Outdated Fact Detection in Knowledge Bases

Shuang Hao[†] Chengliang Chai[‡] Guoliang Li[‡] Nan Tang[§] Ning Wang[†] Xiang Yu[‡]
[†]Beijing Key Lab of Traffic Data Analysis and Mining, School of Computer and Information Technology, Beijing Jiaotong University, China
[‡]Department of Computer Science and Technology, Tsinghua University, China [§]Qatar Computing Research Institute, HBKU, Qatar
{haoshuang@, nwang@}bjtu.edu.cn, {chaicl15@mails., liguoliang@, x-yu17@mails.}tsinghua.edu.cn, ntang@hbku.edu.qa

***Abstract*—Knowledge bases (KBs), which store high-quality information, are crucial for many applications, such as enhancing search results and serving as external sources for data cleaning. Not surprisingly, there exist outdated facts in most KBs due to the rapid change of information. Naturally, it is important to keep KBs up-to-date. Traditional wisdom has investigated the problem of using reference data (such as new facts extracted from the news) to detect outdated facts in KBs. However, existing approaches can only cover a small percentage of facts in KBs. In this paper, we propose a novel human-in-the-loop approach for outdated fact detection in KBs. It trains a binary classifier using features such as historical update frequency and existence time of a fact to compute the likelihood of a fact in a KB to be outdated. Then, it interacts with humans to verify whether a fact with high likelihood is indeed outdated. In addition, it also uses logical rules to detect more outdated facts based on human feedback. The outdated facts detected by the logical rules will also be fed back to train the ML model further for *data augmentation*. Extensive experiments on real-world KBs, such as Yago and DBpedia, show the effectiveness of our solution.**

I. Introduction

Knowledge bases (KBs), such as Yago [1] and DBpedia [2], provide rich structured information of real-world entities and their relations, which can greatly increase the performance of information extraction [3], question answering [4] and data cleaning [5], [6]. However, the facts in KBs may become out-of-date as the world changes, which will limit the utility of KBs. Thus, knowledge in KBs should be viewed as a rapidly evolving set of facts that must update as the world changes.

A commonly used method to detect outdated facts is to extract some up-to-date facts as reference data from news, media texts and encyclopedia websites [7], [8], [9]. These update-to-date facts can be used to detect those outdated ones by comparing with the facts in a KB. However, purely relying on the reference data is not enough because they can only cover a small proportion of entities and relations in a KB. Naturally, one interesting question is that whether we can generalize the information based on knowing a fact to be up-to-date or outdated.

Logical rules are widely used to generalize the information at hand. For example, given a logical rule φ: "*If the current team of x is y and so is z, then x and z are teammates*", and the fact $\langle$*Stephen Curry*, *teammate*, *Kevin Durant*$\rangle$ has confirmed to be outdated, we can deduce that either *Stephen Curry* or *Kevin Durant* has left their team. Certainly, there is another way to generalize: training a machine learning (ML) model. In addition, in order to tackle the low coverage of external sources, an intuitive strategy is to involve the user so as to collect more evidence.

There are two main challenges to tackle. **(C1)** It is non-trivial to build a well-performed model for outdated facts detection. **(C2)** Question selection for user needs to consider two criteria: informative and diversified, so as to maximize the number of outdated facts that can be detected. To tackle **C1**, we map the up-to-date reference data to the facts in KBs to generate the positive and negative examples for the classifier training. Features are extracted from the KB revision history, such as the historical update frequency and existence time of a fact. To tackle **C2**, we propose to jointly use the ML model and rule-based method to reduce human cost. We interact with a user to label k selected facts so as to maximize the number of detected outdated facts, which are then used to "augment" more outdated facts for training.

Contributions. We make the following notable contributions. We propose a novel human-in-the-loop framework to detect the outdated facts in KBs. We build an iterative classification model to predict the outdated facts. We judiciously select questions iteratively for a user to verify the candidate outdated facts. We conducted experiments on real KBs to show the effectiveness and efficiency of our framework.

