International Journal of Business Analytics
Volume 7 • Issue 4 • October-December 2020

Semantic Annotation of Web of Things Using Entity Linking

Ismail Nadim, Ibn Tofail University, Faculty of Sciences, Kenitra, Morocco
https://orcid.org/0000-0002-2208-3375

Yassine El Ghayam, SMARTiLab, EMSI Rabat Honoris Universities, Morocco

Abdelalim Sadiq, Ibn Tofail University, Faculty of Sciences, Kenitra, Morocco
https://orcid.org/0000-0003-1478-8807

ABSTRACT

The web of things (WoT) improves syntactic interoperability between internet of things (IoT) devices by leveraging web standards. However, the lack of a unified WoT data model remains a challenge for the semantic interoperability. Fortunately, semantic web technologies are taking this challenge over by offering numerous semantic vocabularies like the semantic sensor networks (SSN) ontology. Although it enables the semantic interoperability between heterogeneous devices, the manual annotation hinders the scalability of the WoT. As a result, the automation of the semantic annotation of WoT devices becomes a prior issue for researchers. This paper proposes a method to improve the semi-automatic semantic annotation of web of things (WoT) using the entity linking task and the well-known ontologies, mainly the SSN.

KEYWORDS

Dbpedia, Disambiguation, Internet of Things, IoT, Knowledge Base, Ontology, Probabilistic Model, Sensor, SSN

INTRODUCTION

The Internet of Things (IoT) can be viewed as the extension of the current Internet to more things and places in the physical world. The efficient connection of these things facilitates the creation of useful applications and services in numerous domains such as: transport and logistics, health, agriculture etc.However, the IoT is facing numerous challenges such as the interoperability which means that the integration of data and services from various devices- on a large scale- is extremely complex and costly.

In order to improve the interoperability between Internet of Things (IoT) devices, the Web of Things (WoT) publishes the devices capabilities on the Web in form of Web APIs. Indeed, connecting heterogeneous devices to the web makes the integration across systems and applications much simpler. This Web of Things vision allows a syntactic interoperability between devices and make easy the consumption of WoT services. However, the ambiguity of WoT data and the lack of a standardized and unified machine readable and clear semantics model hinder the semantic interoperability

DOI: 10.4018/IJBAN.2020100101

(Guinard, Trifa, Mattern, & Wilde, 2011). In addition to this, the WoT provides the interoperability at only the hardware and communication protocol level and does not add intelligence to the things or facilitate unambiguous interpretation of their data (De, Zhou, & Moessner, 2017). In contrast, service description, common practices, standards and discovery mechanisms should be interoperable to allow interactions between different objects (Elkhodr, Shahrestani, & Cheung, 2017). Thus, the semantic interoperability is essential to build a scalable WoT network because applications have to easily find and understand the WoT devices just by using their URLs. To reach this goal, the different WoT devices have to share - not only a unified syntactic form like Json or Xml- but a unified vocabulary with clear semantics as well. The process of describing WoT devices with such vocabulary is known by the semantic annotation.

The Web semantic technologies like RDF and OWL offer a framework to represent rich and complex knowledge about things, groups of things, and relations between things in a machine understandable form. These technologies have simplified a lot the sharing of semantic vocabularies like the vocabulary described by the Semantic Sensor Network (SSN) ontology (Compton et al., 2012). However, a large domain like the Web of Things needs to be open on a large knowledge from different domains. As a result, automating the semantic annotation process becomes a common goal for researchers to keep up the permanent development of WoT knowledge.

The automatic semantic annotation is usually preceded by the famous Entity linking (EL) task. EL task consists in linking a piece of data called mention from a source document to the entity it represents in a knowledge base (KB) through three steps. Given a source document, the first step is the detection of the mentions to be annotated. Once the mentions are defined, the second step consists in generating a set of candidate entities for each mention. Finally, to be able of selecting the correct entity of a mention a third step is necessary: the disambiguation. The disambiguation consists in ranking the discovered candidate sets using some features (Shen, Wang, & Han, 2015) in order to map each mention to the entity it represents the best in the KB.

In this paper, the authors focus on the disambiguation task. Specifically, they propose a collective disambiguation approach through a probabilistic graphical model which takes advantage of different types of features (in particular the semantic relatedness between candidate entities) in order to improve both the accuracy and the efficiency of the semantic annotation of WoT data. The approach aims to annotate a WoT table that is to say WoT data stored in form of Web table (header and cells).

