Veracity Estimation for Entity-Oriented Search with Knowledge Graphs

Stefano Marchesin
stefano.marchesin@unipd.it
University of Padua
Padua, Italy

Gianmaria Silvello
gianmaria.silvello@unipd.it
University of Padua
Padua, Italy

Omar Alonso[*]
omralon@amazon.com
Amazon
Palo Alto, California, USA

ABSTRACT

In this paper, we discuss the potential costs that emerge from using a Knowledge Graph (KG) in entity-oriented search without considering its data veracity. We argue for the need for KG veracity analysis to gain insights and propose a scalable assessment framework. Previous assessments focused on relevance, assuming correct KGs, and overlooking the potential risks of misinformation. Our approach strategically allocates annotation resources, optimizing utility and revealing the significant impact of veracity on entity search and card generation. Contributions include a fresh perspective on entity-oriented search extending beyond the conventional focus on relevance, a scalable assessment framework, exploratory experiments highlighting the impact of veracity on ranking and user experience, as well as outlining associated challenges and opportunities.

CCS CONCEPTS

• **Information systems → Evaluation of retrieval results**.

KEYWORDS

Entity-Oriented Search, Knowledge Graph, Veracity Estimation, Entity Card Generation

ACM Reference Format:
Stefano Marchesin, Gianmaria Silvello, and Omar Alonso. 2024. Veracity Estimation for Entity-Oriented Search with Knowledge Graphs. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management (CIKM '24), October 21–25, 2024, Boise, ID, USA.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3627673.3679561

1 INTRODUCTION

Web search encompasses diverse content types, extending beyond web page ranking to data from databases, Knowledge Graphs (KGs), entity cards, query responses, and various media formats [55]. Our focus lies in *entity-oriented search*, organizing information around entities, attributes, and relationships [5]. This approach underlies effective entity cards, enhancing the search experience by providing concise summaries on result pages, aiding navigation, and facilitating exploratory search [9, 37]. Despite the central role of KGs [55], widely-used ones, like Wikidata [66], suffer from inherent errors due to {semi-, fully-}automatic construction [68]. Using imperfect KGs in entity-oriented search can result in misinformation issues, impacting the user experience and downstream tasks [31, 46].

Despite the crucial role of KGs in entity-oriented search, the issue of KG veracity, which refers to the accuracy and reliability of KG data, remains understudied. Veracity is a well-known concept in databases [7, 12, 62], where ensuring the accuracy of KG data is paramount [69]. In contrast, previous assessments of system effectiveness in entity-oriented search have predominantly focused on relevance, assuming correct and reliable KGs [5]. This assumption neglects potential risks of misinformation due to errors in KGs. Driven by the recent advances in KG veracity estimation [25, 43, 47, 54], this paper addresses the oversight by exploring the implications of unreliable KG data in entity-oriented search [9, 31, 37, 46], proposing a scalable evaluation framework to assess and enhance the reliability of a KG.

Hence, we investigate three research questions.

RQ1: Can we measure the veracity of a KG employed in entity-oriented search?
RQ2: Can we devise veracity assessment methods that can scale to the size of current real-life KGs?
RQ3: What is the impact of KG veracity on entity search systems?

We argue that developing a KG veracity assessment framework is essential for gaining valuable insights into key aspects of entity-oriented search. Nevertheless, the conventional approach for evaluating data veracity involves manual annotation [69], which becomes prohibitively expensive given the size of current real-life KGs [68], encompassing millions of facts.

To address the research questions, we present an efficient KG assessment framework that estimates veracity through a partition-level optimization approach constrained by a budget. This allows strategic allocation of annotation resources to specific KG subsets, maximizing utility for downstream tasks. Together with the framework, we also propose a veracity-enhanced re-ranking strategy, *v*Rank, that boosts veracity while keeping relevance unscathed. We demonstrate our solution's effectiveness in the context of *dynamic entity summarization* [31], a task integral to entity card generation, which introduces a query-dependent aspect in the generation of summaries. The experimental evaluation reveals that veracity significantly influences entity search and card generation as a distinct dimension from relevance and utility. Through a user preference study on entity cards, we also show that veracity enhances the user experience by delivering higher-quality content without compromising relevance.

[*] Work does not relate to the author's position at Amazon.

CIKM '24, October 21–25, 2024, Boise, ID, USA
© 2024 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-0436-9/24/10
https://doi.org/10.1145/3627673.3679561

Our veracity assessment framework efficiently estimates KG veracity with minimal annotation costs, making it suitable for large-scale applications. The proposed framework is comprehensive for various entity-oriented search tasks, ensuring versatility and applicability across the spectrum of KG veracity assessment.

In summary, the **contributions** of this paper encompass:

- Introducing veracity as a distinct dimension in entity-oriented search tasks.
- Presenting a scalable assessment framework, integrating veracity into the entity search process.
- Conducting exploratory experiments that reveal the significant impact of veracity on different dimensions of entity-oriented search.

Outline. The rest of this paper is as follows. Section 2 reports on related work. Section 3 contextualizes the problem and presents the framework. Section 4 provides a proof of concept of the introduced framework. Section 5 presents the considered case study, together with the corresponding experiments and results. Finally, Section 6 concludes the paper and outlines possible future work directions.

2 RELATED WORK

This section reports related work from the three main areas of the present work: entity cards, data quality, and credible IR.

Entity Cards. Given the significance of entity cards in contemporary web search engines, extensive literature is available, primarily categorized into two areas [57]: (i) entity card generation and presentation and (ii) entity cards' impact on search behavior.

In generating and presenting entity cards, diverse approaches and benchmarks have emerged over recent years from various communities [5, 38, 55]. This involves selecting key facts crucial for a specific entity, with entity summarization aiming to generate an optimal, size-constrained summary by choosing a subset of triples [38]. Numerous entity summarization methods have been proposed, including RELIN by Cheng et al. [14], which leverages the PageRank algorithm for selecting top-h predicate-object pairs based on relatedness and informativeness. SUMMARUM [64], LinkSUM [63], FACES [29], and FACES-E [30] are examples of other methods, each employing distinct strategies such as PageRank ranking, facet-based partitioning, and classification for entity summarization.

Hasibi et al. [31] introduced the task of dynamic entity summarization, releasing an ad hoc benchmark and proposing a learning-to-rank approach, DynES, to generate query-dependent entity summaries. Through a user study, the authors found that users favor dynamic summaries over static ones.

Concerning the impact of entity cards on search behavior, Shokouhi and Guo [60] were pioneers in analyzing user interactions with proactive cards, finding similarities with reactive search log patterns. Bota et al. [9] explored entity cards' effects on search behavior and perceived user workload. User studies by Navalpakkam et al. [46] showed increased attention on relevant entity cards within a non-linear Search Engine Result Page (SERP). Recent research by Salimzadeh et al. [57] examined entity cards' impact on learning-oriented search tasks, noting significant effects on participant behaviors like dwell time and session duration. In the health domain, Jimmy et al. [35] observed users prioritizing entity cards when seeking information on specific conditions. Despite the extensive literature on the topic, none of the considered works discuss issues arising from entity cards containing incorrect information.

To the best of our knowledge, we are the first to introduce veracity as a distinct dimension, proposing a veracity-enhanced reranking strategy, vRank, and exploring its impact on both the generation and users' perception of entity cards.

Data Quality. Data quality is a long-standing research area [40], where vast literature has been published on its dimensions and metrics [11, 41, 70], as well as on methods and tools for the assessment, detection, and repair of data quality issues [1, 21, 56].

In the Big Data Era, the 4 V's (Volume, Velocity, Variety, Value) posed challenges to quality management, leading to the introduction of the fifth V: Veracity [7, 12, 62]. Knowledge Graphs (KGs) have become pivotal in Big Data applications, presenting unique challenges for quality management due to their semi-structured nature, use of Open World Assumption (OWA), substantial noise, and large-scale [69].

Assessing KG veracity is crucial for downstream tasks, impacting IR, RecSys, and Question Answering (QA). Reinanda et al. [55] emphasized the benefits of KG for modern information access systems. Retrieval systems can benefit from the information provided by KG veracity assessment, as the retrieved objects can be biased towards validated facts. RecSys can be revised to include the weighting of recommendations based on the veracity of the facts. For QA systems integrating KGs, veracity assessment identifies reliable information sources, enhancing precision and response time by focusing on accurate subsets in specific domains [58].

To conduct a veracity assessment, manual evaluation is the de facto standard [69]. However, due to the scale of real-life KGs, it is unfeasible to manually evaluate every triple of the KG. Therefore, efficient methods have emerged in recent years to estimate the KG veracity based on a (relatively) small sample [25, 43, 47, 54]. These methods perform iterative sampling and assessment procedures that result in KG veracity estimates with strong statistical guarantees and minimal human efforts. Although still in its infancy, efficient veracity estimation represents a promising solution that can be integrated within entity-centric retrieval, recommender, and QA systems to reduce costs and increase benefits [10].

We propose, for the first time, veracity estimation solutions for entity-oriented search, particularly for entity card generation.

