ATLANTIS PRESS
*Atlantis Studies in Uncertainty Modelling, volume 3*
**Joint Proceedings of the 19th World Congress of the International Fuzzy Systems Association (IFSA), the 12th Conference of the European Society for Fuzzy Logic and Technology (EUSFLAT), and the 11th International Summer School on Aggregation Operators (AGOP)**

# Data Fusion in Question Answering Systems over Multiple-Knowledge Bases

**Nhuan D. To**[a],[b] and * **Marek Z. Reformat**[a,c]
[a]Electrical and Computer Engineering, University of Alberta, Edmonton, Alberta, Canada
{nhuan,reformat}@ualberta.ca
[b]Nam Dinh University of Technology Education, Vietnam
[c]Information Technology Institute, University of Social Sciences, 90-113 Łódź, Poland

## Abstract

The additional ability to span over multiple Knowledge Bases would further increase the usefulness and potentially comprehensiveness of the system's responses. Multiple different Knowledge Bases, as much as they can complement each other regarding the lack of specific information, quite often contain inconsistent pieces of information. This makes data fusing a difficult task.

In this paper, we propose a methodology for fusing data retrieved from multiple different Knowledge Bases that use different naming schemas. It also contains a procedure for determining degrees of trustworthiness in the Bases. These degrees are included in the data fusion process.

**Keywords:** Data fusion, RDF, Equivalent property, Trustworthiness

## 1 Introduction

More and more data in format of Resource Description Framework (RDF) are published on Linked Open Data cloud[1]. There were 203 RDF datasets on the cloud (as of September 2010). Yet, the number of datasets has increased to 1269 as of May 2020. Many of them are cross-domain datasets describing millions of items. For example, the *DBpedia*[2], at the time of writing this paper, describes 38.3 million items in 125 languages, while the *Wikidata*[3] contains over 92 million items. Such Knowledge Bases (KBs) provide useful information for variety of applications, including Question-Answering (QA) systems – programs that answer end-users questions posed in a natural language.

Although a single KB may contain billions of facts, it often does not provide sufficient information for a QA system. Therefore, a process of collecting and fusing data from multiple different KBs seems to be a necessary part of a QA system.

In order to collect and fuse knowledge from multiple KBs two problems need to be addressed: 1) different KBs often use different vocabularies for describing items; and 2) different KBs may provide conflicting information about the same fact. Therefore, data fusion mechanisms able to handle the above mentioned problems are needed to enhance the results of QA systems.

Data fusion is the process of finding a true value from conflicting values provided by different sources [5], as well as the process of fusing multiple records representing the same real-world object into a single, consistent, and clean representation [2].

A simple approach to data fusion is a majority voting with an assumption of equally reliable sources. More advanced techniques often combine the popularity of candidate values with the reliability or trustworthiness of their sources.

Although many QA system over KBs have been developed so far, very few of them collect and resolve conflicting multi-sources data. Höffner et al. analyzed 62 different KB-based QA systems [8], but there is only one system developed by Herzig et al [7] that deals with collecting, ranking, and merging candidate results from multiple KBs. Similar situation – one or non multi-KB systems – can be found in other, more recent surveys [23, 4]. Not only surveys in Question-Answering system, but also in data fusion reveal that the problem of resolving data inconsistency often be ignored [2]. Based on that we can say that the task of fusing data and resolving a conflicting information is still and open problem.

In this paper, we propose a new approach to multi-KBs RDF data fusion that is applied to QA systems. The

[1]https://lod-cloud.net/
[2]https://wiki.dbpedia.org/about
[3]https://www.wikidata.org/wiki/Wikidata:Main_Page

 

main idea is relatively simple: a fact obtained from the most reliable (trustworthy) KB which is, at the same time, the most similar to facts retrieved from other KBs should be used to answer the user's question. The proposed method initiates degrees of trustworthiness in KBs based on analysis of the properties of RDF triples included in each KB, or alternatively based on experts' input. The degrees of trustworthiness are further updated automatically be an algorithm using information about consistency of generated answers.

In summary, the main contributions of this paper are:

- an introduction of the measure called *veracity* that is used for selecting a reliable data from a set of candidates, potentially conflicting ones, retrieved from different data sources/KBs;
- two approaches for initializing trustworthiness of data sources/KBs: an expert-based approach inspired by Saaty's priority method [17], and a data-driven approach based on the degree of equivalence of RDF properties;
- an algorithm for updating the trustworthiness of data sources based on the results of a fusion process.

