Event-driven Semantic Service Discovery based on Word Embeddings

FAGUI LIU1, DACHENG DENG1, JUN JIANG1, AND QUAN TANG1

1School of Computer Science and Engineering, South China University of Technology, Guangzhou 510006, China

ABSTRACT Service discovery is vital to event handling in Internet of Things (IoT) applications which are based on the event-driven service-oriented architecture (EDSOA). However, in service discovery, the problem of service matching that establishes relationships between services and events has been seldom investigated through a semantic way. In this paper, to facilitate the efficiency of service discovery triggered by events, we propose a novel method of semantic service matching based on word embeddings. In this method, two types of semantic services about events (i.e., event-recognition services and event-handing services) are specified and matched through semantic similarity assessment that is conducted with word embeddings. Besides, to obtain high-quality word embeddings, we present a hybrid approach for learning word embedding which treats words in distinct means according to word frequency. Experiments demonstrated on different datasets show that our method of semantic service matching is an effective way to facilitate event-driven service discovery, and the proposed training approach for word embeddings outperforms existing works and is able to improve the accuracy of event-driven service discovery.

INDEX TERMS Event-driven service discovery, service matching, semantic similarity assessment, word embedding

I. INTRODUCTION

By leveraging the capabilities of network devices, Internet of Things (IoT) interconnects a large number of physical entities and integrates them in smart spaces (e.g., smart home and smart warehousing). The smart space can be regarded as a physical place, in which numerous devices are interactive with each other to provide humans with relevant services in given situations [1]. Through application programming interfaces, the capabilities of IoT devices can be commonly exposed and unified as IoT services [2]. Compared to traditional Web services, IoT services provide dynamic interaction and seamless integration to bridge the physical and digital worlds [3, 4]. As an important application domain for smart spaces, services are offered to facilitate physical event handling for users in emergency response scenarios. Due to the inherent dynamic characteristic of IoT, events that indicate the state changes of entities or specific satisfied conditions constantly occur. To cope with the dynamic feature of the physical world and closely-coupled limitation of service-oriented architecture (SOA), some researchers try to apply and adapt the event-driven service-oriented architecture (EDSOA) to IoT service delivery [5-7]. In these works, physical events are extracted from sensing information and delivered through publish/subscribe or request/response paradigms. In the former paradigm, once physical events occur, they are published to service agents that subscribe these events in advance. In the later one, emerging events are delivered as a special request to the service registry for response. Obviously, two paradigms are both related to the key work that establishes the matching relationship between events and services during the subscription phase and service discovery. Due to the rapid increase in the type and quantity of IoT services and events, manual approaches for event-driven service matching are becoming more and more ineffective and inflexible. Hence, in order to meet the urgent demand for establishing relationships among services and events more automatically, service matching in the semantic way offers a promising avenue.

Semantic service discovery is the process of finding semantic services that satisfy specific requirements. In some works, a service discovery system is considered as a “matchmaker”. It is obvious that service matching is the most
important and complicated process in service discovery [8]. Similarity assessment is a significant means for semantic service matching [9]. In those similarity-based approaches, user requests are usually transformed into service patterns and matched with usable services by similarity computing based on different semantic resources. According to resource types, those approaches can be classified into two branches, i.e., knowledge-based and corpus-based methods. As a typical structured knowledge, ontologies of Semantic Web that consist of numerous concepts and properties denoting relationships between two concepts are usually applied for semantic similarity assessment of services. In an ontology, concepts can be treated as nodes of knowledge elements and construct a huge knowledge network along with properties. Therefore, the semantic similarity between two concepts can be evaluated according to the network structure of knowledge. Although ontology-based methods can take advantage of the domain knowledge of experts, researchers have raise concerns about the defects of these approaches, e.g., the limited lexical coverage and inconsistency of heterogeneous domain knowledge. Different from knowledge-based approaches, corpus-based methods usually have a large lexical coverage. These methods depend on non-structured corpora which include abundant semantic and grammatical information. For example, word embeddings have shown promising results for similarity computing in service discovery [10,11]. According to a large amount of texts in corpora, distributed word representation maps each word into a vector of real numbers, i.e., word embedding, and then the semantic similarity of words can be computed by evaluating the similarity of their word embeddings. As the quality of word embeddings has a great influence on the effectiveness of semantic similarity assessment, how to improve the quality of them during the training process becomes a key issue.

To improve the efficiency and accuracy of event-driven service discovery, in this paper, we define two types of semantic services related to events, i.e., event-recognition service and event-handling service, and propose an approach for semantic service discovery where services are matched using word embeddings. Additionally, aiming to obtain high-quality word embeddings, we further introduce a training approach for word embeddings that treats words in different ways according to word frequency and integrate the semantic information contained in knowledge bases and corpora. The remainder of this paper is organized as follows: Section II introduces the related work of event-driven service discovery and semantic similarity assessment. The service matching model for events and training approach for word embeddings are presented in Section III. Experiments and results are given in Section IV. Conclusions and future works are discussed in Section V.