Credible Information Retrieval. For long time, the Information Retrieval (IR) community has been investigating *information disorder* [67] and how it can increase the costs of the users' information-seeking process. This research conflates under Credible IR, referring to the process of obtaining reliable and trustworthy information from sources meeting criteria for reliability and trustworthiness [27]. Within Credible IR, several large-scale initiatives have taken place: the CLEF eHealth CHS tasks [28, 36, 48], the FIRE 2016 CHIS task [61], and the TREC Health Misinformation tracks [2, 15, 16], as well as the ROMCIR workshops [50–52, 59].

However, none of these initiatives focused on KG veracity and its impact on entity-oriented search. Rather, most of them revolve around semi-structured and unstructured textual data. Conversely, our work frames the data veracity problem within KGs in the context of entity-oriented search. Given the structured nature of the contents of most KGs, the task presents unique characteristics. One above all, the atomic nature of facts within KGs makes the assessment of their correctness dichotomous and, to a degree, simpler than that of documents. Indeed, documents can be seen as a collection of different facts, which might be independently regarded as correct or wrong, making it harder to decide on the veracity of the whole document. Hence, we believe that credible IR methods for {semi-, un-}structured textual data could benefit from the development of methods for KG veracity assessment, thus leading to new opportunities in a broad range of search tasks.

3 ENTITY SEARCH: THE ROLE OF VERACITY

In this section, we first frame our work in the context of the utilitarian analysis paradigm and the corresponding Delphic framework for web search [10]. Then, we outline the problem of KG veracity assessment and propose a scalable framework. We also further discuss differences from previous research.

3.1 Delphic Costs and Benefits

In the dynamic and evolving landscape of web search, there has been a growing recognition that the evaluation based solely on the ranking quality is insufficient [10]. The idea is to shift towards a more comprehensive assessment, considering the users' overall search experience and personal context. This paradigm, called *utilitarian analysis*, goes beyond relevance assessment and extends to diverse scenarios, encompassing explicit searches, content feeds, recsys, and, among others, entity-oriented search. The utilitarian analysis states that search operations entail non-monetary costs like time, cognitive effort, and interactivity. Amid these costs, the benefits are susceptible to various impairments, including misrepresentation, misinformation, and disinformation. These costs and benefits, termed as Delphic, are part of any search task across different domains, intents, and expertise [10].

Our research, embedded within this framework, focuses on the role that data veracity has on entity-oriented search [5, 55]. In this context, misinformation emerges as a significant Delphic cost. Indeed, KGs – in particular those built with {semi-, fully-}automatic approaches [68] – are prone to errors [19, 24, 53], which accentuate the Delphic costs related to misinformation. Consequently, all the tasks revolving around entity-oriented search suffer from these increasing costs. A prominent example is the generation of entity cards that, if presented with incorrect information, prevent the benefits these information capsules can bring to the user's experience and further intensify the costs associated with web search.

3.2 KG Veracity Framework

Addressing the data veracity issues associated with entity-oriented search is imperative to improve the overall quality of the user's experience on the Web. To do so, it is necessary to evaluate the veracity of KGs with a focus on the utility this process has on the considered downstream task. To assess the veracity of a KG for entity-oriented search tasks, we need to manually annotate its contents – i.e., its facts – for veracity. However, real-life KGs, such as Wikidata [66], DBpedia [4], YAGO [33], and NELL [45], encompass million of facts. Therefore, manually evaluating the veracity of large-scale KGs is prohibitively expensive.

To overcome this challenge, we would need to use sampling and estimation techniques, as highlighted in previous research [25, 54]. Specifically, active learning strategies that minimize annotation costs while providing statistical guarantees emerge as the most viable solution. These strategies ensure that assessments are cost-effective and represent the entire KG. Efficient sampling and estimation techniques, coupled with active learning strategies, thus offer a practical solution to navigate the challenges associated with KG veracity assessments.

We typically have a limited budget for labeling tasks in real-world tasks. However, we observe that different parts of the KG might have varying utilities concerning the downstream task of interest. For instance, popular entities within the KG usually have the highest query load [26, 34]. Therefore, in this case, we may want to ensure whether such entities (and the corresponding facts) are accurate to activate filtering and/or correction mechanisms in low-quality situations [24, 49].

All considered, we propose a scalable annotation strategy by employing KG partitioning into subgraphs. Dealing with subgraphs permits the allocation of resources strategically, thus directing the annotation efforts toward specific partitions that maximize the utility for downstream tasks. High-traffic or critical parts of the KG can be prioritized for annotation, thus ensuring the budget-constrained resources are used where they can have the most impact.

3.3 Framework Overview

Our KG veracity framework can be divided into three phases, which we depicted in Figure 1. In ❶, a utility model is defined based on the KG and web sources, and it is used to annotate the KG contents concerning their utility for downstream applications. In ❷, the KG is partitioned according to the utility scores associated with its contents. Finally, in ❸, the KG partitions undergo a veracity estimation procedure that relies on sampling, active learning, and estimators to produce veracity estimates for each partition. In Section 4, we present a possible utility model, an effective partitioning strategy, and we outline the efficient partition estimation problem and describe an iterative procedure that solves it. The partition-level veracity estimates obtained can then be used to enhance different applications, such as entity-oriented search tasks.

4 FRAMEWORK COMPONENTS

This section presents a proof of concept for the components of the KG veracity framework. We propose a utility model based on entity popularity on the Web and use utility scores to partition KG data via stratification. Then, we introduce the partition veracity estimation problem and an approach to solve it – for which we outline appropriate sampling strategies and estimators.

In the following, we consider a KG as a directed, edge-labeled multi-graph, usually defined as $G = (V, R, \eta)$, where $V = \{E \cup A \cup B\}$ is the set of nodes in G, where E are entities, A attributes, and B blank nodes; R is the set of relationships between nodes in G; and $\eta : R \rightarrow (E \cup B) \times (E \cup A \cup B)$ is a function assigning an ordered pair of nodes to each relationship. The η function produces the ternary relation T of G [8]. Without loss of generality, this work considers ground RDF graphs (i.e., without blank nodes), hence $\eta : R \rightarrow E \times (E \cup A)$. Thus, the ternary relation T is the set of

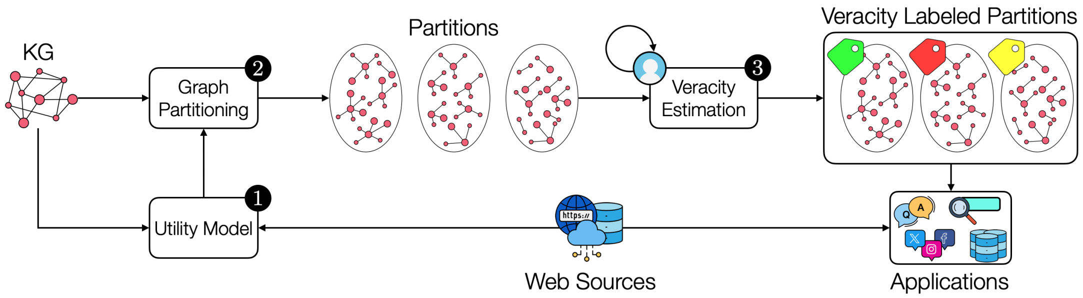

Figure 1: KG veracity assessment framework for downstream applications.

(s, p, o) triples such that $s \in E$, $p \in R$, and $o \in E \cup A$, where $M = |T|$ is its size. Triples whose object is an entity are called triples with entity property, whereas those with attribute objects are known as triples with data property. A triple is also a fact; the two terms are used interchangeably.

❶ **Utility Model.** The utility of a fact is a customized feature that should be defined based on the specific requirements of the downstream task of interest. Following Zheng et al. [71], we compute facts utility according to their popularity level on the Web. Given an entity $e \in E$, its utility is defined as $u(e) = L(\text{webSearch}(e))$, where $L(\text{webSearch}(e))$ denotes the length of the search results list for e on a web search engine. Given $x \in A$, we define $u(x) = 0$. Then, the utility of a fact $t = (s, p, o)$ is $u(t) = u(s) + u(o)$.

❷ **Graph Partitioning.** To partition a KG into subgraphs, we resort to stratification [17]. Stratification is a statistical technique to ensure that certain characteristics are well-represented in each population subset or stratum. With stratification, it may be possible to divide a heterogeneous population (like a KG) into strata, each internally homogeneous. If each stratum is homogeneous, meaning that the measurements vary little from one unit to another, a precise estimate of any stratum mean can be obtained from a small sample in that stratum [17]. Hence, stratification can be applied to achieve a representative and efficient partitioning of the KG.

Stratification divides the population into k subgroups based on features of interest for the specific objective. According to the considered utility model, we perform stratification based on the popularity level of facts on the Web. Once we obtain the popularity score of each fact, we MinMax normalize them and input them to the partitioning strategy to obtain the partition family $\mathcal{P} = \{P_1, P_2, \ldots, P_k\}$.

