knowledge graph embedding techniques benchmarked on link prediction. *Electronics* **11**(23) (2022). <https://doi.org/10.3390/electronics11233866>, <https://www.mdpi.com/2079-9292/11/23/3866>
14. Feurer, M., Eggenberger, K., Falkner, S., Lindauer, M., Hutter, F.: Auto-sklearn 2.0: Hands-free automl via meta-learning. *J. Mach. Learn. Res.* **23**(1) (jan 2022)
15. Freund, Y., Schapire, R.: A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *Journal of Computer and System Sciences* **55**, 119–139 (August 1997). <https://doi.org/10.1006/jcss.1997.1504>, <https://doi.org/10.1006/jcss.1997.1504>
16. Friedman, J.H.: Greedy Function Approximation: A Gradient Boosting Machine. *The Annals of Statistics* **29**, 1189–1232 (October 2001), <https://www.jstor.org/stable/2699986>
17. Gad-Elrab, M.H., Stepanova, D., Urbani, J., Weikum, G.: Exfakt: A framework for explaining facts over knowledge graphs and text. In: *WSDM '19*, ACM, New York, NY, USA (2019). <https://doi.org/10.1145/3289600.3290996>, <https://doi.org/10.1145/3289600.3290996>
18. Gad-Elrab, M.H., Stepanova, D., Urbani, J., Weikum, G.: Tracy: Tracing facts over knowledge graphs and text. In: *The World Wide Web Conference*. p. 3516–3520. *WWW '19*, Association for Computing Machinery, New York, NY, USA (2019). <https://doi.org/10.1145/3308558.3314126>, <https://doi.org/10.1145/3308558.3314126>
19. Gerber, D., Esteves, D., Lehmann, J., Böhmann, L., Usbeck, R., Ngonga Ngomo, A.C., Speck, R.: Defacto-temporal and multilingual deep fact validation. *Web Semantics: Science, Services and Agents on the World Wide Web* **35**(P2), 85–101 (Dec 2015). <https://doi.org/10.1016/j.websem.2015.08.001>, <https://doi.org/10.1016/j.websem.2015.08.001>
20. Hastie, T., Tibshirani, R., Friedman, J.: *The Elements of Statistical Learning*. Springer (2001)
21. Jeh, G., Widom, J.: Simrank: A measure of structural-context similarity. In: *Proceedings of the Eighth ACM SIGKDD Intern. Conf. on Knowledge Discovery and Data Mining* (2002)
22. Katz, L.: A new status index derived from sociometric analysis. *Psychometrika* **18** (1953)
23. Kerschke, P., Hoos, H.H., Neumann, F., Trautmann, H.: Automated algorithm selection: Survey and perspectives. *Evolutionary Computation* **27**(1), 3–45 (2019). https://doi.org/10.1162/evco_a_00242
24. Kim, J., Choi, K.s.: Unsupervised fact checking by counter-weighted positive and negative evidential paths in a knowledge graph. In: *Proceedings of the 28th International Conference on Computational Linguistics*. pp. 1677–1686. International Committee on Computational Linguistics, Barcelona, Spain (Online) (Dec 2020). <https://doi.org/10.18653/v1/2020.coling-main.147>, <https://www.aclweb.org/anthology/2020.coling-main.147>
25. Kim, J., Choi, K.s.: Unsupervised fact checking by counter-weighted positive and negative evidential paths in a knowledge graph. In: *Proceedings of the 28th International Conference on Computational Linguistics*. pp. 1677–1686. International Committee on Computational Linguistics, Barcelona, Spain (Online) (Dec 2020). <https://doi.org/10.18653/v1/2020.coling-main.147>, <https://aclanthology.org/2020.coling-main.147>
26. Lajus, J., Galárraga, L., Suchanek, F.: Fast and exact rule mining with amie 3. In: Harth, A., Kirrane, S., Ngonga Ngomo, A.C., Paulheim, H., Rula, A., Gentile, A.L., Haase, P., Cochez, M. (eds.) *The Semantic Web*. pp. 36–52. Springer International Publishing, Cham (2020)
27. Lao, N., Cohen, W.W.: Relational retrieval using a combination of path-constrained random walks. *Machine learning* **81**(1), 53–67 (2010)
28. Lehmann, J., Isele, R., Jakob, M., Jentzsch, A., Kontokostas, D., Mendes, P.N., Hellmann, S., Morsey, M., van Kleef, P., Auer, S., Bizer, C.: *DBpedia – A large-scale, mul-*

- tilingual knowledge base extracted from Wikipedia. *Semantic Web* **6**(2), 167–195 (2015). <https://doi.org/10.3233/SW-140134>
29. Li, F., Dong, X.L., Langen, A., Li, Y.: Knowledge verification for long-tail verticals. *Proc. VLDB Endow.* **10**(11), 1370–1381 (Aug 2017). <https://doi.org/10.14778/3137628.3137646>, <https://doi.org/10.14778/3137628.3137646>
 30. Liben-Nowell, D., Kleinberg, J.: The link prediction problem for social networks. In: *Proceedings of the Twelfth Intern. Conf. on Information and Knowledge Management* (2003)
 31. Lin, P., Qi, S., Shen, J., Wu, Y.: Discovering Graph Patterns for Fact Checking in Knowledge Graphs, pp. 783–801. Springer International Publishing (01 2018). https://doi.org/10.1007/978-3-319-91452-7_50
 32. Lin, P., Song, Q., Wu, Y., Pi, J.: Discovering patterns for fact checking in knowledge graphs. *J. Data and Information Quality* **11**(3) (May 2019). <https://doi.org/10.1145/3286488>, <https://doi.org/10.1145/3286488>