II. Problem Formulation

Outdated Fact. Considering a set of facts $\mathcal{K} = \{\langle s, r, o\rangle\}$ in a *knowledge base* (KB), each *fact* is a triple where s is an entity, o is either an entity or a literal, r is a relation or property between s and o. A fact $\langle s, r, o\rangle$ in $\mathcal{K}$ is an *outdated fact* if it is not synchronous with the changes of the real world.

Human Intelligence Task. The *human intelligence task* (HIT) in this paper is specifically referred to the task for a fact that asks human to indicate whether it is out-of-date or not. For example, two HITs are shown in Figure 1.

Logical Rule. We follow the definition of logical rule in [10]: A *logical rule* φ defined in a KB is formalized as $\mathcal{A} \Rightarrow \mathcal{H}$ where (1) the body $\mathcal{A}$ is a conjunction of atoms $a_1 \wedge a_2 \wedge ... \wedge a_n$. Each atom $a_i \in \mathcal{A}$ corresponds to a triple that has variables at the subject and object position and with a fixed relation or property. (2) the head $\mathcal{H}$ denotes a single atom. An *instantiation* of a rule is the one where all variables have been substituted by entities in a KB.

2375-026X/20/$31.00 ©2020 IEEE
DOI 10.1109/ICDE48307.2020.00196

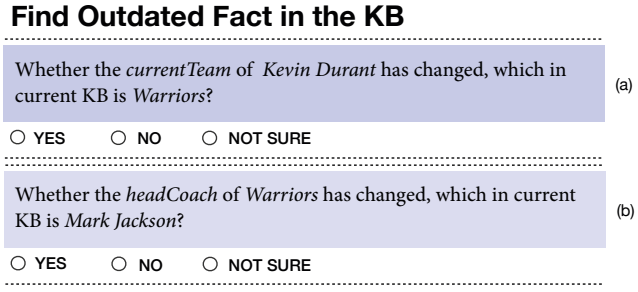


Fig. 1. Human Intelligence Task (HIT)

Semantics. Given an instantiation of rule φ, it has the following properties that can help us to infer more outdated facts. (1) $\forall a_i \in \mathcal{A}$, if the instantiations of them are up-to-date, we can conclude that the instantiation of $\mathcal{H}$ is up-to-date. (2) Based on the inverse and negative proposition, we can conclude that if the instantiation of $\mathcal{H}$ is outdated, $\exists a_i \in A$, the instantiation of a_i is outdated.

Problem Statement. Given a KB $\mathcal{K}$, a set of reference facts $\mathcal{T}$, and a budget B that determines the maximum number of HITs answered by human, this paper aims to detect the maximum number of outdated facts in $\mathcal{K}$.

III. Framework Overview

We introduce the framework as depicted in Figure 2, which contains three phases: (I) Outdated fact prediction, which aims at predicting outdated facts using an iterative classifier model. The output of the model is the likelihood of each fact being outdated (Section III-A); (II) Human-based verification, which is to verify the outdated facts detected by the ML model of phase I. Given a budget B, our goal is to detect as many outdated facts as possible within B (Section III-B); (III) Rule-based fact expansion, which aims to expand the human answers to detect more outdated facts. It takes the human labels from phase II as input, infers more outdated facts and feed these inferred facts into the ML model in phase I (Section III-C).