We adopt the *Cumulative Square root of Frequency* (Cumulative $\sqrt{F}$) method [18], previously used in a similar context by Gao et al. [25], as partitioning strategy. The Cumulative $\sqrt{F}$ method has strong theoretical groundings, aiming to achieve minimal intra-stratum variance in scores. The method first computes an empirical estimate of the cumulative square root of the distribution of scores. Then, it defines strata as equal-width bins based on the cumulative $\sqrt{F}$ scale and the number of desired strata k. Finally, the bins are mapped from the cumulative $\sqrt{F}$ scale to the score scale, and the facts are binned into the corresponding partitions.

❸ **Partition Veracity Estimation.** Let $P_i \in \mathcal{P}$ be a partition of G, the correctness of a triple $t \in P_i$ is denoted by an indicator function $\mathbb{1}(t) \longrightarrow \{0, 1\}$, where 1 indicates correctness and 0 incorrectness.[1] The partition veracity can then be defined as the mean accuracy of its triples $\mu(P_i) = \frac{\sum_{t \in P_i} \mathbb{1}(t)}{M_i}$, where $M_i = |P_i|$ is the partition size. $\mathbb{1}(t)$ is computed by manual annotation within each partition P_i.

However, the partitions obtained from a large-scale KG are also large-scale. Literature on stratification suggests partitioning the KG into a small number of strata k [17],[2] thus keeping partition sizes considerably large. Too many strata may lead to small sample sizes within each stratum, making it challenging to draw meaningful conclusions or even exceeding budget constraints before the annotation of all strata is completed.

Therefore, it becomes impractical to manually evaluate every triple of the partition to audit its veracity. This situation is further aggravated if we consider that, as outlined above, in several scenarios, we also have a limited annotation budget b that must be allocated across partitions. To overcome this limitation, common practice is to estimate $\mu(P_i)$ with an estimator $\hat{\mu}_i$ calculated over a relatively small sample drawn according to a sampling strategy $\mathcal{S}$ designed to select $\mathcal{S}(P_i) \subset P_i$ triples to annotate. To evaluate the veracity of P_i, the estimator $\hat{\mu}_i$ must be unbiased; that is, $E[\hat{\mu}_i] = \mu(P_i)$. Moreover, being $\hat{\mu}_i$ a point estimator, it also requires a $1 - \alpha$ Confidence Interval (CI) at a given significance level α to quantify the uncertainties in the sampling procedure. A relevant measure associated with CIs is the Margin of Error (MoE), which represents half the width of a CI.

Now, let $\mathcal{S}(P_i)$ be a sample drawn using a sampling design $\mathcal{S}$, and $\hat{\mu}_i$ be an estimator of $\mu(P_i)$ based on $\mathcal{S}$. Let $\text{cost}(\mathcal{S}(P_i))$ be a function denoting the cost of manually evaluating the correctness of the elements in $\mathcal{S}(P_i)$, and $b_i < b$ is the budget portion allocated to the partition P_i. Inspired by prior work [25, 54], we can define the problem of efficient partition veracity estimation as a constrained optimization problem, which we extend to also take into account budget constraints:

Problem. Given a a partition P_i, an upper bound ε_i for the MoE of a $1 - \alpha$ CI, and a budget b_i for annotating P_i:

$$\begin{aligned} &\underset{\mathcal{S}}{\text{minimize}} && \text{cost}(\mathcal{S}(P_i)) \\ &\text{subject to} && E[\hat{\mu}_i] = \mu(P_i) \wedge (\text{MoE}(\hat{\mu}_i, \alpha) \leq \varepsilon_i \vee |\mathcal{S}(P_i)| = b_i) \end{aligned}$$

[1] We consider correctness a binary problem as in the classic triple validation task [22], given that an atomic fact is correct or incorrect.

[2] Typical number of strata for KGs range from $k = 2$ to $k = 5$ [25, 54].

Algorithm 1 Partition Veracity Estimation

```
Input:
    A partition $P_i$;
    The significance level $\alpha$;
    Upper bound $\varepsilon_i$ for MoE;
    A budget $b_i$ for annotating $P_i$.
Output: The partition veracity estimate ($v_i$, MoE$_i$).
1: $v_i \leftarrow 0, \mathrm{MoE}_i \leftarrow 1, S_i \leftarrow \emptyset$          ▷ Initialization
2: while $\mathrm{MoE}_i > \varepsilon_i$ and $|S_i| < b_i$ do
3:     $B \leftarrow \mathcal{S}(P_i)$          ▷ Sample batch of facts from $P_i$ via $\mathcal{S}$
4:     $\bar{B} \leftarrow \mathbb{1}(B)$          ▷ Annotate facts and store annotations in $\bar{B}$
5:     $S_i \leftarrow S_i \cup \bar{B}$          ▷ Append annotations $\bar{B}$ to sample pool $S_i$
6:     $v_i \leftarrow \hat{\mu}_i(S_i)$          ▷ Estimate $P_i$ veracity from $S_i$
7:     $\mathrm{MoE}_i \leftarrow \mathrm{MoE}(\hat{\mu}_i, \alpha)$          ▷ Compute MoE
8: end while
9: return ($v_i$, MoE$_i$)
```

Solution. The problem can be addressed via an iterative procedure divided into four steps. We report pseudocode in Algorithm 1. At each iteration, a small batch of facts from the partition is sampled by a specific sampling design $\mathcal{S}$ (line 3). Secondly, the sampled facts are manually annotated and stored in the sample pool (lines 4-5). Given the annotated pool, the estimator $\hat{\mu}_i$ based on the considered sampling strategy $\mathcal{S}$ is used to compute an unbiased estimation of the partition veracity (line 6) and its associated MoE (line 7). Then, a quality control phase checks whether the assessment result satisfies the $\mathrm{MoE}(\hat{\mu}_i, \alpha) \leq \varepsilon_i$ or the $|\mathcal{S}(P_i)| = b_i$ constraints (line 2). If any of the two constraints is satisfied, the process is halted, and the veracity estimate is reported (lines 8-9). Otherwise, the process loops back to step two (line 3).

The considered procedure stops once the estimation result meets the user-specified threshold ε_i or the budget b_i allocated for the partition is exhausted. When the first condition happens ($\mathrm{MoE}(\hat{\mu}_i, \alpha) \leq \varepsilon_i$), the approach prevents oversampling and unnecessary manual annotations, consistently providing veracity estimations with robust statistical guarantees. When the second condition occurs ($|\mathcal{S}(P_i)| = b_i$), the approach ensures the best possible solution is obtained given the available budget. In both cases, the procedure operates efficiently, minimizing costs.

Sampling and Estimation. As sampling strategy $\mathcal{S}$, we resort to Simple Random Sampling (SRS). SRS draws a sample of n triples from P_i without replacement. However, if the partition P_i is large, we can safely use sampling with replacement to approximate sampling without replacement [13]. Based on the sample obtained with SRS, we can estimate the partition veracity using the sample mean $\hat{\mu}_i = \frac{1}{n}\sum_{j=1}^{n} \mathbb{1}(t_j)$, which is an unbiased estimator [17] – that is, $E[\hat{\mu}_i] = \mu(P_i)$ – with estimation variance $\mathrm{Var}(\hat{\mu}_i) = \frac{\hat{\mu}_i(1-\hat{\mu}_i)}{n}$.

By adopting SRS, $\hat{\mu}_i$ takes the form of the mean of n i.i.d. random variables with equal expectation $\mu(P_i)$. If the sample size is sufficiently large, then, by the Central Limit Theorem (CLT) [13], the $1-\alpha$ CI of $\mu(P_i)$ can be constructed as $\hat{\mu}_i \pm z_{\alpha/2}\sqrt{\mathrm{Var}(\hat{\mu}_i)}$, where $z_{\alpha/2}$ is the normal critical value with right-tail probability $\alpha/2$ and $z_{\alpha/2}\sqrt{\mathrm{Var}(\hat{\mu}_i)}$ represents the MoE of the considered CI. The larger the sample size n, the smaller the estimation variance $\mathrm{Var}(\hat{\mu}_i)$, and the more confident we can be in $\hat{\mu}_i$. In other words, the CI shrinks as n increases, and the iterative procedure used to minimize the optimization problem stops when $z_{\alpha/2}\sqrt{\mathrm{Var}(\hat{\mu}_i)} \leq \varepsilon_i$ or $n = b_i$.

We emphasize that the estimator and its corresponding variance rely solely on the sample and its size, independent of the underlying partition. Consequently, the iterative procedure is not contingent on the partition size. As a result, the performance achieved on relatively small partitions accurately reflects that on larger-scale ones, making our approach efficient at scale.

5 EVALUATION

We introduce the entity card generation as case study, describe the experimental setup, present the experiments targeting the research questions, and report the achieved results.

Case Study. We focus on the task of dynamic entity summarization [31] to investigate the impact of the proposed KG veracity framework. In its full meaning, dynamic entity summarization entails ranking entity facts based on their importance for the entity and relevance to the query.