II. RELATED WORK

The essential goal of event-driven service matching is to establish relevant relationships between events and services, especially in IoT context. Existing researches mainly accomplish it manually and stably by a priori knowledge. That is to say, services and events are appointed matched pairs by subjective experience rather than automatic matchmaking. Liu et al. [5] proposed an event-driven service-oriented computing platform, where an event agent is responsible for event collection and assigns events to particular service agents for event handling. Nevertheless, the assignment strategy is not definitely given. Zhao et al. [6] put forward an event-driven service provisioning mechanism, in which services are described by ontology Web language for service (OWL-S) [12] and the information about events is depicted using ontologies. Service systems should subscribe specific events empirically in advance. Once events are triggered, the information of events will be published to corresponding services. Yachir et al. [13] presented a service-oriented event-aware framework for performing services to cope with occurred events in the scenario of ambient assisting living. In this framework, event rules that specify functional requirements when a specific event occurs are used to implicitly match services. Park et al. [14] presented a service pattern for context-aware services that are defined as a predefined rule consisting of a condition and an action. If physical events happen, certain actions of services are triggered according to the rule for serving users. Although events and service are directly matched by rules, these rules still should be set manually and stably. Consequently, with the increase of events and services in IoT context, a major problem with these non-automated matchmaking is the poor efficiency. In order to make event-driven service discovery more efficiently, semantic service matching is indeed required while few of existing works address this issue.

Currently, there are a number of research works that use similarity assessment for semantic service matching. In these works, Web services could be depicted by semantic description languages such as OWL-S and Web service description language semantics (WSDL-S) [15]. OWL-S is a specific OWL ontology based on description logic and capable of providing a framework for semantic description of services. Especially for IoT services, some extended specification models are put forward based on OWL-S, such as OWL-S\textsuperscript{tm} [16] and OWL-SE [17]. In terms of semantic Web services, similarity assessment can be generally accomplished by using semantic relationships between concepts in ontologies [18,19]. On the foundation of a hierarchical ontology, semantic similarity measures can roughly fall into two types, i.e., information content (IC)-based approaches and path-based approaches [20]. As an example of former approaches, Gao et al. [21] proposed a measurement method for semantic similarity where similarity between concepts is assessed on the basis of IC values which are computed from WordNet [22]. As a typical knowledge base, WordNet represents each concepts of words as a set of synonymous words (named synset) and includes diverse semantic relations between concepts, such as hypernym/hyponyms (is-a) and antonyms. Path-based approaches employ the structure of knowledge instead of statistical information of words for similarity computing. Focusing on IoT services, Jia et al. [23] proposed a service matching method of service discovery where IoT services are described by OWL-S\textsuperscript{iot} and parameters of services such as input and output are used for similarity assessment. In this method, the similarity of services is evaluated through concept similarity.
computing, in which the depth of concepts and path distance between concepts in a tree-like ontology are taken into account. However, ontology-based approaches requires various domain ontologies which are usually limited in lexical coverage and lead to the limitation of applicability.

In corpus-based approaches for similarity assessment, the semantic similarity between words can be measured by word vectors. As an alternative of distributional word vector, word embeddings are obtained by neural-network language modeling and represented by low-dimensional continuous vectors [24-27]. The quality of word embeddings that has a significant impact on semantic similarity assessment is depended on the training process. Mikolov et al. [27] proposed an unsupervised learning model, continuous bag-of-words (CBOW). For a target word, the purpose of this model is to maximize the log-likelihood of the word in certain given context. In order to further improve the quality of word embeddings, several methods that integrate both the information of prior knowledge and corpora into vectors are proposed. Lu et al. [28] proposed a multiple semantic fusion (MSF) model that first train the initial word embeddings by CBOW, and then utilize different strategies to modify word embeddings. These strategies are mathematical vector operators, such as average and maximum. Eventually, the final word embeddings are used for service matching. Yu et al. [29] presented a relation constrained model (RCM) that not only considers the neighbors of words in a corpus but also the relationships between words in a lexicon to establish the objective function in the training process. Similar to RCM, Bollegala et al. [30] put forward a method to learn word embeddings both using the corpus and lexicon. Especially, this method considers the co-occurrence of words in a corpus which is described by a word co-occurrence matrix. Faruqui et al. [31] proposed a retrofitting model to improve the quality of word embeddings. This model utilizes the relationship between words in additional lexicons to optimize the word embeddings as a second stage of learning. In a word, the quality of word embeddings has a crucial impact on semantic similarity assessment and relies on the training process. These training approaches use corpora and knowledge bases to learn word embeddings, and treat all words in the same way without making difference between words according to their word frequency. Whereas, word frequency is a potential influence factor on the quality of word embeddings during the training process [32]. That is to say, for the words with a low frequency in a corpus, the above training methods tend to obtain poor word embeddings for low-frequency words due to the lack of context.