## 2 Resource Description Framework

The proposed approach applies to RDF data. Therefore, we provide a short introduction the RDF format, as well as a concept of RDF-based Knowledge Bases.

### 2.1 RDF Basics

Resource Description Framework[4] (RDF) is a data framework introduced by the World Wide Web Consortium (W3C) for representing Web resources. In RDF, a resource is represented by a set of *triples*. Each triple – we call it an RDF triple – is composed of a *subject*, a *predicate* and an *object*.

A single subject could be a subject of multiple RDF triples making it an item defined/described by these triples. An example of that is show in Figure 1. The item 'defined' by multiple triples is *Lionel Messi* – an Argentinian player of the FC Barcelona. This simple set of triples provides a good illustration what to expect from the RDF data: some of the properties – *mass, birthDate, birthPlace* are functional, while others – *participant in* – are relational. In the brackets, we included names/IDs/labels of properties and nodes – *P's* and *Q's*, respectively – as they are used in *Wikidata*.

[4]https://www.w3.org/TR/rdf-primer/

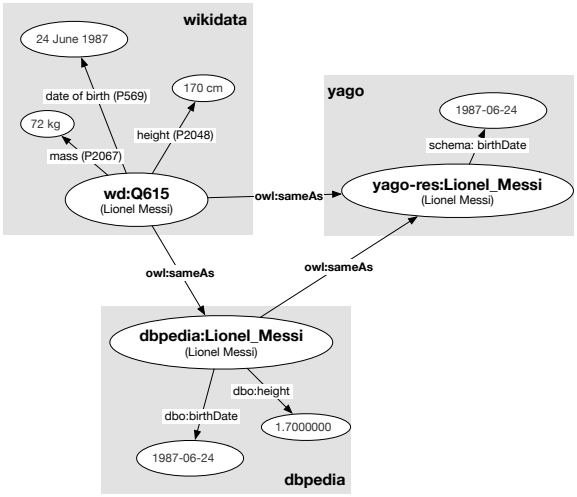


Figure 1: Multi-KB answer to query about Lionel Messi (snippet).

In general, multiple sets of names for nodes and properties exist. They are called vocabularies. Some vocabularies are used by all repositories, for example, Dublin Core [5] (dc) terms, while some are defined just for specific repositories, for example *DBpedia* ontology [6].

### 2.2 Knowledge Bases

RDF triples are building blocks for Linked Open Data (LOD) – a freely accessible collections of RDF triples that are called RDF Knowledge Bases (KBs). LOD connects thousands of RDF datasets describing items in the areas of Geography, Government, Life Sciences, Linguistics, Media, Publications, Social Networking User-Generated, and cross-domain.

In the LOD cloud, *DBpedia* and *Wikidata* are two of the largest cross-domain KBs. *DBpedia* contains close to 2 billion pieces of information (RDF triples) out of which 400 million were extracted from the English edition of Wikipedia. It is connected with *Wikidata*, *YAGO*, *GeoNames*, etc. via around 50 million RDF links. *DBpedia* is considered 'the hub' of the LOD cloud.

A very small snippet of *DBpedia* is shown in Figure 2. It includes RDF triples describing a few players of the football club *FC Barcelona*. As it can be seen, there are nodes that are connected to multiple other nodes. We can imagine that a single KB is highly interconnected.

[5]http://dublincore.org/
[6]https://www.dbpedia.org/

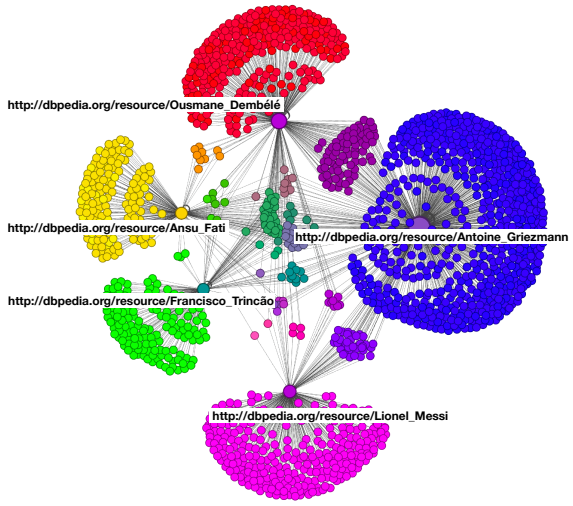


Figure 2: RDF triples representing players of *FC Barcelona* (*DBpedia*).