The remaining of this paper is organized as follows: Section 2 presents the problem description and the requirements. Section 3 presents some related works. Section 4 details and discusses the proposed approach and Section 5 concludes this paper.

PROBLEM DESCRIPTION AND REQUIREMENTS

Within this first section, the authors explains formally the entity linking task and shed light, particularly, on the importance of the features it uses. Formally, given a text document D, a knowledge base KB and N mentions M={ $m_1, m_2, ..., m_N$ }, M⊂ D. The EL task consists in identifying a set of entities E ={ $e_1, e_2, ..., e_N$ }, $E \subset$ KB such as: e_i represents the referent entity of the mention m_i, i∈[1,N].

In general, for a given mention, several candidate entities may be generated when querying the KB. Fortunately, there are a set of features (statistics) that could be used for the disambiguation task such as:

- String similarity: indicates how similar is the query mention to the title of the candidate entity.
- Prior Popularity: indicates how “famous” is a candidate entity in the KB.
- Entity Type: indicates the coherence between the mention and the candidate entity types (location, person, etc..).
- Context: indicates how similar is the contextual texts of the mention and the candidate entity.

- Semantic coherence: measures the semantic relatedness between the candidate's entities

The features above can be categorized in three categories: (1) the context-independent features which rely basically on the surface form of the mention and the candidate entities, (2) the local context-dependent features which take into consideration the local context in which the mention appears and (3) the global context-dependent features which means the semantic relatedness between entities. Many disambiguation approaches have been proposed, as will be mentioned on the related works section. However, only few initiatives have combined these three types. Two disambiguation approaches are worth noting here:

- Local disambiguation: this approach ranks the candidate entities using some context-independent features as well as some local context features. After that, the best ranked entities are mapped to their corresponding mentions. However, this approach doesn't consider the interdependencies between the candidate entities.
- Global or collective disambiguation: considers that the correct disambiguation entities are not only the most similar to their corresponding mentions but further are the most "coherent" concepts. In most cases, this approach leverages the local and the global context-dependent features without considering the context-independent ones.

Accordingly, the collective disambiguation approach can be used to improve the accuracy of the entity linking task on Web tables. Indeed, considering the global context of the mentions - which means the semantic relatedness between their candidate entities- may increase the likelihood of choosing the correct mapping entities, especially because relational data tend to be semantically related. However, the local approach is still providing a baseline which is very hard to beat (Ratinov, Roth, Downey, & Anderson, 2011). The combination between these two approaches seems to be a promising solution.

In addition, the context independent features like surface form similarities, entity popularity and entity type are also significantly important for the EL task, Shen et al. (2015) reported that a naive candidate ranking method only based on the Web popularity can achieve 71% accuracy. The surface form similarity and the entity type features are also important since they help to capture the most probable entities to link the mention while maintaining a small set of candidates. Despite this, the authors have noted that few initiatives take this type of features into account during the collective disambiguation. Consequently, the main aim of this paper is to improve the EL task accuracy through a collective disambiguation method that takes advantage of different types of features. This early work toward the development of a semantic knowledge base framework is guided by similar works on EL task and knowledge engineering. In what follows some of these works are stated.

RELATED WORK

The ultime goal behind using the Entity Linking task is to automate the process of the semantic annotation. This automation should however take into account the choice of the best features to ensure the accuracy of the task. Besides, it should consider the computation time and the convergence of the EL task algorithms. In this section, the authors focus on giving some related works.

During the last years, alternative approaches for semantic annotation using the Entity Linking task and the collective disambiguation were proposed. For intance, Han et al. (2011) have proposed a global interdependence model to link mentions from text document. The proposed method leveraged the local context features, then the semantic relation between entities. Similarly, Rong et al. (2016) have used three features in their graphical model: the local similarities, the semantic relatedness and the prior popularity of a candidate entity. Although, they have added this context-independent feature to reinforce the collective disambiguation, the authors have used the Normalized Google Distance

(Cilibrasi, & Vitanyi, 2007) to measure the semantic coherence just between entities and the remaining name mentions in the same document and not between each pair entities. This consideration which can save a large amount of calculation, may affect the final accuracy of the EL task. Ganea et al. (2016) have conducted a probabilistic approach which consists in learning a conditional probability model from data and employing approximate probabilistic inference in order to find the maximum a posteriori (MAP) assignment (Koller, & Friedman, 2009). The authors have not considered any additional context-independent feature. Ratinov et al. (2011) have conducted both a local and global disambiguation to Wikipedia articles. The authors have leveraged a relatedness measure to calculate the semantic relatedness between Wikipedia entities. However, they have not mentioned the use of the context independent features in their approach.