5.1 Setup

Collection. We considered the collection proposed by Hasibi et al. [31] for dynamic entity summarization. The collection adopts DBpedia (version 2015-10) as the underlying KG, restricting entities to those with title, abstract, and at least 5 "valid" predicates. Queries are taken from the DBpedia-entity dataset [6], consisting of queries from four categories: named entity, entity list, natural language, and keyword. To build the collection, a single entity e_i is selected for each query q_i, thus forming query-entity pairs (q_i, e_i) that constitute the input for query-dependent entity summarization. The task is outlined as follows: for a given input pair (q_i, e_i), the system generates a ranking of predicate-object pairs (p, o) where the entity e_i acts as the subject. This ranking is determined by a combination of the relevance to the query and importance for the entity of predicate-objects pairs.

The collection consists of 100 query-entity pairs and $4,069$ corresponding facts, with an average of 41 facts per query-entity. Judgments for fact ranking are based on a combination of importance for the entity and relevance to the query, spread on a 5-point scale.

Utility Model. To compute utility scores, we automatically used the Google search engine to retrieve the first SERP corresponding to the target entity. The SERP was then used to extract information on the total number of search results related to the entity.

Graph Partitioning. We set the partitioning strata to $k = 5$. Opting for five strata allows for sufficiently nuanced partitioning, capturing different aspects without excessive granularity.

Partition Veracity Estimation. We set $\alpha = 0.05$, and $\varepsilon_i = 0.05$, for all P_i. This entails that we required a 95% confidence level for the estimator, admitting CIs not larger than 10% (i.e., MoE $\leq$ 5%). Given the small collection size, we initially assumed an unconstrained annotation budget.

We conducted manual annotations, avoiding taking information from Wikipedia and DBpedia. Due to the exploratory nature of this work, a single expert annotator was engaged in the task of annotating facts. To measure the cost of manually evaluating the correctness of facts within partitions, we used the cost function: $\mathrm{cost}(\mathcal{S}(P_i)) = |E_{\mathcal{S}}| \cdot c_1 + |T_{\mathcal{S}}| \cdot c_2$, where $E_{\mathcal{S}}$ and $T_{\mathcal{S}}$ represent the set of entities and triples in the sample $\mathcal{S}(P_i)$, while c_1 and c_2 are the

Table 1: Partition popularity and veracity estimates with corresponding 95% CIs.

Rank	Popularity score	Veracity estimate
1	0.2535	69% ± 5%
2	0.0607	80% ± 5%
3	0.0135	73% ± 5%
4	0.0021	77% ± 5%
5	0.0002	70% ± 5%

corresponding average costs. We set $c_1 = 45$ and $c_2 = 25$ (seconds), as estimated on real-life KGs by [25].

Summarization Methods. As summarizers, we adopted the main methods considered in [31]: DynES [31] and RELIN [14]. DynES is a learning-to-rank approach that generates query-dependent entity summaries, while RELIN uses PageRank to select top-h facts based on relatedness and informativeness. We considered cutoff values $h \in \{5, 10\}$ to compare and evaluate fact rankings.

Reproducibility. We release data and code used in this work.[3]

5.2 Experiments

RQ1: Veracity Measurement. We leverage the KG veracity framework to annotate partitions and produce veracity estimates. Subsequently, we analyze the distribution of partitions across the 100 query entities, exploring the correlation between veracity and popularity (i.e., utility). Additionally, using the partition veracity estimates, we measure entity-level veracity, defined as the mean of the veracity estimates of its triples, for which we also compute the corresponding $1 - \alpha$ CI, with $\alpha = 0.05$. Computing entity veracity allows us to investigate the relationship between veracity and utility at a more fine-grained, task-oriented level. Furthermore, this analysis can help decide whether to activate filtering and/or correction mechanisms in response to low-quality situations and how the *correction budget* should be allocated.

The veracity estimates obtained through the KG veracity framework for the considered partitions are detailed in Table 1, ranked from the most to least popular. Partition popularity is determined by the mean popularity of its facts. Note that low popularity scores are due to the effect of MinMax normalization, which rescales scores between 0 and 1.

Examining Table 1, we observe an interesting phenomenon: veracity and popularity (i.e., utility) appear nearly orthogonal. In other words, there is no discernible trend connecting veracity and popularity. This aligns with previous research by Dong et al. [20], who found a similar lack of correlation between source trustworthiness and popularity. Their study revealed that numerous popular websites were unreliable, whereas less popular ones provided highly accurate information. Interestingly, the most and least popular partitions exhibit the lowest veracity estimates in our case. Thus, an investigation at the entity level is needed to gain deeper insights into how veracity influences entity-oriented search tasks.

To this end, we consider how the entity facts (i.e., the (p, o) pairs) are distributed across the partitions, and Table 2 showcases the distribution across the partitions of the facts related to each query-entity pair in the collection. We can see that 84 entities comprise facts from various partitions, and only 16 entities have all the facts within a single partition. This underscores the faceted nature of entities, where their veracity is not atomic but rather necessitates aggregation from the associated facts.

Table 2: Number of entities whose facts are distributed across one or multiple partitions.

Number of entities w/ one partition	16
Number of entities w/ two partitions	32
Number of entities w/ three partitions	20
Number of entities w/ four partitions	22
Number of entities w/ five partitions	10
Number of entities (total)	100

Table 3: Partition statistics. The reported statistics are the partition sizes, the size of the samples used to estimate the veracity of partitions, the annotation costs in hours, and the veracity estimates with 95% CIs. For sample sizes, we also report the sample proportion w.r.t. the partition size (in %).

Partition statistics					
Popularity rank	1	2	3	4	5
Partition size	756	928	640	832	913
Sample size	325 (43%)	247 (27%)	303 (47%)	268 (32%)	318 (35%)
Annotation cost	2.83	2.17	2.73	2.40	2.78
Veracity estimate	69%±5%	80%±5%	73%±5%	77%±5%	70%±5%

Building on this insight, Figure 2 presents entity-level veracity, defined as the mean of the veracity estimates of the entity facts. The trend observed at the partition level is reaffirmed at the entity level, confirming the orthogonal relationship between veracity and popularity. Indeed, both popular and unpopular entities exhibit a range of high and low veracity scores. The six most popular entities in the collection (left-end of the plot) have the lowest veracity overall, as their facts originate from the most popular yet lowest-quality partition.

Hence, analyzing veracity at the entity level can serve as a valuable quality monitoring system for entity-oriented downstream tasks. Specifically, it can help make informed decisions on corrective actions, such as adjusting the content for popular, low-quality entities or filtering out unpopular, low-quality ones. It becomes especially crucial when dealing with a limited correction budget that requires careful and strategic allocation.

RQ2: Veracity at Scale. We examine the statistics about the KG veracity estimation process. This analysis aims to empirically validate the efficiency of the proposed iterative procedure, verifying that the number of annotations required for each partition is small with respect to the size of the partition.

The outcomes of the iterative procedure used to estimate partition veracity are reported in Table 3. We can see that the number of annotations required for each partition varies depending on the estimated veracity. Specifically, the number ranges from a minimum of 247 annotations for the most reliable partition (popularity rank 2) to a maximum of 325 annotations for the least reliable one (popularity rank 1). This reveals that the iterative procedure demands more annotations to converge as the veracity deviates further from perfection (100%), and the demand increases as it approaches 50%,

[3] https://github.com/KGAccuracyEval/kg-accuracy4entity-search

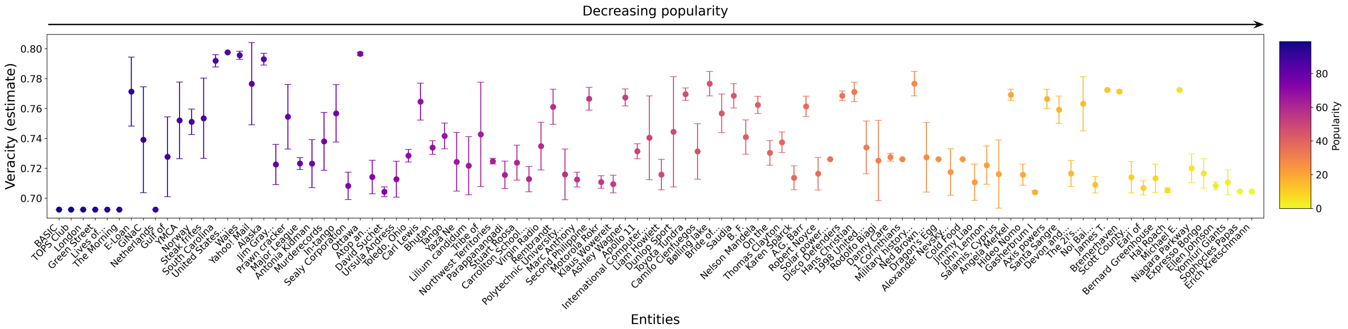

Figure 2: Entity-level veracity. Entities are ordered by decreasing popularity. Veracity is computed by averaging veracity estimates associated with each fact of the target entity. We report mean veracity and its 95% CI for each entity. Popular and unpopular entities exhibit high and low veracity scores, highlighting the orthogonal relationship between veracity and popularity.

representing the point with the highest variance. This behavior confirms the theoretical properties of our approach, which depends exclusively on the (estimated) veracity of the underlying partition, irrespective of its size.