## 3 Fusion of Query Answers

The process of fusing information/data presented in this paper focuses on RDF data. In general, we propose an approach to identify the most reliable piece of information/data based on the data collected from multiple RDF Knowledge Bases (KBs).

### 3.1 Overview

In a nutshell, the idea is to compare triples obtained from multiple KBs and identify a single object – among all objects from the retrieved RDF triples – that is the most representative. In the process of comparing the triples, we consider three sources of inconsistency and uncertainty: 1) properties used in different KBs have different names and not always they have the same meaning, some degree of equivalence needs to be determined; 2) objects of triples, i.e., values that are subject of comparison are not always identical, some degree of similarity between should be calculated; and 3) sources of information, i.e., knowledge bases could be linked with different level of trustworthiness.

The proposed process of determining the most representable item (object) among a set of RDF triples resemble a process of identifying medoid – we look for an item with the highest degree of similarity to other items from the set.

### 3.2 Finding Single Answer

Let us have a set of triples collected from multiple KBs. For example, we query KBs [21] for a birthday of Lionel Messi (Figure 1), and obtain a set of triples with *Lionel Messi* as the subject. However, names of the property *birthDay* could be different (vocabulary dependent), as well as dates themselves could be of different format and value. In general, the set is composed of triples with the same subject, but with different properties and objects

$$RdfT = \{\langle s, p_i, x_i\rangle \mid i = 1, \ldots, N\}. \quad (1)$$

Our goal is to select a single $x_i$, a date of birth in the above example, that is a reliable value in the set *RdfT*. For that purpose, we introduce a measure called *veracity*. It is a measure that indicates how similar a single item is to other items. The selected item is identified by finding a maximum over all elements of *RdfT*

$$x^* = \max_i(veracity(x_i)). \quad (2)$$

The measure takes into account mentioned above three sources of inconsistency and uncertainty. It is calculated using the following formula

$$veracity(x_i) = \left[EQ(p, p_i) * \sum_{\substack{j=1 \\ j\neq i}}^{N} sim(x_i, x_j)\right] * T_i \quad (3)$$

where $EQ(p, p_i)$ is a degree of equivalence between $p_i$ of the RDF triple with $x_i$ as its object and the property $p$ that is treated as the reference property determined during generation of the query [21]; $\sum sim(x_i, x_j)$ represents similarity between $x_i$ and all $x_j$ of the triples from *RdfT*, and finally $T_i$ is a degree of trustworthiness of the KB from which $x_i$ is obtained.

Based on calculated values of veracity measures for all elements of the set *RdfT* we can determine a level of confidence in the value selection

$$conf(x^*) = \frac{\sum_{i=1}^{n} veracity(x_i | x_i = x^*)}{\sum_{j=1}^{n} veracity(x_j)}. \quad (4)$$

It reflects a level of consistency in the obtained answers. If all $x_i$'s are the same regardless of their veracity, the level of confidence is 1.0 – all queried KBs agree on the answer. On the other hand, if KBs do not agree on the answer – different values of $x_i$ as well as of $EQ(p, p_i)$ and of $T_i$ – the confidence value reflects that and drops below 1.0.

## 4 Property Equivalence

In this section, we provide a short description of the approach we use to determine equivalence between properties, i.e., $EQ(p, p_i)$ required for calculating veracity.

Let us have $KB_1, KB_2, \ldots, KB_n$ as data sources (Knowledge Bases) whose corresponding trustworthiness are

$T_1, T_2, ..., T_n$. One of them – say $KB_1$ without loss of generality – is selected as a reference KB – $KB_r$. It is a primary KB used in our query process [21]. A subject (in a sense of RDF triple) of a query, Lionel Messi in our example from Section 2.1, is mapped to items on other KBs, for the purpose of running queries there, via a property *owl:sameAs*. This property is defined by Web Ontology Language (OWL), to indicate that two items refer to the same thing. In such case, we denote $p$ and $p^i$ are properties of $KB_r$ and $KB_i$, respectively.