As far as the tables annotation is concerned, Limaye et al. (2010) have used a probabilistic method to annotate web table columns and cells values with entities (persons, organizations, locations, etc.). In addition, they have used the table content (headers and data rows), and also some amount of textual context around table as a context feature. Recently, Wu et al. (2016) have provided a unified WoT Knowledge Base construction framework, and used it to annotate two types of data: plain and formatted. They have leveraged semi-automatic annotation to annotate web tables. The EL framework they have proposed, annotates entities, types and relations using features from (Mulwad, Finin, Syed, & Joshi, 2010). The authors (Mulwad et al., 2010) have used a probabilistic graphical model to manage a collective disambiguation. To infer the best disambiguation entities, they have finally adopted an iterative message passing algorithm from (Mulwad, Finin, & Joshi, 2013).

According the aforementioned related works, a categorization and reorganization of the features used by the entity task seems necessary to take the most of them. In addition to this, the semantic annotation of a formatted document may help to improve the task accuracy. Therefore, the authors assume that the WoT data to be annotated are presentable as a two-dimensional table.

PROPOSED APPROACH

In this section, the authors present their collective disambiguation approach. Firstly, the approach combines both local context-dependent and context-independent features in a local disambiguation. Secondly, the outputs of this local disambiguation is combined with a semantic relatedness measures to perform a global disambiguation. The approach is applied to a simple WoT table modeled as a probabilistic graph. An inference method is finally used to select the mapping entities leveraging a loopy message passing algorithm.

Schema Annotation

The WoT devices data can be represented as WoT tables. The approach proposes a manual annotation of the header (schema or keys) and an automatic annotation of the cells (values).

The manual annotation of the header is based on two assumptions: (1) the manual annotation is not anymore a tedious task when the number of the data to be annotated is reasonable (2) the manual annotation is safer when the data to be annotated are used as an important context to annotate other data. For example: the location (place), the type (sensor, actuator), the property observed or acts on, the unit of measurement, the value of properties for a given WoT device are important contextual data. The complete manual annotation of these data may guarantee a well construction of the KB, and helps in search and mash ups (Mulwad et al., 2010). For the moment, the first step is to determine the semantic vocabulary to use for annotating WoT devices.

Numerous conceptual models have been proposed to model devices using generic vocabularies, but no standard is yet defined: Hachey et al. (2013) grouped high-level concepts and their relations that describes three examples of real devices. CG1: Actuator, Sensor, System, CG2: Global and Local Coordinates, CG3: Communication Endpoint, CG4: Observations, Features of Interest, Units, and

Dimensions, CG5: Vendor, Version, Deployment Time. Mulwad et al. (2010) formalized that typical semantic triples in IoT scenarios as: Sensor- observes Observation, Observation-Generates-Event, Actuator-Triggers-Action, Action-Changes-Observation (State), Object Locates-Location and Owner-owns- Object. Guinard et al. (2011) proposed the web of things model which is a conceptual model of a web Thing that can describe the resources of a web Thing using a set of well-known concepts. The authors specified four resources to describe a web thing: Model, Properties, Actions and Things.