Consequently, this means that the obtained annotation costs represent those that can be obtained on large-scale KGs, thus affirming the efficiency of our devised iterative procedure. In practical terms, all annotation tasks associated with partitions were completed in less than 3 hours, making our solution viable in real-case scenarios. Still, it is important to underline that annotation times are subject to variations based on the level of expertise of the considered annotators, as well as their number.

On a side note, it is worth mentioning that, given the same sample size n, the estimation variance defined in Section 4 is symmetric for $\hat{\mu}_i$ and $1 - \hat{\mu}_i$. This implies that the iterative procedure requires the same number of annotations to converge whether $\hat{\mu}_i$ is, for instance, equal to 0.8 or 0.2. In other words, the costs required to annotate partitions are consistent for both high-quality partitions and their low-quality counterparts.

RQ3: Veracity Impact. We conduct three sets of experiments to assess the veracity influence on entity-oriented search systems.

Fact ranking. The first set of experiments aims to understand how veracity impacts ranking and performance when integrated into the scoring function of ranking methods. To this end, we introduce a straightforward veracity-enhanced re-ranking strategy named vRank. This strategy involves applying MinMax normalization to scores provided by entity summarization methods and then adding the associated veracity estimate to the normalized score, resulting in the veracity-enhanced fact score: $v\text{Score}(t) = \text{nScore}(t) + v(t)$. Here, $\text{nScore}(t)$ represents the normalized score of the fact t and $v(t)$ its veracity estimate. We use the veracity-enhanced fact score to re-rank facts, and we compare the performance of the original methods against the veracity-enhanced re-ranking on nDCG@5 and nDCG@10.

The relationship between the veracity of queries (entities) and the ranking performance (nDCG@10) of both DynES and RELIN is illustrated in Figure 3.

Table 4: Performance of original (orig) and veracity-based, re-ranked (vRank) runs on nDCG@5 and nDCG@10. There is no difference in performance between original and re-ranked runs.

	nDCG@5	nDCG@10
DynES (orig)	0.76	0.79
DynES (vRank)	0.76	0.79
RELIN (orig)	0.46	0.52
RELIN (vRank)	0.46	0.53

The plots reveal no correlation between veracity and relevance. Queries (entities) yielding high and low nDCG@10 scores are evenly distributed across the (estimated) veracity scale for both DynES, a state-of-the-art system for query-dependent entity summarization, and RELIN, a query-agnostic system that relies on the importance of facts for summarization. This underscores the distinct and orthogonal nature of veracity and relevance dimensions.

Motivated by these findings, we explore the impact of veracity on ranking performance when used to re-rank facts, as detailed in Table 4, where we report performance on queries presenting multiple partitions (84/100). Remarkably, the performance of original and re-ranked runs is the same for nDCG@5 and nDCG@10. This suggests that the vRank strategy can boost the quality of the top-ranking positions – by prioritizing facts with higher (estimated) veracity – while maintaining the effectiveness (in terms of relevance) of the original systems unchanged.

To further validate that vRank prioritizes higher-quality facts, we compute Kendall's τ Union (KTU) [42] at cutoffs 5 and 10 between original and re-ranked runs for both DynES and RELIN. Compared to standard Kendall's τ, KTU allows correlations to be computed on rankings with partial overlap, making it suited for the entity cards scenario. The obtained correlation values are 0.68 and 0.65 for DynES at cutoffs 5 and 10, respectively, and 0.68 and 0.55 for RELIN. According to Voorhees [65], correlations lower than 0.8 reflect noticeable changes in rankings, thus confirming the impact of the vRank strategy on ranking.

Entity cards. The following set of experiments investigates the impact veracity has on users' perception of entity cards. To do so, we compare the entity cards of size five obtained via DynES with those obtained via its veracity-enhanced re-ranking (vRank). We choose DynES as reference model since it has been shown to provide better query-dependent entity cards compared to RELIN [31].

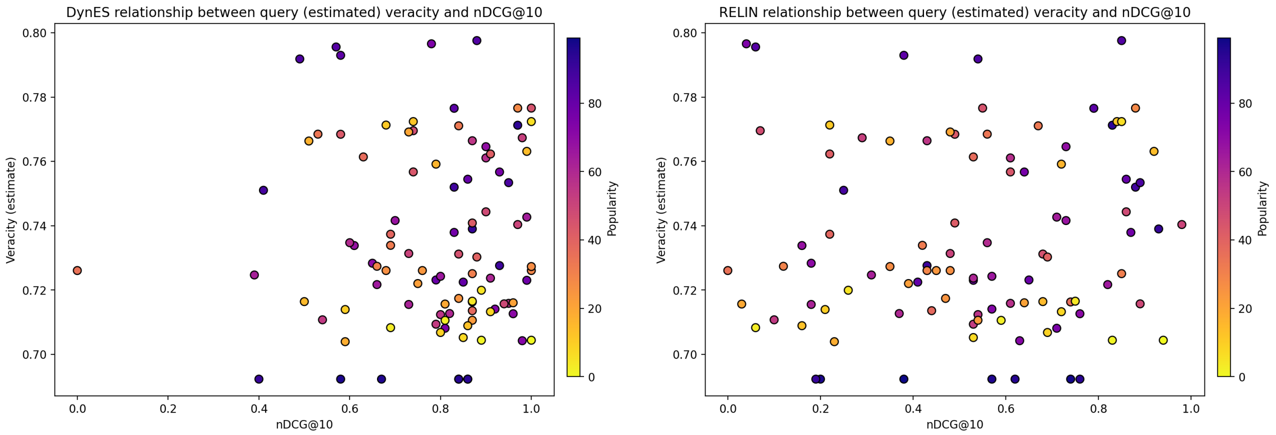

Figure 3: Relationship between query (entity) veracity and nDCG@10 for DynES (left) and RELIN (right). Queries with high and low nDCG@10 scores are equally distributed across the veracity scale for both DynES and RELIN, underscoring the orthogonal nature of veracity and relevance.

The objective of this evaluation is to understand whether users perceive a disparity between entity cards generated solely for relevance (DynES) versus those optimized for both relevance and veracity (vRank). To ensure a meaningful comparison, we restrict the evaluation to those cards whose rankings have a KTU correlation lower than 0.8 – resulting in a subset of 31 entity cards. On these cards, we perform a side-by-side evaluation where five expert annotators are presented with DynES and vRank summaries of the same entity along with the corresponding query.[4] Annotators are instructed to select the preferred summary in relation to the query or indicate a tie if both summaries are equally good. Providing users with a tie option avoids random judgments and facilitates the understanding of user preferences. To avoid bias, the summaries from the two systems are placed on the left or right side randomly. Results are obtained by aggregating user preferences via majority voting.

The results of the cards comparison, quantifying user agreements on win, loss, and tie options, are reported as follows. vRank summaries are deemed superior to DynES for nine entity cards (29%), inferior for seven (23%), and equally good for 15 (48%). Thus, in 77% of cases, the vRank strategy either proves better than DynES or maintains user perception without detriment. This finding further remarks the potential of veracity to enhance user experience by delivering higher-quality content without compromising relevance.

Filtering and correction. The last set of experiments explores the impact of veracity on generating entity cards and influencing ranking performance when quality filtering and correction mechanisms are activated. We consider two sizes for entity cards: 5 and 10. This implies that the corresponding card is not generated if an entity lacks a minimum of 5 or 10 facts. After filtering out facts below a given threshold, we assess the number of cards produced. Additionally, we evaluate post-filtering ranking performance for nDCG@5 and nDCG@10. Following the filtering stage, we apply correction mechanisms and analyze the effects of correcting facts on the number of generated entity cards and the ranking performance.

Table 5: Amount (%) of partition(s) covered by error correction when filtering is applied.

		Partition popularity ranks (and veracity estimates)				
Filtering	Budget	1 (69%)	2 (80%)	3 (73%)	4 (77%)	5 (70%)
$\hat{\mu}_i < 70\%$	1%	41(5%)	0	0	0	0
	5%	203(27%)	0	0	0	0
	10%	407(54%)	0	0	0	0
$\hat{\mu}_i < 75\%$	1%	30(4%)	0	8(1%)	0	3(0.3%)
	5%	149(20%)	0	37(6%)	0	17(2%)
	10%	299(40%)	0	75(12%)	0	33(4%)
$\hat{\mu}_i < 80\%$	1%	29(4%)	0	7(1%)	3(0.4%)	2(0.2%)
	5%	142(19%)	0	36(6%)	16(2%)	9(1%)
	10%	286(38%)	0	71(11%)	32(4%)	18(2%)

As with veracity estimation, annotation remains the go-to solution also for error correction [69]. Indeed, automated methods can give rise to new errors and thereby are often avoided in actual business scenarios [21]. Again, a limited budget is typically allocated for error correction, leaving us with the problem of deciding how to distribute the budget across partitions. To address this, we consider a simple, popularity-based allocation strategy, which distributes the budget across the filtered-out partitions based on the inverse of the squared rank of the partition ranking, induced by the mean popularity of their facts. Since error correction is not the primary focus of this work, we employ a synthetic error correction approach. This entails randomly sampling facts from the filtered-out partitions and marking them as correct. In this way, previously filtered facts re-enter the rankings and contribute to the generation of entity cards. We consider correction budgets of 1%, 5%, and 10% of the collection. We repeat the process 1, 000 times to ensure robust results, reporting mean and standard deviation.