Previously, we have introduced a simple approach to determine degrees of equivalences between properties defined by different vocabularies [20]. In this paper, we propose its modified version

$$EQ(p,p') = \alpha * labelSim(p,p') + \\ + (1-\alpha) * tripleSim(p,p') \quad (5)$$

where $'$ means $i = 2,\ldots,n$.

The proposed equivalence degree is a linear combination of label-based similarity and triple-based similarity between the two properties. The former is calculated as follows

$$labelSim(p,p') = cosine(labelV(p), labelV(p')) \quad (6)$$

where $labelV(p)$ is a function that returns a vector embedding of $p$'s label.

The later is computed as follows

$$tripleSim(p,p') = \frac{\sum_{j=1}^{M} objectSim(O_j, O'_j)}{M} \quad (7)$$

where $O_j$ and $O'_j$ are non-empty sets of objects (in a sense of RDF); $M$ is the number of triple pairs $\left(\langle s_j, p, O_j\rangle \in KB_r;\ \langle s'_j, p_i, O'_j\rangle \in KB_i\right)$ such that $\langle s_j$,*owl:sameAs*,$s'_j\rangle$; and $objectSim(O_j, O'_j)$ is

$$objectSim(O_j, O'_j) = max(sim(x,y)) \quad (8)$$

where $sim(x,y)$ is a similarity degree between $x \in O_j$ and $y \in O'_j$ and is computed w.r.t their datatype (see Section 5).

## 5 Data Similarity

Objects of RDF triples could be of different datatype. The most common ones are DATE, NUMBER, STRING, and URI. The similarity between two values for each of these datatypes is calculated differently. Besides, if any value is missing we denote it as 'unknown'. The similarity of an 'unknown' with another value regardless of its datatype is always qual to 0.

To accommodate different datatypes, we proposed the following similarity measures for each of them. For DATE, we have

$$dSim(x_i, x_j) = \frac{len(commonStr(str(x_i), str(x_j))}{8} \quad (9)$$

where *commonStr* is a function that returns a leftmost longest common string, while *str* is a function that returns a string representation of a DATE in the format 'DDMMYYYY'.

The similarity between two NUMBERs is given by following formula with $\alpha \geq 0$

$$nSim(x_i, x_j) = \begin{cases} 1.0 & \text{if } x_i = x_j = 0 \\ 1 - \frac{|x_i - x_j|}{|x_i| + |x_j|} & \text{if } x_i * x_j > 0 \\ 1 - \frac{|x_i - x_j|}{|x_i| + |x_j| + max(x_i, x_j)} & \text{if } x_i * x_j < 0 \\ \frac{\alpha}{|x_i| + |x_j|} & \text{if } x_i * x_j = 0. \end{cases} \quad (10)$$

If $x_i$ and $x_j$ are two STRINGs whose vector representations are available, their similarity is computed using a vector similarity metric such as cosine

$$sSim(x_i, x_j) = cosine(V(x_i), V(x_j)) \quad (11)$$

where $V$ is an embedding vector of a string. If $x_i$ and $x_j$ are two STRINGs whose vector representations are not available, their similarity is computed using a string similarity metric to $x_i$ and $x_j$ such as Levenshtein distance, Jaro distance, or Smith-Waterman distance.

If $x_i$ and $x_j$ are two URIs, their similarity degree is calculated as follows

$$uSim(x_i, x_j) = \begin{cases} 1.0, & \text{if } \langle x_i, owl:sameAs, x_j\rangle \\ 0, & \text{otherwise} \end{cases} \quad (12)$$

## 6 Trustworthiness

The comparison of answers obtained using different KBs should include a measure indicating a degree to which we trust a given KB. It means that a process of identifying trustworthiness of KBs is quite important. In this section, we look at processes of initialization of levels of trustworthiness, as well as their update.

### 6.1 Initialization – Saaty-based Method

Given a collection of data sources (Knowledge Bases) $KB_1, KB_2, ..., KB_n$ we want to estimate their trustworthiness $T_1, T_2, ..., T_n$. The method is introduced by Saaty in 1980s, and the process of estimation is as follows.