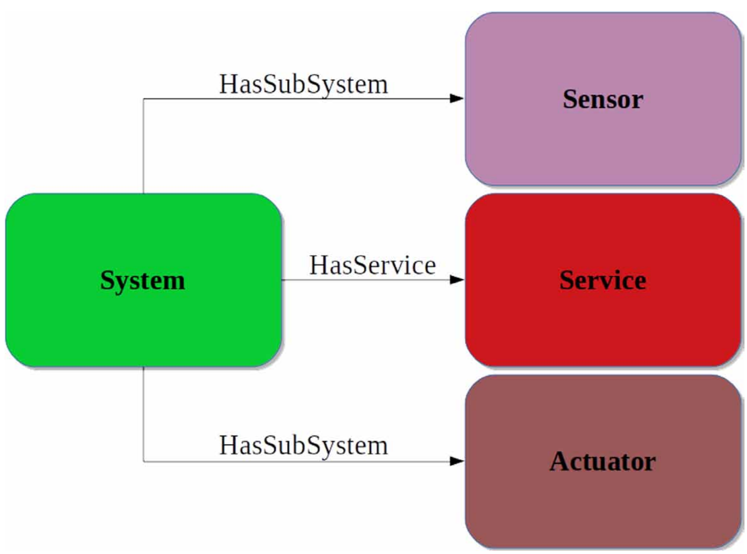
As can be noted, the ontology modelling efforts have largely concentrated on the WoT objects like sensors while other important elements like actuators, produced data, services, localization and domain ontologies should require also the same attention. Besides, the existence of different ontologies modelling the same knowledge domain like units of measurement or localization require alignment techniques to determine the correspondences between the concepts in these ontologies. For simplicity purposes, the authors propose a four components to model a WoT device.

In general, the key elements of a WoT device can be formalized as follows: System which is The WoT device and its meta-data, Sensors which are the sensors of the system and its meta-data and Actuators which represent the actuators of the system and its meta-data. And service which is the service offered by the device.

The components in Figure 1 summarizes the key concept of WoT devices:

- System: System is used here to represent the WoT device which may have subsystems mainly sensors and actuators. A system has several meta-data (name, description, manufacturer, owner …) which can be considered as a product meta-data from schema.org/Product. A system has also a location which indicates simply its deployment place or its Geo-coordinates.
- Sensor: is a subsystem of a System that can observes some property.
- Actuator: is a subsystem of a System that can acts on some property.
- Service: the service offered by the device. The service allows the control and the monitoring of the devices by capturing the sensing data or submitting actuating actions.

Figure 1. Key concepts of a WoT device



The manual annotation can be performed using a predefined mapping file which contains the mapping between the different concepts to annotate and their corresponding concepts from ontologies (SSN, SAN, Schema.org...). The data of these four elements can be stored in form of WoT tables (columns and cells). An example of such table is shown in Figure 2.

Figure 2. A WoT device in form of WoT table

ID	Value	Unit	Time	Location	Type	...

After annotating the WoT schema (table header) manually with the appropriate entities and properties from ontologies (e.g. "Location" which means the location of the device is annotated manally by the entity type "http://dbpedia.org/ontology/Place"). The next step is to annotate the content of the cells. As already mentioned, the authors use the Entity Linking task to automate this process because of the huge number of WoT data contained in the WoT tables. For this end, they first query the knowledge base (Dbpedia, …) to generate initial candidate entities sets. After that, they rank in a local disambiguation the generated sets using some features. Finally, they use - through a global disambiguation - a probabilistic graphical model to infer the best combinations of the candidate entities. In what follows, these two disambiguation methods are described.

Local Disambiguation

The search or the candidate entity generation is the process of generating a set of candidate KB entities for a mention. According to Hachey et al. (2013), this process is as important as the disambiguation task since it should capture the most probable entities to link the mention while maintaining a small set of candidates.

To improve the search task, the authors generate the initial sets of candidate entities for each cell mention by performing a string comparison between the surface form of the entity mention and the name of the entity existing in the KB. Then, by using the table header which is already annotated as context when querying the KB.

The example bellow (*Table 1*) shows the result of a Sparql query against Dbpedia to generate 20 top candidate entities to the mention "Rabat". The example uses the surface form, the entity type "Place" and the entity popularity to have better results.

Table 1. Example of Sparql query and its result

PREFIX rdf:<http://www.w3.org/1999/02/22-rdf-syntax-ns#>	http://dbpedia.org/resource/Rabat 41.0279
PREFIX dbo:<http://dbpedia.org/ontology/>	http://dbpedia.org/resource/Rabat 41.0279
PREFIX vrank:<http://purl.org/voc/vrank#>	http://dbpedia.org/resource/Rabat 41.0279
SELECT ?p ?c	http://dbpedia.org/resource/Rabat 41.0279
FROM <http://dbpedia.org>	http://dbpedia.org/resource/Rabat 41.0279
FROM <http://people.aifb.kit.edu/ath/#DBpedia_PageRank>	http://dbpedia.org/resource/Rabat 41.0279
WHERE {	http://dbpedia.org/resource/Rabat 41.0279
?p rdf:type dbo:Place.	http://dbpedia.org/resource/Rabat 41.0279
?p vrank:hasRank/vrank:rankValue ?c.	http://dbpedia.org/resource/Victoria,_Gozo 4.6521
?p rdfs:label ?x .	http://dbpedia.org/resource/Victoria,_Gozo 4.6521
?x bif:contains "(Rabat)" .	http://dbpedia.org/resource/Victoria,_Gozo 4.6521
Filter regex (str(?p),"resource").	http://dbpedia.org/resource/Rabat,_Malta 4.23087
}	….
ORDER BY DESC(?c) LIMIT 20	