In Table 5, we present the number of correction operations feasible for each partition based on the allocated budget and filtering threshold using the popularity-based allocation strategy. For filtering, we set three quality levels that simulate the minimum standards a search company or KG provider might have for presenting entity cards to users: 70%, 75%, or 80%.

[4] Compared to the extensive veracity annotation process of DBpedia facts, assessing a limited number of entity cards for preference is a more streamlined and cost-effective procedure, enabling us to engage multiple annotators.

Table 6: Number of entity cards with sizes 5 and 10 generated once we remove partitions not meeting the specified quality standards and after performing error correction. Relative improvement (%) over the no-correction scenario is reported next to the results. We also report the number of entity cards generated without filtering (default).

	N. of cards (size=5)			N. of cards (size=10)		
	$\hat{\mu}_i < 70\%$	$\hat{\mu}_i < 75\%$	$\hat{\mu}_i < 80\%$	$\hat{\mu}_i < 70\%$	$\hat{\mu}_i < 75\%$	$\hat{\mu}_i < 80\%$
No correction	93	51	15	75	25	8
1% correction	95.0 ± 0.2(2%)	54.1 ± 0.9(6%)	18.8 ± 1.1(25%)	76.9 ± 0.7(3%)	26.6 ± 0.8(6%)	9.2 ± 0.7(15%)
5% correction	97.9 ± 0.8(5%)	59.8 ± 1.7(17%)	26.7 ± 1.9(78%)	78.9 ± 1.1(5%)	30.9 ± 1.2(24%)	11.7 ± 0.7(46%)
10% correction	99.5 ± 0.5(7%)	66.8 ± 1.9(31%)	37.5 ± 2.3(150%)	82.9 ± 0.9(11%)	36.7 ± 1.6(47%)	15.5 ± 1.4(94%)
No filtering	100			86		

Table 7: Ranking performance for entity cards of size 5 (nDCG@5) and 10 (nDCG@10) generated once we remove partitions not meeting the specified quality standards and after performing error correction. For performance, we omit standard deviation being always < 0.01. Relative improvement (%) over no-correction is reported next to results. For reference, we also report performance obtained without filtering (default).

		nDCG@5			nDCG@10		
		$\hat{\mu}_i < 70\%$	$\hat{\mu}_i < 75\%$	$\hat{\mu}_i < 80\%$	$\hat{\mu}_i < 70\%$	$\hat{\mu}_i < 75\%$	$\hat{\mu}_i < 80\%$
No correction	DynES	0.69	0.46	0.26	0.71	0.41	0.22
	RELIN	0.43	0.36	0.24	0.49	0.34	0.20
1% correction	DynES	0.70(1%)	0.47(2%)	0.28(8%)	0.72(1%)	0.43(5%)	0.23(5%)
	RELIN	0.44(2%)	0.38(6%)	0.25(4%)	0.50(2%)	0.35(3%)	0.21(5%)
5% correction	DynES	0.73(6%)	0.52(13%)	0.35(35%)	0.75(6%)	0.47(15%)	0.29(32%)
	RELIN	0.46(7%)	0.41(14%)	0.30(25%)	0.51(4%)	0.38(12%)	0.26(30%)
10% correction	DynES	0.74(7%)	0.56(22%)	0.41(58%)	0.77(8%)	0.51(24%)	0.34(55%)
	RELIN	0.46(7%)	0.43(19%)	0.35(46%)	0.52(6%)	0.41(21%)	0.30(50%)
No filtering	DynES	0.76			0.79		
	RELIN	0.46			0.52		

Then, Table 6 outlines the count of generated entity cards after filtering out partitions with estimated veracity below 70% (partition 1), 75% (partitions 1, 3, 5), and 80% (partitions 1, 3, 4, 5). It also reports results after error correction based on the given budget and the proposed allocation strategy.

On the one hand, the impact of filtering is evident, leading to a substantial reduction in the number of generated entity cards as the number of filtered partitions increases. On the other hand, the correction mechanism confirms its restorative effect, recovering an increasing number of entity cards with a higher budget. Naturally, the effect is more pronounced on scenarios requiring high-quality levels, potentially doubling the number of generated entity cards when the threshold is set at 80%. Nevertheless, even with a more lenient quality threshold (i.e., 70%), the correction mechanism still manages to impact performance, almost restoring the number of entity cards of size five generated without employing any filtering.

The ranking performance results, presented in Table 7, mirror those observed in entity card generation. Correction mechanisms exhibit the most significant improvements when stringent quality levels are enforced. Yet, when the threshold is set at 70%, both DynES and RELIN almost entirely recover performance levels achieved without filtering for budgets of 5% and 10%.

Thus, veracity significantly impacts entity cards when used as a filter for low-quality data. In this regard, employing budget-constrained error correction mechanisms alongside appropriate allocation strategies can effectively mitigate this impact.

6 CONCLUSIONS AND FUTURE WORK

In this paper, we investigated the impact of KG veracity on entity-oriented search. We reviewed existing literature on entity search and data quality, emphasizing a noticeable lack of interaction between these domains. To fill this gap, we introduced an efficient KG veracity framework, which we used to conduct an exploratory investigation into the impact of veracity on entity cards. The experimental results underscored the efficiency of the proposed framework, highlighting the significant role that veracity plays – serving not only as a monitoring system for entity card generation, but also as an additional dimension in ranking functions. Hence, veracity should be considered a first-class citizen alongside relevance.

Since the efficient, budget-constrained, utility-oriented framework seamlessly lends itself to estimating the veracity of the entire DBpedia (or any other KG), the proposed framework can be naturally extended to entity search on DBpedia [3, 32]. Furthermore, the promising results obtained by using Large Language Models (LLMs) to generate relevance judgments [23, 39, 44] open up new opportunities to combine crowdsourcing and LLMs for KG veracity assessment. Undoubtedly, more research on data veracity for entity-oriented search is warranted in the future, and this paper serves as a foundational starting point for such endeavors.

ACKNOWLEDGMENTS

The work was supported by the HEREDITARY project, as part of the EU Horizon Europe program under Grant Agreement 101137074.

REFERENCES

[1] Z. Abedjan, X. Chu, D. Deng, R. Castro Fernandez, I. F. Ilyas, M. Ouzzani, P. Papotti, M. Stonebraker, and N. Tang. 2016. Detecting Data Errors: Where are we and what needs to be done? *Proc. VLDB Endow.* 9, 12 (2016), 993–1004.

[2] M. Abualsaud, C. Lioma, M. Maistro, M. D. Smucker, and G. Zuccon. 2019. Overview of the TREC 2019 Decision Track. In *Proc. of TREC (NIST Special Publication, Vol. 1250)*. National Institute of Standards and Technology (NIST).

[3] N. Arabzadeh, A. Bigdeli, and E. Bagheri. 2024. LaQuE: Enabling Entity Search at Scale. In *Proc. of the 46th European Conference on Information Retrieval, ECIR 2024, Glasgow, UK, March 24-28, 2024 (Lecture Notes in Computer Science, Vol. 14609)*. Springer, 270–285. https://doi.org/10.1007/978-3-031-56060-6_18

[4] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. G. Ives. 2007. DBpedia: A Nucleus for a Web of Open Data. In *The Semantic Web, 6th International Semantic Web Conference, 2nd Asian Semantic Web Conference, ISWC 2007 + ASWC 2007, Busan, Korea, November 11-15, 2007 (LNCS, Vol. 4825)*. Springer, 722–735. https://doi.org/10.1007/978-3-540-76298-0_52

[5] K. Balog. 2018. *Entity-Oriented Search.* The Information Retrieval Series, Vol. 39. Springer. https://doi.org/10.1007/978-3-319-93935-3

[6] K. Balog and R. Neumayer. 2013. A test collection for entity search in DBpedia. In *Proc. of SIGIR.* ACM, 737–740. https://doi.org/10.1145/2484028.2484165

[7] C. Batini, A. Rula, M. Scannapieco, and G. Viscusi. 2015. From Data Quality to Big Data Quality. *J. Database Manag.* 26, 1 (2015), 60–82. https://doi.org/10.4018/JDM.2015010103

[8] A. Bonifati, G. H. L. Fletcher, H. Voigt, and N. Yakovets. 2018. *Querying Graphs.* Morgan & Claypool Publishers. https://doi.org/10.2200/S00873ED1V01Y201808DTM051

[9] H. S. Bota, K. Zhou, and J. M. Jose. 2016. Playing Your Cards Right: The Effect of Entity Cards on Search Behaviour and Workload. In *Proc. of CHIIR.* ACM, 131–140. https://doi.org/10.1145/2854946.2854967

[10] A.Z. Broder and P. McAfee. 2023. Delphic Costs and Benefits in Web Search: A utilitarian and historical analysis. *CoRR* abs/2308.07525 (2023). https://doi.org/10.48550/ARXIV.2308.07525 arXiv:2308.07525

[11] A. Bronselaer, R. De Mol, and G. De Tré. 2018. A Measure-Theoretic Foundation for Data Quality. *IEEE Trans. Fuzzy Syst.* 26, 2 (2018), 627–639. https://doi.org/10.1109/TFUZZ.2017.2686807

[12] L. Cai and Y. Zhu. 2015. The Challenges of Data Quality and Data Quality Assessment in the Big Data Era. *Data Sci. J.* 14 (2015), 2. https://doi.org/10.5334/DSJ-2015-002

[13] G. Casella and R. L. Berger. 2002. *Statistical Inference.* Thomson Learning.