For each pair of KBs – $KB_i$ and $KB_j$ – an expert, a designer, or a user is asked to provide her pair-wise trust in them using a scale with values from 1 and 7 (a

scale of 5, or 9 levels can be used as well). If $KB_i$ is trusted more than $KB_j$ then she is willing to assign a higher value, say 6 or 7 to the entry $(i, j)$ in a so-called reciprocal matrix R. A low value, say 3 or 2 is assigned otherwise. The value of 1 indicates that $KB_i$ and $KB_j$ are equally trusted. The inverse of the number in the entry $(i, j)$ is automatically assign to the entry $(j, i)$.

$$R = \begin{pmatrix} 1 & r_{1,2} & \cdots & r_{1,n} \\ \frac{1}{r_{2,1}} & 1 & \cdots & r_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{r_{n,1}} & \frac{1}{r_{n,2}} & \cdots & 1 \end{pmatrix}$$

Next, the maximal eigenvalue and its corresponding eigenvector are computed. The eigenvector is then normalized such that the sum of elements is equal to $n$. The normalized eigenvector is then the estimated trustworthiness.

The constructed reciprocal matrix R has an important property of transitivity. It means that for all indexes $i$, $j$, and $k$ $r_{ij} * r_{jk} = r_{ik}$. This property grantees the consistency in evaluation. Also, due to characteristic of $R$, its largest eigenvalue $\lambda_{max}$ is never less than $n$. The following ration is suggested in [16]

$$\vartheta = \frac{\lambda_{max} - n}{1 - n}$$

as an index of data inconsistency. If $\vartheta$ is less than 0.1, the estimation process is sought to be consistent. Whereas a higher value of $\vartheta$ calls for a rerun of the process.

The aforementioned one-expert priority method of trustworthiness estimation is not free of bias. To alleviate this, multiple experts can be asked to compare the trustworthiness of $Kb_i$ and $Kb_j$ using the same scale.

### 6.2 Initialization – Property-Equivalence Method

Another approach that can be used to initialize values of trustworthiness is based on determined values of equivalence of properties of two different KBs.

Assume that our reference Knowledge Base $KB_r$ has $N$ properties $p_1, p_2, ..., p_N$. The trustworthiness of another knowledge base is initialized with respect to $KB_r$ following the formula

$$T_i = \frac{\sum_{j=1}^{N} EQ(p_j, p_i)}{N} \tag{13}$$

Please, note that $p_i$ is a $KB_i$ property that is the most equivalent to $p_j$ according to $EQ(p_j, p_i^k)$ for $k = 1$ to M (M is the number of properties of $KB_i$). Again, $KB_r$ is the reference KB.

### 6.3 Trustworthiness Update

The trustworthiness $T_i$ of $KB_i$ is updated every time a reliable data ($x^*$) is determined . The update process follows Algorithm 1.

**Algorithm 1** Updating trustworthiness

```
1: procedure UPDATING(x* : trueValue, r : reward, n : numberOfSources )
2:     count ← 0
3:     simTotal ← 0
4:     for i = 1 to n do
5:         simTotal ← simTotal + sim(x_i, x*)
6:         if x_i = x* then
7:             count ← count + 1
8:         end if
9:     end for
10:    if count ≠ n then
11:        for i = 1 to n do
12:            if x_i = x* then
13:                T_i ← T_i + r/count
14:            else
15:                T_i ← T_i − r ∗ (1 − sim(x_i, x*))/(n − simTotal)
16:            end if
17:        end for
18:    end if
19: end procedure
```

The main idea of the algorithm is that whenever a reliable value is selected we update trustworthiness of KBs if they provide conflicting values. We increase the trustworthiness of the $KB_i$ if it provides $x_i$ that is identical to $x^*$ but decrease its trustworthiness an amount that is proportional to degree of dissimilarity between $x_i$ and $x^*$. Overtime, the data source $KB_i$ will be promoted more credits if it provides more values that is equal to true value.

## 7 Case Study: Results and Discussion

In order to evaluate a QA system over multiple KBs, we need a dataset that is to 'tied' to any KB, i.e., KB-independent. The existing benchmarks are KB specific: Free917 [3] and WebQuestions [1] are *Freebase*-based; versions of QALD[7] benchmark contain questions to be answered over either *DBpedia* or *Wikidata*; while the LC-QuAD [22] contains a set of complex questions that can be answered over *DBpedia*.