The use of these three features is explained below:

- The surface form: the individuals of the knowledge base Dbpedia which contain the string "Rabat" will be chosen.
- The entity type: the individuals which have the entity type "http://dbpedia.org/ontology/Place" will be privilegied.
- The entity popularity: the filtred individuals and which are the best ranked -in terms of presence in the knowledge base- will be chosen. For example, the candidate "http://dbpedia.org/resource/Rabat" having the rank 41.0279 has more chance to be the reference entity of the mention "Rabat" than the candidate "http://dbpedia.org/resource/Victoria,_Gozo" which has only the rank 4.6521.

Once the candidate entities set is generated for each mention, these initial lists are ranked using the following features:

- By measuring the surface form similarity between the mention and the entity, using features from (Mulwad et al., 2010).
- By measuring the context similarity between the mention and the candidate entities. For that, a relevant context is created for the mention from the text surrounding it, and from the description of the column headers. The context of each candidate entity is extracted from its text description. Then, the bag of words model can be used to represent the contexts as vectors. Finally, the cosine similarity (Equation 1) is used to measure the similarity between these vectors.

$$\text{cosine}\left(\text{m,e}\right) = \frac{m * e}{\left|m\right| * \left|e\right|} \tag{1}$$

where the mention m and the entity e are represented as vectors of their context. Alternatively, some frameworks can be used like Word2Vec which is able to guess the similarity between two words based on their contexts, or Wiki2Vec that generates vectors for DBpedia entities via Word2Vec and Wikipedia Dumps.

- By considering the popularity of each entity for example the entity PageRank.

These features are computed for each candidate entity and a SVM (Support vector machine) classifier is used to rank the candidate entities for a given mention. The final result of the local disambiguation is a set of ranked candidate entities. The different steps of this process are summarized in the algorithm (*Table 2*) below:

Table 2. Local disambiguation algorithm

Algorithm 1: Local disambiguation **Input**: $\mathrm{M}=\{m_1,\mathrm{m}_2,\ldots,\mathrm{m}_N\}$, the KB **Output**: Indexed and ranked entities list L.

Let SVM be a SVM classifier.
For each $m \in M$ do
Query KB to get an initial, not null, candidate entities set I_m for the mention m.
For each I ∈ I_m do
Create a feature vector V_I.
Calculate and store features (surface form, popularity, context similarity etc.) in V_I.
Input V_I to SVM.
End For
Get L_m : the SVM output list.
Add L_m to L.
End For

Global Disambiguation

As mentioned before, the main goal of the local disambiguation process is to take the most of the context-independent features as well as the local context ones in order to get the most relevant sets of candidate entities. However, only one candidate entity should be chosen in each candidate set. Consequently an inference process should be done which is the global disambiguation.

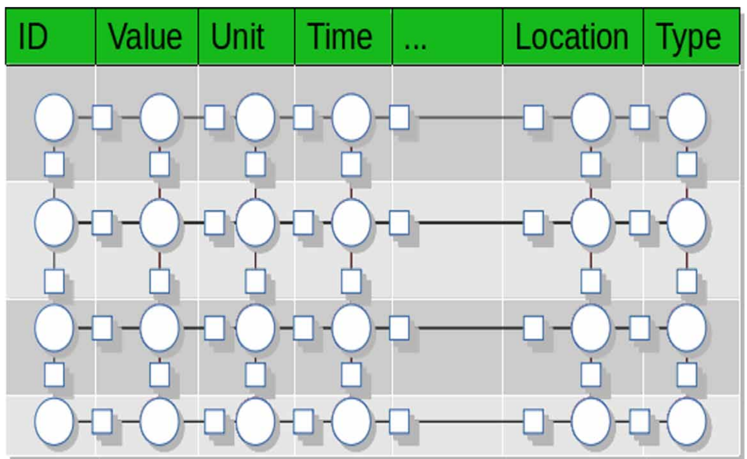
The author's approach consists in using a global disambiguation through a probabilistic graphical model (Koller et al., 2009). The idea behind using a collective disambiguation is that the entities in a given row or column of a table tend to be related. This relation can be represented mathematically through a probabilistic graphical models. For this reason, the authors represent the table cells as a Markov Network Graph (MNG) (Koller et al., 2009) such as the cells mentions represent the variable nodes and the relatedness between them represent the graph edges. To capture the dependency between a variable node (circle) and its neighbors a factor node (square) ψ is defined.