A. Outdated Fact Prediction Model

To start training the model, we need some training examples. Given an up-to-date fact $\langle s, r, o\rangle \in \mathcal{T}$, we can find out the corresponding outdated facts and up-to-date facts from the current KB $\mathcal{K}$ as positive and negative examples respectively. Concretely, if $\langle s, r, o\rangle$ has already in $\mathcal{K}$, it can be confirmed as an up-to-date fact. Otherwise, we check three kinds of triples in $\mathcal{K}$ whether they are outdated or not: $\langle s, r, *\rangle$, $\langle *, r, o\rangle$ and $\langle s, *, o\rangle$, where '$*$' is a wildcard. For the facts with pattern $\langle s, r, *\rangle$ or $\langle *, r, o\rangle$ in $\mathcal{K}$, it depends on the correspondence of relation r to determine whether they are out-of-date. For example, if r is a one-to-many relation, which means that one subjective entity may be linked to many objective entities but one objective entity is linked to only one subjective entity, $\langle *, r, o\rangle$ is out-of-date. For the facts with pattern $\langle s, *, o\rangle$, we should find whether there exists a relation r' that is incompatible with r in $\mathcal{K}$, which means that the relation between two entities in $\mathcal{K}$ cannot be both r and r', *i.e.,* $\langle s, r, o\rangle$ and $\langle s, r', o\rangle$ cannot co-exist in $\mathcal{K}$ at the same time no matter what these two entities s and o are. If so, $\langle s, r', o\rangle$ is an outdated fact.

Given the training data and ML model which is treated as a black box, we have to do subtle feature engineering customized to the outdated fact detection problem as follows. For each fact $f : \langle s, r, o\rangle$ in the training example, we generate its features based on the current and previous versions of KBs. The first three features are used to determine whether the entity s is an active entity, *i.e.,* whether the properties of s will change. The fourth feature is to determine whether the relation r is an active relation and the last is for the entire fact. The features are listed as below:

- *Completeness of entity s.* It is the ratio of the number of properties of entity s to the maximum number of properties among the entities with the same type.
- *Historical update frequency of entity s.* It measures how many times entity s has been updated after being added to the KB.
- *#Links from other entites to entity s.* It measures the number of entities that direct to s. More specifically, s' directs to s if there exists a triple $\langle s', r, s\rangle$.
- *Historical update frequency of* $\langle s, r, *\rangle$. It measures how many times the object of relation r with entity s has been updated after being added to the KB, which can reflect the stability of relation r.
- *Time of existence of fact f.* This is used to quantify how long fact f exists in the KB.

Each training example will be constructed as a labelled data $\langle\langle x_1(f), x_2(f), x_3(f), x_4(f), x_5(f)\rangle, y(f)\rangle$, where each $x_i(f)$ is one of the feature about fact f. If f is outdated, $y(f) = 1$. Otherwise, $y(f) = 0$. These labelled training data are further fed into a classifier to predict the probability of the fact changes. The classifier will return a value in [0, 1] for each triple as the likelihood of being outdated.

B. HITs Selection for Human Verification

To guarantee the high quality of outdated facts detection, we aim to leverage the users to verify the facts predicted by the ML model through HITs. The biggest challenge is which k facts should be asked in each iteration because it directly determines the number of outdated facts detected, which consist of the outdated facts verified by the users directly and inferred by the rules. We build a graph model for this problem.

Graph Model. A directed graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ is built as below. Each vertex $v \in \mathcal{V}$ is a fact in $\mathcal{K}$ associated with a weight $w(v)$ which is the probability of v to be outdated. A directed edge from vertex u to vertex v indicates that u and v exist in the right and left hand of an instantiation of a rule φ respectively, which means that if u is out-of-date, it can be inferred that v is outdated (when rule φ has two atoms) or v is very likely to be outdated (when rule φ has more than two atoms).