[14] G. Cheng, T. Tran, and Y. Qu. 2011. RELIN: Relatedness and Informativeness-Based Centrality for Entity Summarization. In *Proc. of ISWC (Lecture Notes in Computer Science, Vol. 7031)*. Springer, 114–129. https://doi.org/10.1007/978-3-642-25073-6_8

[15] C. L. A. Clarke, M. Maistro, and M. D. Smucker. 2021. Overview of the TREC 2021 Health Misinformation Track. In *Proc. of TREC (NIST Special Publication, Vol. 500-335)*. National Institute of Standards and Technology (NIST).

[16] C. L. A. Clarke, S. Rizvi, M. D. Smucker, M. Maistro, and G. Zuccon. 2020. Overview of the TREC 2020 Health Misinformation Track. In *Proc of TREC (NIST Special Publication, Vol. 1266)*. National Institute of Standards and Technology (NIST).

[17] W. G. Cochran. 1977. *Sampling Techniques, 3rd Edition.* John Wiley. https://doi.org/10.1017/S0013091500025724

[18] T. Dalenius and J. L. Hodges. 1959. Minimum Variance Stratification. *J. Amer. Statist. Assoc.* 54, 285 (1959), 88–101. https://doi.org/10.1080/01621459.1959.10501501

[19] O. Deshpande, D. S. Lamba, M. Tourn, S. Das, S. Subramaniam, A. Rajaraman, V. Harinarayan, and A. Doan. 2013. Building, maintaining, and using knowledge bases: a report from the trenches. In *Proc. of SIGMOD.* ACM, 1209–1220. https://doi.org/10.1145/2463676.2465297

[20] X. L. Dong, E. Gabrilovich, K. Murphy, V. Dang, W. Horn, C. Lugaresi, S. Sun, and W. Zhang. 2015. Knowledge-Based Trust: Estimating the Trustworthiness of Web Sources. *Proc. VLDB Endow.* 8, 9 (2015), 938–949.

[21] L. Ehrlinger and W. Wöß. 2022. A Survey of Data Quality Measurement and Monitoring Tools. *Frontiers Big Data* 5 (2022), 850611. https://doi.org/10.3389/FDATA.2022.850611

[22] D. Esteves, A. Rula, A. J. Reddy, and J. Lehmann. 2018. Toward Veracity Assessment in RDF Knowledge Bases: An Exploratory Analysis. *ACM J. Data Inf. Qual.* 9, 3 (2018), 16:1–16:26. https://doi.org/10.1145/3177873

[23] G. Faggioli, L. Dietz, C. L. A. Clarke, G. Demartini, M. Hagen, C. Hauff, N. Kando, E. Kanoulas, M. Potthast, B. Stein, and H. Wachsmuth. 2023. Perspectives on Large Language Models for Relevance Judgment. In *Proc. of the 2023 ACM SIGIR International Conference on Theory of Information Retrieval, ICTIR 2023, Taipei, Taiwan, 23 July 2023.* ACM, 39–50. https://doi.org/10.1145/3578337.3605136

[24] S. Faralli, A. Lenzi, and P. Velardi. 2023. A Benchmark Study on Knowledge Graphs Enrichment and Pruning Methods in the Presence of Noisy Relationships. *J. Artif. Intell. Res.* 78 (2023), 37–68. https://doi.org/10.1613/JAIR.1.14494

[25] J. Gao, X. Li, Y. E. Xu, B. Sisman, X. L. Dong, and J. Yang. 2019. Efficient Knowledge Graph Accuracy Evaluation. *Proc. VLDB Endow.* 12, 11 (2019), 1679–1691.

[26] D. Garigliotti, D. Albakour, M. Martinez, and K. Balog. 2019. Unsupervised Context Retrieval for Long-tail Entities. In *Proc. of ICTIR.* ACM, 225–228. https://doi.org/10.1145/3341981.3344244

[27] A. L. Gînsca, A. Popescu, and M. Lupu. 2015. Credibility in Information Retrieval. *Found. Trends Inf. Retr.* 9, 5 (2015), 355–475. https://doi.org/10.1561/1500000046

[28] L. Goeuriot, H. Suominen, G. Pasi, E. Bassani, N. Brew-Sam, G. N. González Sáez, L. Kelly, P. Mulhem, S. Seneviratne, R. Upadhyay, M. Viviani, and C. Xu. 2021. Consumer Health Search at CLEF eHealth 2021. In *Proc. of the Working Notes of CLEF (CEUR Workshop Proceedings, Vol. 2936)*. CEUR-WS.org, 751–769.

[29] K. Gunaratna, K. Thirunarayan, and A. P. Sheth. 2015. FACES: Diversity-Aware Entity Summarization Using Incremental Hierarchical Conceptual Clustering. In *Proc. of AAAI.* AAAI Press, 116–122. https://doi.org/10.1609/AAAI.V29I1.9180

[30] K. Gunaratna, K. Thirunarayan, A. P. Sheth, and G. Cheng. 2016. Gleaning Types for Literals in RDF Triples with Application to Entity Summarization. In *Proc. of ESWC (Lecture Notes in Computer Science, Vol. 9678)*. Springer, 85–100. https://doi.org/10.1007/978-3-319-34129-3_6

[31] F. Hasibi, K. Balog, and S. E. Bratsberg. 2017. Dynamic Factual Summaries for Entity Cards. In *Proc. of SIGIR.* ACM, 773–782. https://doi.org/10.1145/3077136.3080810

[32] F. Hasibi, F. Nikolaev, C. Xiong, K. Balog, S. E. Bratsberg, A. Kotov, and J. Callan. 2017. DBpedia-Entity v2: A Test Collection for Entity Search. In *Proc. of SIGIR.* ACM, 1265–1268. https://doi.org/10.1145/3077136.3080751

[33] J. Hoffart, F. M. Suchanek, K. Berberich, and G. Weikum. 2013. YAGO2: A spatially and temporally enhanced knowledge base from Wikipedia. *Artif. Intell.* 194 (2013), 28–61. https://doi.org/10.1016/j.artint.2012.06.001

[34] F. Ilievski, P. Vossen, and S. Schlobach. 2018. Systematic Study of Long Tail Phenomena in Entity Linking. In *Proc. of COLING.* Association for Computational Linguistics, 664–674. https://aclanthology.org/C18-1056/

[35] Jimmy, G. Zuccon, B. Koopman, and G. Demartini. 2019. Health Cards to Assist Decision Making in Consumer Health Search. In *Proc. of AMIA.* AMIA. https://knowledge.amia.org/69862-amia-1.4570936/t005-1.4574828/t005-1.4574829/3201885-1.4574890/3201686-1.4574887

[36] Jimmy, G. Zuccon, J. R. M. Palotti, L. Goeuriot, and L. Kelly. 2018. Overview of the CLEF 2018 Consumer Health Search Task. In *Working Notes of CLEF (CEUR Workshop Proceedings, Vol. 2125)*. CEUR-WS.org.

[37] D. Lagun, C. H. Hsieh, D. Webster, and V. Navalpakkam. 2014. Towards better measurement of attention and satisfaction in mobile search. In *Proc. of SIGIR.* ACM, 113–122. https://doi.org/10.1145/2600428.2609631

[38] Q. Liu, G. Cheng, K. Gunaratna, and Y. Qu. 2021. Entity summarization: State of the art and future challenges. *J. Web Semant.* 69 (2021), 100647. https://doi.org/10.1016/J.WEBSEM.2021.100647

[39] S. MacAvaney and L. Soldaini. 2023. One-Shot Labeling for Automatic Relevance Estimation. In *Proc. of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2023, Taipei, Taiwan, July 23-27, 2023.* ACM, 2230–2235. https://doi.org/10.1145/3539618.3592032

[40] S. Madnick and R. Y. Wang. 1992. Introduction to total data quality management (TDQM) research program. *Total Data Qual. Manag. Program MIT Sloan Sch. Manag* 1 (1992), 92.