Therefore, to illustrate the usefulness of the proposed data fusion method for our QA system, we constructed a KB-independent dataset. It is based on the 2018 FIFA World Cup Russia List of Players downloaded from www.fifa.com. The dataset contains information about

[7] http://qald.aksw.org/

736 players from 32 national teams. This dataset is created as an KB-independent, and is treated as a 'golden standard' against which we compare the results of the data fusion experiments.

### 7.1 Data Collection

The three best-known and largest cross-domain KBs are *Wikidata*, *DBpedia*, and *YAGO*. Each of them contains billions of RDF triples. Items described by triples in these KBs are highly overlapping. Although each of them is represented by a very different name (to be precise by a different Unique Resource Identifier – URI) they are explicitly linked by the property *owl:sameAs* – and this means they all represent the same items, see Figure 1 for illustration of that.

The QA system queries each KB for player's *Birth Date*, *Height*, *Weight*, and *Club* based on the player's team and name as included in the List of Players. The details of retrieved data are in Table 1.

| Knowledge Base | Birth Date | Height | Weight | Club |
|---|---|---|---|---|
| *Wikidata* | 736 | 680 | 612 | 736 |
| *DBpedia* | 736 | 735 | – | 729 |
| *YAGO* | 717 | – | – | 735 |

Table 1: Number of datapoints collected from KBs.

### 7.2 Analysis of Retrieved Data

We evaluate the correctness of the retrieved data in terms of how accurately it matches the data provided on the List of Players. The data about a player is correct if there is an exact match with the information provided by the List. For a case of multiple football clubs, we consider the information as correct if the club on the FIFA list is included in the retrieved data. If a club name is available as a string its correctness is assessed using Levenshtein distance (the distance has to be less than three characters). If a club is identified via URI then it has to be connected via *owl:sameAs* with the URI of club on the List. Table 2 shows the results.

If we compare the content of Table 2 with one of Table 1, we see that retrieved information from KBs matches quite well data regarding *Birth Date* and *Clubs*, yet it shows many incorrect entries for *Height*.

To be accurate, we examine inconsistencies between pieces of information collected from the KBs. Table 3 includes numbers of conflicts regarding information about players' *Birth Dates*, *Heights* and *Clubs* for which they played when World Cup 2018 took place.

| Knowledge Base | Birth Date | Height | Weight | Club (string) | Club (URI) |
|---|---|---|---|---|---|
| *Wikidata* | 723 | 339 | 157 | 179 | 525 |
| *DBpedia* | 729 | 548 | – | 253 | 580 |
| *YAGO* | 708 | – | – | 171 | 508 |

Table 2: Correct answers provided by each KB.

| Birth Date | Height | Club (URI) |
|---|---|---|
| 13 | 292 | 24 |

Table 3: Conflicting information between KBs.

### 7.3 Fusing of Retrieved Data

We start experiments with fusing the data without considering trustworthiness of KBs – let us call the experiment as **Exp_0**. Table 4 shows the obtained results.

| Method | Birth Date | Height | Club (string) | Club (URI) |
|---|---|---|---|---|
| **Exp_0** | 727 | 538 | 195 | 530 |

Table 4: Correct answer for fused data; KBs' trustworthinesses not considered.

In the next experiments, we integrate the retrieved data – *Birth Date* and *Club* from all three KBs; and *Height* from *Wikidata* and *DBpedia* – with three different methods of initialization of trustworthiness. For the first and simplest method (**Exp_1**) we assign a value of 1.0 as the degree of trustworthiness for each KB. In the second case, (**Exp_2**), the values of trustworthiness are initialized using the Satty's priority method. In this approach, we use the scale of 7 to compare KBs pairwise. A reciprocal matrix R used in our experiment is

$$R = \begin{pmatrix} 1 & \frac{1}{3} & 5 \\ 3 & 1 & 7 \\ \frac{1}{5} & \frac{1}{7} & 1 \end{pmatrix}.$$

The maximal eigenvalue equals $\lambda_{max} = 3.06$, which is slightly higher than the reciprocal's dimension. The corresponding eigenvector is equal to [0.39, 0.91, 0.10] and it represents the trustworthiness values of *Wikidata*, *DBpedia*, and *YAGO*, respectively.