In Figure 3, the circles represent the variables and the squares represent the factor nodes of the MNG.

- ψ estimation

The ψ function can be defined as the product of two functions: R(a,b) representing the score of two candidate entities a and b in the local disambiguation, and SR(a,b) measuring the semantic relatedness between a and b.

Figure 3. WoT table represented as a Factor Graph



- *R(a,b)*: as previously seen, the output of the local disambiguation algorithm (*Table 2*) is a list of indexed and ranked entities.

Let R_a and R_b be the respective rank scores of the candidate entities a and b. The authors define the rank score of a and b as the average of their rank scores: R(a,b)= $\frac{R_a+\mathrm{R}_b}{2}$.

- *SR(a,b)*: The Milne and Witten (equation 2) (Milne et al., 2008) equation is leveraged to measure the semantic relatedness.

$$\mathrm{SR(a,b)}=1-\frac{\log\left(\max\left(\left|A\right|,\left|B\right|\right)\right)-\log\left(\left|A\cap B\right|\right)}{\log\left(\left|W\right|\right)-\log\left(\min\left(\left|A\right|,\left|B\right|\right)\right)} \quad (2)$$

where a and b are two Wikipedia (or Dbpedia) entities, A and B are the sets of all entities that link to a and b in Wikipedia (or Dbpedia) respectively, and W is the entire Wikipedia (or Dbpedia) entities.

Consequently, the function ψ can be defined as follows: ψ(a,b)= R(a,b) x SR(a,b)

And the set of mapping entities E can be determined by equation 3:

$$\sum_{j=1, j \neq i}^{N} \psi\left(e_i, e_j\right), \text{ i} \in [1, \text{N}] \tag{3}$$

- Graph inference

After defining the aim of the graph (equation 3), the problem to face is a computation one. Usually, to face the inference problem of high connected graphs the message passing algorithm - which is an approximate algorithm for graphs with loops -is used. The approach adopts a variation of this algorithm used by Mulwad et al. (2013) to avoid the pre-computation of the probability distribution tables which is costly. A brief description of the global disambiguation algorithm is given below (*Table 3*).

Table 3. Global disambiguation algorithm

Algorithm 2: Global disambiguation **Input**: The list L (Output of Algorithm 1), A maximum of iterations MaxIter, An agreement threshold T, A factor graph G(X,F) where: $X = \left\{x_1, x_2, \ldots, x_N\right\}$ is the set of variable nodes and F is the set of factor nodes. **Output**: $E = \left\{e_1, e_2, \ldots, e_N\right\}$
Initialize the variable nodes of the graph with candidate entities from the list L: $e_{\text{i,R}}$, i is the mention index, R is the first ranked candidate score and R-1 is the second ranked candidate score etc. 1. for each $x_i \in X$ do $x_i = e_{\text{i,R}}$ end for 2. converge = False, count=0 3. While (converge = False and count < MaxIter) do 4. for each $x_i \in X$ do x_i sends its current value to all the factors is connected to. 5. end for 6. for all the Factors nodes do 7. If $\psi\left(x_i, x_j\right) \geq T$ Send a no-change message to nodes x_i and x_j . 8. Else Send a change message to these nodes. end for 9. converge = True 10. for each $x_i \in X$ do 11. Let Msg be the set of messages received by x_i . 12. If all m ∈ Msg are no-change do nothing 13. Else $x_i = e_{\text{i,R}-1}$, converge = False 14. end for 15. count = count +1 16. end While 17. If (converge = True) 18. for each $i \in \left[1, N\right]$ do $e_i = x_i$ end for

First the variable nodes of the graph are initialized with the top ranked candidate entities from the list L. After that, the variable nodes in the graph send their current assignment to the factor nodes they are connected to. Once the factors receive the values of their neighboring variable nodes, they calculate the agreement between the received values using the function ψ. To decide if two values agree or not the result of the function ψ is compared to a threshold T. the value of T can be adjusted during the implementation of the approach (the higher this threshold is, the better the results after convergence are). If the value of ψ is higher than the threshold T the variable nodes values are accepted else, they are rejected and the variables receive a change message to update their assignments. The algorithm converges when no variable node receives a change message.