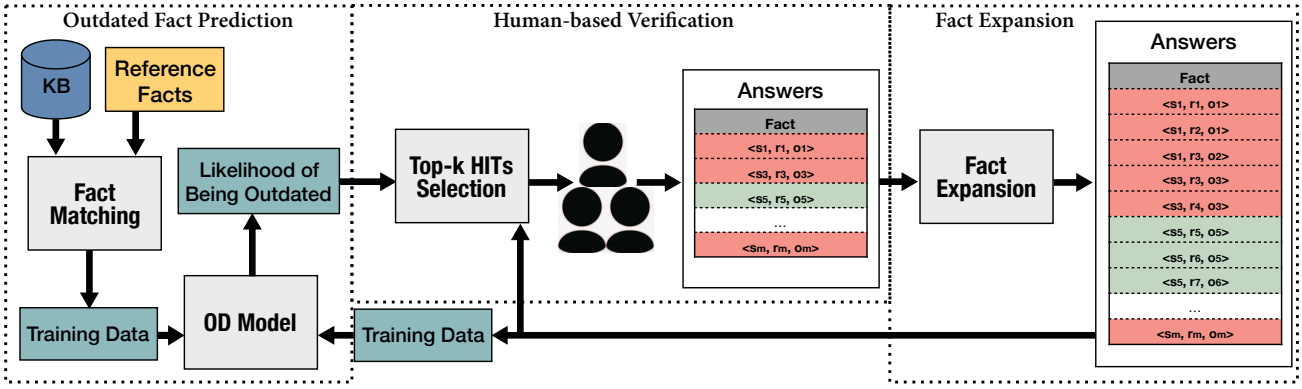


Fig. 2. Framework Overview

Independent Set, K-size Independent Set, Maximum Weighted K-size Independent Set. A subset $\mathcal{I} \subseteq \mathcal{V}$ is an *independent set*, if for any two vertices u, v in $\mathcal{I}$, there is no edge connecting them. An independent set containing exactly k vertices is a *k-size independent set*. The *maximum weighted k-size independent set problem* is finding a k-size independent set $\widehat{\mathcal{I}}^k$ with maximum total vertex weight in all independent sets of size k.

Problem Formulation. The top-k best HITs selection problem can be formulated as follows.

$$\begin{aligned} &\texttt{max} \sum_{v \in \mathcal{H}^k} \alpha \times w(v) + \beta \times \tilde{d}_o(v) \\ &\texttt{s.t.}\ \nexists (u,v) \in \mathcal{E}, u \in \mathcal{H}^k, v \in \mathcal{H}^k \end{aligned} \tag{1}$$

(1) $w(v)$ denotes the weight of vertex v which is the probability of v to be outdated. Maximizing the total vertex weight can maximize the number of outdated facts in k HITs. (2) $\tilde{d}_o(v)$ denotes the normalized out-degree of vertex v which is $\frac{d_o(v)-d_o^{min}}{d_o^{max}-d_o^{min}}$ where $d_o(v)$, d_o^{min} and d_o^{max} are the out-degree of vertex v, the smallest out-degree in $\mathcal{G}$, and the largest out-degree in $\mathcal{G}$ respectively. Maximizing the total vertex out-degree can maximize the number of outdated facts can be inferred. (3) The above two goals "maximize the number of outdated facts confirmed by users in k HITs" and "maximize the number of outdated facts inferred by rules" may not be reached at the same time. Thus, $\alpha \in [0,1]$ and $\beta \in [0,1]$ are used to balance the weights of these two goals and $\alpha + \beta = 1$. (4) The constraint means that $\mathcal{H}^k$ must be an independent set. The reason is that if asking u can deduce the answer of v, it is wasteful to ask both vertices.

We can prove that the top-k best HITs selection problem is NP-complete by a reduction from the 3SAT problem. In the following paragraphs, we use $w'(v) = \alpha \times w(v) + \beta \times \tilde{d}_o(v)$ as the weight of vertex v. Then selecting the top-k best HITs is equivalent to finding the maximum weighted k-size independent set $\widehat{\mathcal{I}}^k$ in the graph model.

Algorithms to Find $\mathcal{H}^k$. We can simply enumerate all k-size independent sets and find the one with the largest weight as $\mathcal{H}^k$. However, it is time-consuming and an efficient greedy algorithm is needed. The traditional greedy strategy is to choose the vertex with higher weight and smaller degree to maximize the total vertex weight of an independent set. But in our problem, minimizing the degree of a vertex will lead to a reduction in the number of outdated fact can be inferred. Thus, we prefer the vertex v which can maximize the score function $g(v) = \frac{w'(v)}{1+d_i(v)}$, where $d_i(v)$ is the in-degree of vertex v. It is meaningful because the independent set generated in this way can reach k-size as fast as possible, and contains the vertices that are difficult to be referred by other vertices. Note that if there is no k-size independent set in $\mathcal{G}$, we can first greedily find a maximal independent set $\mathcal{I}^*$ then add $k - |\mathcal{I}^*|$ vertices into it based on their priorities to obtain $\mathcal{H}^k$.