 33. Malyshev, S., Krötzsch, M., González, L., Gonsior, J., Bielefeldt, A.: Getting the most out of wikidata: Semantic technology usage in wikipedia’s knowledge graph. In: Vrandečić, D., Bontcheva, K., Suárez-Figueroa, M.C., Presutti, V., Celino, I., Sabou, M., Kaffee, L.A., Simperl, E. (eds.) *The Semantic Web – ISWC 2018*. pp. 376–394. Springer International Publishing, Cham (2018)
 34. Manning, C., Raghavan, P., Schütze, H.: *Introduction to Information Retrieval*. Cambridge University Press (2008)
 35. McCrae, J.P.: The Linked Open Data Cloud. Website (May 2021), <https://www.lod-cloud.net/>, last time accessed, August 24th 2021
 36. Ngonga Ngomo, A.C., Röder, M., Syed, Z.H.: Semantic web challenge 2019. Website (2019), <https://dice-group.github.io/semantic-web-challenge.github.io/>, last time accessed, May 22nd 2023
 37. Ortona, S., Meduri, V.V., Papotti, P.: Rudik: Rule discovery in knowledge bases. *Proc. VLDB Endow.* **11**(12), 1946–1949 (Aug 2018). <https://doi.org/10.14778/3229863.3236231>, <https://doi.org/10.14778/3229863.3236231>
 38. Paulheim, H., Ngonga Ngomo, A.C., Bennett, D.: Semantic web challenge 2018. Website (2018), <http://iswc2018.semanticweb.org/semantic-web-challenge-2018/index.html>, last time accessed, May 22nd 2023
 39. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E.: Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research* **12**, 2825–2830 (2011)
 40. Qudus, U., Röder, M., Kirrane, S., Ngomo, A.C.N.: Temporalfc: A temporal fact checking approach over knowledge graphs. In: Payne, T.R., Presutti, V., Qi, G., Poveda-Villalón, M., Stoilos, G., Hollink, L., Kaoudi, Z., Cheng, G., Li, J. (eds.) *The Semantic Web – ISWC 2023*. pp. 465–483. Springer Nature Switzerland, Cham (2023)
 41. Qudus, U., Röder, M., Saleem, M., Ngomo, A.C.N.: Hybridfc: A hybrid fact-checking approach for knowledge graphs. In: Sattler, U., Hogan, A., Keet, M., Presutti, V., Almeida, J.P.A., Takeda, H., Monnin, P., Pirrò, G., d’Amato, C. (eds.) *The Semantic Web – ISWC 2022*. pp. 462–480. Springer International Publishing, Cham (2022). https://doi.org/10.1007/978-3-031-19433-7_27, https://papers.dice-research.org/2022/ISWC_HybridFC/public.pdf
 42. Sagi, O., Rokach, L.: Ensemble learning: A survey. *WIREs Data Mining and Knowledge Discovery* **8**(4) (2018). <https://doi.org/https://doi.org/10.1002/widm.1249>, <https://wires.onlinelibrary.wiley.com/doi/abs/10.1002/widm.1249>
 43. Shi, B., Weninger, T.: Fact checking in large knowledge graphs - A discriminative predicate path mining approach. *CoRR* **abs/1510.05911** (2015)

44. Shi, B., Weninger, T.: Discriminative predicate path mining for fact checking in knowledge graphs. *Knowledge-based systems* **104**, 123–133 (2016)
45. Shiralkar, P., Flammini, A., Menczer, F., Ciampaglia, G.L.: Finding streams in knowledge graphs to support fact checking. In: 2017 IEEE International Conference on Data Mining (ICDM). pp. 859–864. IEEE (2017)
46. da Silva, A.A.M., Röder, M., Ngomo, A.C.N.: Using compositional embeddings for fact checking. In: Hotho, A., Blomqvist, E., Dietze, S., Fokoue, A., Ding, Y., Barnaghi, P., Haller, A., Dragoni, M., Alani, H. (eds.) *The Semantic Web – ISWC 2021*. pp. 270–286. Springer International Publishing, Cham (2021), https://papers.dice-research.org/2021/ISWC2021_Esther/ESTHER_public.pdf
47. Speck, R., Ngomo, A.C.N.: Ensemble learning of named entity recognition algorithms using multilayer perceptron for the multilingual web of data. In: *Proceedings of the Knowledge Capture Conference. K-CAP 2017*, Association for Computing Machinery, New York, NY, USA (2017). <https://doi.org/10.1145/3148011.3154471>, <https://doi.org/10.1145/3148011.3154471>
48. Suchanek, F.M., Kasneci, G., Weikum, G.: Yago: a core of semantic knowledge. In: *Proceedings of the 16th international conference on World Wide Web*. pp. 697–706. ACM (2007)
49. Syed, Z.H., Röder, M., Ngomo, A.C.N.: FactCheck: Validating RDF Triples Using Textual Evidence. In: *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*. p. 1599–1602. CIKM '18, Association for Computing Machinery, New York, NY, USA (2018). <https://doi.org/10.1145/3269206.3269308>, https://svn.aksw.org/papers/2018/CIKM_FACTCHECK/public.pdf
50. Syed, Z.H., Röder, M., Ngomo, A.C.N.: Unsupervised discovery of corroborative paths for fact validation. In: Ghidini, C., Hartig, O., Maleshkova, M., Svátek, V., Cruz, I., Hogan, A., Song, J., Lefrançois, M., Gandon, F. (eds.) *The Semantic Web – ISWC 2019*. pp. 630–646. Springer International Publishing, Cham (2019). https://doi.org/10.1007/978-3-030-30793-6_36, https://papers.dice-research.org/2019/ISWC2019_COPAAL/public.pdf
51. Xu, Z., Pu, C., Yang, J.: Link prediction based on path entropy. *Physica A: Statistical Mechanics and its Applications* **456**, 294–301 (2016)