DISCUSSION

The authors approach combines both local context-dependent and context-independent features in a local disambiguation. After that, the outputs of this local disambiguation is combined with a semantic relatedness measures. An inference method is finally used to select the mapping entities leveraging a loopy message passing algorithm. The main advantage of this three steps approach is on one hand to ensure the quality of candidate entities. Effectively, using different features in to generate and rank the sets of candidate entities increase the probability to get better Entity Linking final result. In addition to this, the number of the candidate entities can be reduced to a reasonable number (e.g. 20 entries) without impacting the results quality. Moreover, this small number will decrease the computation task. However, some remarks are worth to mention here: (1) While the authors justified the use of the SVM classifier by the small number of entries (just dozens of entries.), the SVM classifier has proven the ability to handle large dimensions (i.e. high variables). (2) On the other hand, the approach allows the use of the relation between the cells in a table. For this end, the inference algorithm uses two thresholds which are the maximum of iterations and the convergence thresholds that must be given as inputs. While the first threshold can be chosen quit high, the second threshold should be well quantified because it may affect the convergence of the algorithm. For these remarks, the implementation of the approach is necessary to validate its efficiency. Moreover, the implementation should be preceded by a bench-marking study concerning: text processing tools, data Models, knowledge bases, training data-sets, ranking algorithms, graphical models, test data-sets, baselines, etc. This prior study will pave the path to an experiment study and a comparative evaluation of the approach.

CONCLUSION

Among the factors that improve significantly the entity linking accuracy, the authors have shed the light on the involvement of different types of features which are: the context-independent, the local context-dependent and the global context-dependent features. In this paper, they have proposed a method to combine these three types of features. First, the authors have conducted a local disambiguation in which they have reduced the contribution of the used features to a simple ranking score. After that the authors have combined this ranking score with a semantic relatedness measure in order to perform a global disambiguation through a probabilistic graphical model. Finally, they have proposed the use of a loopy message passing algorithm to infer the mapping entities.

This work is a first step towards the construction of a semantic knowledge base framework, the future step is to perform an experiment study and a comparative evaluation to prove the validity of the approach.

REFERENCES

Cilibrasi, R. L., & Vitanyi, P. M. (2007). The google similarity distance. *IEEE Transactions on Knowledge and Data Engineering*, *19*(3), 370–383. doi:10.1109/TKDE.2007.48

Compton, M., Barnaghi, P., Bermudez, L., Garcca-Castro, R., Corcho, O., Cox, S., … Taylor, K. (2012). The SSN Ontology of the W3C Semantic Sensor Network Incubator Group. *SSRN Electronic Journal*.

De, S., Zhou, Y., & Moessner, K. (2017). Ontologies and context modeling for the Web of Things. In *Managing the Web of Things* (pp. 3–36). Morgan Kaufmann. doi:10.1016/B978-0-12-809764-9.00002-0

Elkhodr, M., Shahrestani, S., & Cheung, H. (2017). Internet of things research challenges. In *Security solutions for hyperconnectivity and the internet of things* (pp. 13–36). IGI Global. doi:10.4018/978-1-5225-0741-3.ch002

Ganea, O.-E., Ganea, M., Lucchi, A., Eickhoff, C., & Hofmann, T. (2016). Probabilistic Bag-Of-Hyperlinks Model for Entity Linking. *Proceedings of the 25th International Conference on World Wide Web - WWW '16*. doi:10.1145/2872427.2882988

Guinard, D., Trifa, V., Mattern, F., & Wilde, E. (2011). From the Internet of Things to the Web of Things: Resource-oriented Architecture and Best Practices. *Architecting the Internet of Things*, 97–129.