The third method of initialization of KB's trustworthiness (**Exp_3**) uses degrees of equivalence between properties. We selected *DBpedia* as the primary KB that it is *de facto* a hub for Linked Open Data [11]. The average values for the property-equivalence between the hub and *Wikidata*, and the hub and *YAGO* are 0.62 and 0.43. Thus, the values of trustworthiness for *Wikidata*, *DBpedia*, and *YAGO* are 0.62, 1.0, 0.43.

With such an initialization of trustworthiness, we perform three experiments with the proposed method for data fusion. The evaluated correctness of the fused data is shown in Table 5.

| Method | Birth Date | Height | Club (string) | Club (URI) |
|---|---|---|---|---|
| **Exp_1** | 724 | 339 | 179 | 525 |
| **Exp_2** | 729 | 549 | 253 | 629 |
| **Exp_3** | 729 | 549 | 253 | 629 |

Table 5: Correct answers for fused; different methods of initialization of trustworthiness.

### 7.4 Discussion

Let us analyze the obtained results. The QA system performs slightly better when we fuse data from multiple different KBs than it does based on the data collected from any single KB, Tables 2 and 5. The tables also show that the system performs much better for the case of answering questions related to clubs of players when clubs are represented by URIs than by their names (strings).

An interesting observation is that 'static' data, such as *Birth Date* are easier to be correct while 'dynamic' data such as *Height* and *Weight* are the most inconsistent, Tables 1 and 5.

It seems that the nonuniform assignment of trustworthiness to KBs produce much more accurate answers than the uniform one, Tables 5 and 4. When we compare the results of **Exp_0** with the results of **Exp_1** we can say that the numbers of correct answers when fused without taking into consideration of KBs' trustworthiness are approximately equal to ones when KBs' trustworthiness are used but equally initialized. Another interesting observation can be done when the results of **Exp_2** are **Exp_3** are analyzed: the results much better when compared with the ones obtained for **Exp_0** and **Exp_1**, and they are the same. This indicates that both initialization methods are equally effective and either of them can be used – it would depend if any experts are available or not. This also suggests that trustworthiness play an imporant role in determining a true value from ones that are provided by multiple sources of data.

## 8 Related Work

Data fusion has been mentioned for the first time by Dayal [2] in 1983. Since then it attracted a lot of attention. A simple approach is just voting. More complex approaches combine candidate voting with quality (reliability/trustworthiness/authority) of data sources; and different methods of estimating source quality have been introduced.

Page et al. [13] propose PageRank that takes advantage of the link structure of the Web. At the same time, Kleinberg [9] propose Authority-Hub analysis that is relatively similar but more complicated method to PageRank. Some researchers [6][15] introduced iterative methods that estimates both truth values of facts and trust of data sources.

In case of QA systems, many scholars [10, 14] employ data fusion techniques to increase completeness and reliability of answers. Data fusion techniques have also been developed for RDF data. Liu et al. [12] propose triple-embedding similarity-based RDF algorithms for resolving conflicts. Thoma et al. [19] introduce an entity-centric hierarchical clustering. Tasnim et al. [18] use some fusion policies to resolve conflicts when integrating different versions of RDF graph to generate entity summaries.

## 9 CONCLUSION

The exceptions of better utilization of information stored on the web in a semantically rich format of RDF create needs to build QA systems able to retrieved information and data from multiple different Knowledge Bases (KBs).

In the paper, we propose a data fusion method that can determine the most reliable (a true value) data that stands for a collection of data points retrieved from multiple KBs. We also propose and investigate methods for the initialization of trustworthiness in KBs.

Our experiments show that the trustworthiness of data sources/KBs should be included in data fusion processes. It seems that the trustworthiness should be updated over time based on how accurate information each KB provides.

We also investigate various approaches for trustworthiness initialization. The expert-based one is simple, universal, and computationally efficient, yet subjective. On the other hand, the data-based method is time-consuming but objective. However, when collecting data from multiple KBs with various vocabularies the process of determining equivalent properties is inevitable, thus the degrees of equivalence are already computed and can be utilized.

## References

[1] J. Berant, A. Chou, R. Frostig, P. Liang, Semantic parsing on freebase from question-answer pairs.

In Proc. of the 2013 Conf. on Empirical Methods in NLP (pp. 1533-1544).

[2] J. Bleiholder, F. Naumann, Data fusion. ACM Computing Surveys, 41(1), 2009, pp.1-41.