[41] S. E. Madnick, R. Y. Wang, Y. W. Lee, and H. Zhu. 2009. Overview and Framework for Data and Information Quality Research. *ACM J. Data Inf. Qual.* 1, 1 (2009), 2:1–2:22. https://doi.org/10.1145/1515693.1516680

[42] M. Maistro, T. Breuer, P. Schaer, and N. Ferro. 2023. An in-depth investigation on the behavior of measures to quantify reproducibility. *Inf. Process. Manag.* 60, 3 (2023), 103332. https://doi.org/10.1016/J.IPM.2023.103332

[43] S. Marchesin and G. Silvello. 2024. Efficient and Reliable Estimation of Knowledge Graph Accuracy. *Proc. VLDB Endow.* 17, 9 (2024), 2392–2404. https://doi.org/10.14778/3665844.3665865

[44] C. Meng, N. Arabzadeh, A. Askari, M. Aliannejadi, and M. de Rijke. 2024. Query Performance Prediction using Relevance Judgments Generated by Large Language Models. arXiv:2404.01012 [cs.IR]

[45] T. M. Mitchell, W. W. Cohen, E. R. Hruschka Jr., P. P. Talukdar, B. Yang, J. Betteridge, A. Carlson, B. D. Mishra, M. Gardner, B. Kisiel, J. Krishnamurthy, N. Lao, K. Mazaitis, T. Mohamed, N. Nakashole, E. A. Platanios, A. Ritter, M. Samadi, B. Settles, R. C. Wang, D. Wijaya, A. Gupta, X. Chen, A. Saparov, M. Greaves, and J. Welling. 2018. Never-ending learning. *Commun. ACM* 61, 5 (2018), 103–115. https://doi.org/10.1145/3191513

[46] V. Navalpakkam, L. Jentzsch, R. Sayres, S. Ravi, A. Ahmed, and A. J. Smola. 2013. Measurement and modeling of eye-mouse behavior in the presence of nonlinear page layouts. In *Proc. of WWW.* International World Wide Web Conferences Steering Committee / ACM, 953–964. https://doi.org/10.1145/2488388.2488471

[47] P. Ojha and P. P. Talukdar. 2017. KGEval: Accuracy Estimation of Automatically Constructed Knowledge Graphs. In *Proc. of EMNLP.* ACL, 1741–1750. https://doi.org/10.18653/v1/d17-1183

[48] J. R. M. Palotti, G. Zuccon, Jimmy, P. Pecina, M. Lupu, L. Goeuriot, L. Kelly, and A. Hanbury. 2017. CLEF 2017 Task Overview: The IR Task at the eHealth Evaluation Lab - Evaluating Retrieval Methods for Consumer Health Search. In *Working Notes of CLEF (CEUR Workshop Proceedings, Vol. 1866)*. CEUR-WS.org.

[49] H. Paulheim. 2017. Knowledge Graph Refinement: A Survey of Approaches and Evaluation Methods. *Semantic Web* 8, 3 (2017), 489–508. https://doi.org/10.3233/

SW-160218

[50] M. Petrocchi and M. Viviani. 2022. ROMCIR 2022: Overview of the 2nd Workshop on Reducing Online Misinformation Through Credible Information Retrieval. In *Proc. of ECIR (Lecture Notes in Computer Science, Vol. 13186)*. Springer, 566–571. https://doi.org/10.1007/978-3-030-99739-7_71

[51] M. Petrocchi and M. Viviani. 2023. ROMCIR 2023: Overview of the 3rd Workshop on Reducing Online Misinformation Through Credible Information Retrieval. In *Proc. of ECIR (Lecture Notes in Computer Science, Vol. 13982)*. Springer, 405–411. https://doi.org/10.1007/978-3-031-28241-6_45

[52] M. Petrocchi and M. Viviani. 2024. ROMCIR 2024: Overview of the 4th Workshop on Reducing Online Misinformation Through Credible Information Retrieval. In *Proc. of ECIR (Lecture Notes in Computer Science, Vol. 14612)*. Springer, 403–408. https://doi.org/10.1007/978-3-031-56069-9_54

[53] J. Pujara, E. Augustine, and L. Getoor. 2017. Sparsity and Noise: Where Knowledge Graph Embeddings Fall Short. In *Proc. of EMNLP*. ACL, 1751–1756. https://doi.org/10.18653/v1/d17-1184

[54] Y. Qi, W. Zheng, L. Hong, and L. Zou. 2022. Evaluating Knowledge Graph Accuracy Powered by Optimized Human-Machine Collaboration. In *Proc. of SIGKDD*. ACM, 1368–1378. https://doi.org/10.1145/3534678.3539233

[55] R. Reinanda, E. Meij, and M. de Rijke. 2020. Knowledge Graphs: An Information Retrieval Perspective. *Found. Trends Inf. Retr.* 14, 4 (2020), 289–444. https://doi.org/10.1561/1500000063

[56] S. W. Sadiq, N. K. Yeganeh, and M. Indulska. 2011. 20 Years of Data Quality Research: Themes, Trends and Synergies. In *Proc. of ADC (CRPIT, Vol. 115)*. Australian Computer Society, 153–162. http://crpit.scem.westernsydney.edu.au/abstracts/CRPITV115Sadiq.html

[57] S. Salimzadeh, D. Maxwell, and C. Hauff. 2021. The Impact of Entity Cards on Learning-Oriented Search Tasks. In *Proc. of ICTIR*. ACM, 63–72. https://doi.org/10.1145/3471158.3472255

[58] M. Samadi, P. P. Talukdar, M. M. Veloso, and T. M. Mitchell. 2015. AskWorld: Budget-Sensitive Query Evaluation for Knowledge-on-Demand. In *Proc. of IJCAI*. AAAI Press, 837–843. http://ijcai.org/Abstract/15/123

[59] F. Saracco and M. Viviani. 2021. ROMCIR 2021: Reducing Online Misinformation through Credible Information Retrieval. In *Proc. of ECIR (Lecture Notes in Computer Science, Vol. 12657)*. Springer, 714–717. https://doi.org/10.1007/978-3-030-72240-1_87

[60] M. Shokouhi and Q. Guo. 2015. From Queries to Cards: Re-ranking Proactive Card Recommendations Based on Reactive Search History. In *Proc. of SIGIR*. ACM, 695–704. https://doi.org/10.1145/2766462.2767705

[61] M. Sinha, S. Mannarswamy, and S. Roy. 2016. CHIS@FIRE: Overview of the Shared Task on Consumer Health Information Search. In *Working notes of FIRE 2016 - Forum for Information Retrieval Evaluation, Kolkata, India, December 7-10, 2016 (CEUR Workshop Proceedings, Vol. 1737)*. CEUR-WS.org, 193–196.

[62] I. Taleb, M. A. Serhani, and R. Dssouli. 2018. Big Data Quality: A Survey. In *2018 IEEE International Congress on Big Data, BigData Congress 2018*. IEEE Computer Society, 166–173. https://doi.org/10.1109/BIGDATACONGRESS.2018.00029

[63] A. Thalhammer, N. Lasierra, and A. Rettinger. 2016. LinkSUM: Using Link Analysis to Summarize Entity Data. In *Proc. of ICWE (Lecture Notes in Computer Science, Vol. 9671)*. Springer, 244–261. https://doi.org/10.1007/978-3-319-38791-8_14

[64] A. Thalhammer and A. Rettinger. 2014. Browsing DBpedia Entities with Summaries. In *Proc. of ESWC Satellite Events (Lecture Notes in Computer Science, Vol. 8798)*. Springer, 511–515. https://doi.org/10.1007/978-3-319-11955-7_76

[65] E. M. Voorhees. 2001. Evaluation by Highly Relevant Documents. In *SIGIR 2001: Proc. of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, September 9-13, 2001, New Orleans, Louisiana, USA*. ACM, 74–82. https://doi.org/10.1145/383952.383963

[66] D. Vrandecic and M. Krötzsch. 2014. Wikidata: a free collaborative knowledgebase. *Commun. ACM* 57, 10 (2014), 78–85. https://doi.org/10.1145/2629489

[67] C. Wardle and H. Derakhshan. 2017. Information disorder: Toward an interdisciplinary framework for research and policy making (2017). (2017).

[68] G. Weikum, X. L. Dong, S. Razniewski, and F. M. Suchanek. 2021. Machine Knowledge: Creation and Curation of Comprehensive Knowledge Bases. *Found. Trends Databases* 10, 2-4 (2021), 108–490. https://doi.org/10.1561/1900000064

[69] B. Xue and L. Zou. 2023. Knowledge Graph Quality Management: A Comprehensive Survey. *IEEE Trans. Knowl. Data Eng.* 35, 5 (2023), 4969–4988. https://doi.org/10.1109/TKDE.2022.3150080

[70] A. Zaveri, A. Rula, A. Maurino, R. Pietrobon, J. Lehmann, and S. Auer. 2016. Quality assessment for Linked Data: A Survey. *Semantic Web* 7, 1 (2016), 63–93. https://doi.org/10.3233/SW-150175

[71] L. Zheng, P. Cheng, L. Chen, J. Yu, X. Lin, and J. Yin. 2022. Crowdsourced Fact Validation for Knowledge Bases. In *Proc. of ICDE*. IEEE, 938–950. https://doi.org/10.1109/ICDE53745.2022.00075