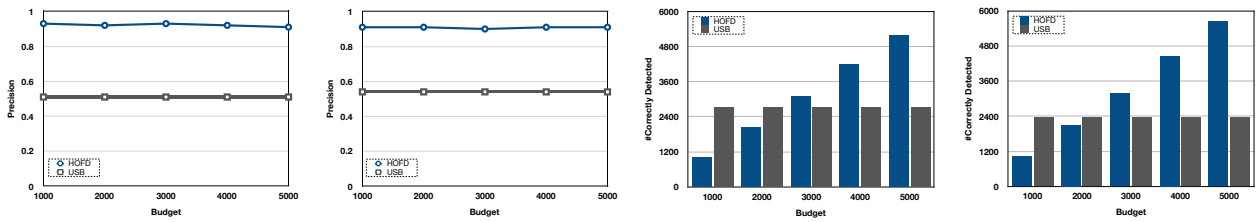
C. Fact Expansion

After collecting user answers, all verified vertices and the corresponding edges should be removed from the graph model. Meanwhile, we should expand more outdated facts based on the semantics of logical rules. That is, given an instantiation $v \Rightarrow u$ of a two-atom rule, where $v, u \in \mathcal{V}$ and $(u, v) \in \mathcal{E}$, if u is outdated, we can deduce that v is also outdated and safely remove vertex v from $\mathcal{G}$. Meanwhile, if v is confirmed to be up-to-date by the user, we can also infer that u is up-to-date and also remove vertex u from $\mathcal{G}$.

However, consider a three-atom rule's instantiation $v \wedge y \Rightarrow u$ where $v, y, u \in \mathcal{V}$ and $(u, v), (u, y) \in \mathcal{E}$, even if u is outdated, it is still hard to determine which of v and y is outdated. At this time, we cannot remove v or y from $\mathcal{G}$ but just update the weights of these two vertices. Since u is confirmed to be outdated, *i.e.,* $w''(u) = 1$, we can infer that either v or y is not up-to-date. Since $w''(v) : w''(y) = w'(v) : w'(y)$ could be held, these two pieces of information can be combined to get the values of $w''(v)$ and $w''(y)$.

IV. Experiments

Experimental Setup. We used two real-world KBs to evaluate our method. (1) `Yago` [1]. We detected the outdated facts in Yago-1.1.0 (released in 2009-10) and Yago-2.0.0



(a) Yago (Precision) (b) DBpedia (Precision) (c) Yago (#Correctly Detected) (d) DBpedia (#Correctly Detected)

Fig. 3. Overall Evaluation of the Framework

(released in 2010-08) was considered as the ground truth. (2) DBpedia [11]. We detected the outdated facts in DBpedia-3.4 (released in 2009-09) and took DBpedia-3.6 (released in 2010-10) as the ground truth. We chose the above versions of KBs since we can only obtain the news data from the Gigaword v.5 dataset[1] for that time period as the reference data.

We recruited 10 students to take part in the evaluation, and all participants had necessary skills to answer the questions, but none of them was expert in outdated fact detection. Before starting the evaluation, we generated some HITs for qualification test. The correct rate of each participant was recorded for further use in answer aggregation *i.e.,* weighted majority voting.

Experimental Results. We compared the effectiveness of our framework with USB [9] which can extract hot entities from the reference data and then crawl their encyclopedia websites to obtain the latest facts for outdated fact detection. Since it was difficult to get the contents of Wikipedia pages from 2009 to 2010, we can only simulate the process of USB as follows: USB first extracted hot entities from the reference data. Then a model was built to detect some active entities linked with those hot ones in KBs. Finally it regarded the facts of those hot entities as well as linked entities as the outdated facts.