[3] Q. Cai, A. Yates, Large-scale semantic parsing via schema matching and lexicon extension. In Proc. of the 51st Annual Meeting of the Association for Computational Linguistics(Vol. 1), 2013, pp. 423-433.

[4] E. Dimitrakis, K. Sgontzos, Y. Tzitzikas, A survey on question answering systems over linked data and documents. Journal of Intelligent Information Systems, 2019, pp.1-27.

[5] X. L. Dong, E. Gabrilovich, G. Heitz, W. Horn, K. Murphy, S. Sun, & W. Zhang, From data fusion to knowledge fusion, 2015, arXiv preprint arXiv:1503.00302.

[6] A. Galland, S. Abiteboul, A. Marian, P. Senellart, Corroborating information from disagreeing views. In Proc. of the third ACM international Conf. on Web search and data mining, February 2010, (pp. 131-140).

[7] D. M. Herzig, P. Mika, R. Blanco, T. Tran, Federated entity search using on-the-fly consolidation. In International Semantic Web Conf. (pp. 167-183). Springer, Berlin, Heidelberg, October 2013.

[8] K. Höffner, S. Walter, E. Marx, R. Usbeck, J. Lehmann, & A. C. Ngonga Ngomo, Survey on challenges of question answering in the semantic web. Semantic Web, 8(6), pp.895-920, 2017.

[9] J. M. Kleinberg, Authoritative sources in a hyperlinked environment. Journal of the ACM (JACM), 46(5), pp.604-632, 1999.

[10] J. Ko, L. Si, E. Nyberg, T. Mitamura, Probabilistic models for answer-ranking in multilingual question-answering. ACM Transactions on Information Systems (TOIS), 28(3), pp.1-37, 2010.

[11] G. Kobilarov, C. Bizer, S. Auer, J. Lehmann, DbpediaâĂŞa linked data hub and data source for web applications and enterprises, 2009.

[12] W. Liu, , C. Zhang, B. Yu, Y. Li, A General Multi-Source Data Fusion Framework. In Proc. of the 2019 11th International Conf. on Machine Learning and Computing (pp. 285-289).

[13] L. Page, S. Brin, R. Motwani, T. Winograd, The PageRank citation ranking: Bringing order to the web, 1999, Stanford InfoLab.

[14] S. Park, S. Kwon, B. Kim, S. Han, H. Shim, & G. G. Lee, Question Answering system using multiple information source and open type answer merge. In Proc. of the 2015 Conf. of the North American Chapter of the Association for Computational Linguistics: Demonstrations (pp. 111-115).

[15] J. Pasternack, D. Roth, Making better informed trust decisions with generalized fact-finding. In Twenty-Second International Joint Conf. on Artificial Intelligence, June 2011.

[16] W. Pedrycz, F. Gomide, "Fuzzy systems engineering: toward human-centric computing". John Wiley and Sons, 2007.

[17] R. W. Saaty, The analytic hierarchy processâĂŤwhat it is and how it is used, Mathematical modelling, 9(3-5), (1988) pp.161-176.

[18] M. Tasnim, D. Collarana, D. Graux, F. Orlandi, & M.E. Vidal, Summarizing entity temporal evolution in knowledge graphs. In Companion Proc. of The 2019 World Wide Web Conf. (pp. 961-965).

[19] S. Thoma, A. Thalhammer, A. Harth, & R. Studer, Fuse: entity-centric data fusion on linked data. ACM Transactions on the Web (TWEB), 13(2), 2019, pp.1-36.

[20] N. D. To, M. Z. Reformat, R. R. Yager, Linked open data: Uncertainty in equivalence of properties. In Advances in Fuzzy Logic and Technology 2017 (pp. 418-429). Springer.

[21] N. D. To, M. Z. Reformat, R. R. Yager. XYZ. In IEEE SMC Conf. 2020 (pp. 418-429). IEEE.

[22] P. Trivedi, G. Maheshwari, M. Dubey, & J. Lehmann, Lc-quad: A corpus for complex question answering over knowledge graphs. In International Semantic Web Conf, October 2017, (pp. 210-218). Springer.

[23] P. Wu, X. Zhang, Z. Feng, A survey of question answering over knowledge base. In China Conf. on Knowledge Graph and Semantic Computing (pp. 86-97). Springer, Singapore, August 2019.