Hachey, B., Radford, W., Nothman, J., Honnibal, M., & Curran, J. R. (2013). Evaluating entity linking with Wikipedia. *Artificial Intelligence*, *194*, 130–150. doi:10.1016/j.artint.2012.04.005

Han, X., Sun, L., & Zhao, J. (2011). Collective entity linking in web text. *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information - SIGIR '11*. doi:10.1145/2009916.2010019

Kolchin, M., Klimov, N., Andreev, A., Shilin, I., Garayzuev, D., Mouromtsev, D., & Zakoldaev, D. (2015, September). Ontologies for Web of things: a pragmatic review. In *International Conference on Knowledge Engineering and the Semantic Web* (pp. 102-116). Springer International Publishing. doi:10.1007/978-3-319-24543-0_8

Koller, D., & Friedman, N. (2009). *Probabilistic graphical models: principles and techniques*. MIT Press.

Limaye, G., Sarawagi, S., & Chakrabarti, S. (2010). Annotating and searching web tables using entities, types and relationships. *Proceedings of the VLDB Endowment International Conference on Very Large Data Bases*, *3*(1-2), 1338–1347. doi:10.14778/1920841.1921005

Mulwad, V., Finin, T., & Joshi, A. (2013, October). Semantic message passing for generating linked data from tables. In *International Semantic Web Conference* (pp. 363-378). Springer. doi:10.1007/978-3-642-41335-3_23

Mulwad, V., Finin, T., Syed, Z., & Joshi, A. (2010, November). Using linked data to interpret tables. In *Proceedings of the First International Conference on Consuming Linked Data-Volume 665* (pp. 109-120). Academic Press.

Ratinov, L., Roth, D., Downey, D., & Anderson, M. (2011, June). Local and global algorithms for disambiguation to wikipedia. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1* (pp. 1375-1384). Association for Computational Linguistics.

Rong, S., & Iwaihara, M. (2016). A collective approach to ranking entities for mentions. *2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS)*.

Shen, W., Wang, J., & Han, J. (2015). Entity Linking with a Knowledge Base: Issues, Techniques, and Solutions. *IEEE Transactions on Knowledge and Data Engineering*, *27*(2), 443–460. doi:10.1109/TKDE.2014.2327028

Wu, Z., Xu, Y., Zhang, C., Yang, Y., & Ji, Y. (2016). Towards Semantic Web of Things: From Manual to Semi-automatic Semantic Annotation on Web of Things. *Lecture Notes in Computer Science*, *9784*, 295–308. doi:10.1007/978-3-319-42553-5_25

Ismail Nadim obtained his Bachelor's Degree on mathematical sciences in 2007 from Sijilmassa High School, Errachidia, Morocco. He is an engineer in computer sciences from the National School of Mineral Industry of Rabat, Morocco. He graduated in 2013 and continues his studies in the doctoral cycle at the Ibn Toufail University of Kénitra. He is a professor of Computer Sciences in Engineering School (EMSI-Rabat) where he teaches relational databases and programming. His domain of study is Web semantic, Internet of Things, Web of Things, data sciences as well as artificial intelligence. Ismail is also a senior SQL Server and Oracle database administrator.

Yassine El Ghayam currently works at the SMARTiLab, EMSI-RABAT. Yassine did research in Software Engineering and Artificial Intelligence. He obtained a PhD degree in 2011 in the subject of Teleconferencing systems in a mobile environment, from National School of Computer Science and Systems Analysis (ENSIAS) of Rabat, Morocco. He is a professor of Computer Sciences in Engineering School (EMSI-Rabat) where he teaches programming, databases, information systems modelisation, etc. His domain of study is Web semantic, Internet of Things, Web of Things, data sciences as well as artificial intelligence. Yassine has numerous publications and contributions, namely in intelligent systems and Internet of Things.

Abdelalim Sadiq was born in Errachidia, Morocco. He received a B.S in software engineering from Sciences and technologies Faculty Moulay Ismail University - Errachidia in 1999, and DESA degree in computer network and telecommunication from National School of Computer Science and Systems Analysis (ENSIAS) Mohammed V University - Rabat Morocco in 2002 and Ph.D. degree in Computer Science from ENSIAS, Mohammed V University - Rabat, Morocco, in 2007. He is currently an Associate Professor in computer science Department of Sciences Faculty, Ibn Tofail University - Kenitra, Morocco and Team Leader Information System and Multimedia (SIM). His research interests include multimedia information retrieval and processing, sentiments analysis, IoT and data science. He has served as a reviewer for several international conferences and journals.