In this experiments, we took precision and the number of outdated facts correctly detected as evaluation metrics. Here, the precision is the ratio of outdated facts we correctly detected to the number of all outdated facts confirmed by users and deduced by rules. We changed the value of budget B from 1000 to 5000 and fixed the number of HITs in each iteration to 100. The experimental results are presented in Figure 3.

Figure 3(a)-3(b) show that the precision of our algorithm in two datasets was higher than 0.9, which means that most of the outdated facts confirmed by users and inferred by logical rules were truly out-of-date. The precision of USB did not exceed 0.6 since the facts of entities that appeared in the news and their linked entities did not necessarily change. For example, consider the fact ⟨Obama, livesIn, Washington⟩ in the reference data, USB also regarded Washington as hot entities but all facts of Washington (Washington is the subjective entity) did not changed. In Figure 3(c)-3(d), we can see that as the rise of budget, the number of outdated facts we correctly detected also increased. When the budget reached 3000, it had exceeded the number of outdated facts detected by USB, which borne out the claim that purely relying on the reference data is not enough because the news can only cover a small proportion of knowledge in KBs.

[1] https://catalog.ldc.upenn.edu/LDC2011T07

V. Conclusion

In this paper, we have proposed a novel framework to detect the outdated facts in the KBs given a set of latest reference facts. We have built an iterative outdated fact prediction model to provide the likelihood of each fact being outdated. We have judiciously selected questions iteratively for the user to verify the candidate outdated facts. We have also utilized the logical rules to deduce more outdated facts. We have demonstrated the effectiveness of our approaches on real KBs.

Acknowledgement

This work was supported by the Fundamental Research Funds for the Central Universities (2019RC015).

References

[1] J. Hoffart, F. M. Suchanek, K. Berberich, and G. Weikum, "Yago2: A spatially and temporally enhanced knowledge base from wikipedia," *Artificial Intelligence*, vol. 194, pp. 28–61, 2013.

[2] M. Morsey, J. Lehmann, S. Auer, and A.-C. N. Ngomo, "Dbpedia sparql benchmark–performance assessment with real queries on real data," in *International semantic web conference*. Springer, 2011, pp. 454–469.

[3] R. Hoffmann, C. Zhang, X. Ling, L. Zettlemoyer, and D. S. Weld, "Knowledge-based weak supervision for information extraction of overlapping relations," in *ACL*, 2011.

[4] Y. Kiyota, S. Kurohashi, and F. Kido, "Dialog navigator: A question answering system based on large text knowledge base," in *COLING*, 2002.

[5] X. Chu, J. Morcos, I. F. Ilyas, M. Ouzzani, P. Papotti, N. Tang, and Y. Ye, "Katara: A data cleaning system powered by knowledge bases and crowdsourcing," in *SIGMOD*, 2015.

[6] S. Hao, N. Tang, G. Li, and J. Li, "Cleaning relations using knowledge bases," in *ICDE*, 2017.

[7] K. Leetaru and P. A. Schrodt, "Gdelt: Global data on events, location, and tone, 1979–2012," in *ISA annual convention*, vol. 2, no. 4. Citeseer, 2013, pp. 1–49.

[8] M. Morsey, J. Lehmann, S. Auer, C. Stadler, and S. Hellmann, "Dbpedia and the live extraction of structured data from wikipedia," *Program*, vol. 46, no. 2, pp. 157–181, 2012.

[9] J. Liang, S. Zhang, and Y. Xiao, "How to keep a knowledge base synchronized with its encyclopedia source," in *IJCAI*, 2017.

[10] L. A. Galárraga, C. Teflioudi, K. Hose, and F. Suchanek, "Amie: association rule mining under incomplete evidence in ontological knowledge bases," in *WWW*, 2013.

[11] J. Lehmann, R. Isele, M. Jakob, A. Jentzsch, D. Kontokostas, P. N. Mendes, S. Hellmann, M. Morsey, P. Van Kleef, S. Auer *et al.*, "Dbpedia–a large-scale, multilingual knowledge base extracted from wikipedia," *Semantic Web*, vol. 6, no. 2, pp. 167–195, 2015.