III. EVENTS-DRIVEN SEMANTIC SERVICE DISCOVERY

Different from the traditional service discovery, the triggers of event-driven service discovery are physical events rather than user requests. Therefore, in this paper, we specify two types of services about events, (i.e., event-recognition service and event-handling service) and introduce a framework of event-driven service discovery, as shown in Figure 1. In this framework, once events are recognized by event-recognition services, these services are triggered to be matched with event-handling services for response. For example, when an event-recognition service is aware of an anomalous change about temperature, an automatic matchmaking between this service and event-handling services in the service base will start. As a result, matched services will be candidates for event response, such as thermoregulation and alarm. Particularly, services here are described in a semantic way and matched based on word embeddings. During the training process of word embeddings, knowledge bases and corpora are used as training resources and words are treated in distinct ways according to their frequency. Hence, the proposed training approach for word embeddings contains three phases, i.e., high-frequency word processing, low-frequency word processing and joint processing.
A. THE APPROACH FOR TRAINING WORD EMBEDDINGS

1) High-frequency word processing

Because a wealth of co-occurrence relations among high-frequency words are implicit in corpora, in the first phase, we apply CBOW, due to its effectiveness and simplicity, to train word embeddings of high-frequency words. The goal of CBOW is to maximize the log-likelihood of target words in given context. In other words, for a word in a sentence, words around it are called context. The purpose of CBOW is to maximize the probability of occurrence when context is given. The objective function is as follows:

\[
\text{Obj} = \frac{1}{T} \sum_{t=1}^{T} \log p(w_t | w_{t-c}^t + c),
\]

where \(w_t\) is the target word, \(T\) is the size of corpora (i.e., the total number of words in corpora), \(c\) is the window size of context around target word \(w_t\), and hence \(w_{t-c}^t + c = \{w_{t-c}, \ldots w_{t-1}, w_{t+1}, \ldots w_{t+c}\}\) is the context of \(w_t\). Besides, \(p(w_t | w_{t-c}^t + c)\) is defined as follows:

\[
p(w_t | w_{t-c}^t + c) = \frac{\exp(\hat{e}(w_t)) \cdot \sum_{i=-c}^{c} e(w_{t+i})}{\sum_{i=1}^{T} \exp(\hat{e}(w_i)) \cdot \sum_{i=-c}^{c} e(w_{t+i})},
\]

where \(\hat{e}(w)\) and \(e(w)\) denote the input and output embeddings of word \(w\) respectively, and \(N\) is the size of vocabulary. Empirically, in this paper, the context window \(c\) is set to 5, and the dimensionality of word embeddings is set to 400.

2) Low-frequency word processing

Given word embeddings of high-frequency words are obtained in phase 1, in this processing phase, word embeddings of low-frequency words are obtained based on these high-frequency words which have a semantic relationship with low-frequency words in knowledge bases. Inspired by the basic ideal that algebraic operations performed on word vectors might reflect the semantic relationships between words [27,33], the semantic generation model (SGM) is proposed below:

\[
e(w) = \sum_{k=1}^{n} \omega_k \sum_{w_i \in R^w_k} e(w_i),\]

where \(n\) is the number of types of semantic relationships (e.g., synonym, hypernym and hyponym), \(\omega_k\) is the weight for different relationships, and \(R^w_k\) is the set of high-frequency words \(w_i\) that has a semantic relationship \(R_k\) with low-frequency word \(w\).

3) Joint processing

After the foregoing two phases, obtained word embeddings of high-frequency and low-frequency words are processed together in the phase of joint processing to further encode the information of semantic relationships into word embeddings. In this phase, we refer to the retrofitting model [31] and propose an advanced model named cosine similarity retrofitting (CSR) to enhance the cosine similarity of word embeddings of target words.

Let vertex set \(V = \{w_1, w_2, \ldots, w_N\}\) be the vocabulary and edge set \(E\) be the set of word pair \((w_i, w_j)\) \(\subseteq V \times V\), with an interesting relationship in a knowledge base. For illustration, a word-embedding graph is established, and a simple example is shown in Figure 2. The grey and blue vertices represent the initial and retrofitted word embeddings of words in \(V\) respectively, and these edges belong to relationships in \(E\). Aims to learn the retrofitted word embeddings that close their correspondent and consecutive vertices, the objective function is established to be maximized as below:

\[
\Phi(v) = \sum_{i=1}^{N} [\alpha \cdot \cosSim(v_i, \hat{v}_i) + \sum_{(w_i, w_j) \in E} \beta \cdot \cosSim(v_i, v_j)],
\]

where \(N\) is the size of vocabulary, \(\hat{v}_i\) denotes the vector representation of \(w_i\), \(v_i\) is the retrofitted word embeddings of \(w_i\) and \(\alpha\) and \(\beta\) are weights for two similarities. Besides, the cosine similarity between two vectors is

\[
\cosSim(v_i, \hat{v}_i) = \frac{v_i \cdot \hat{v}_i}{|v_i||\hat{v}_i|}.
\]

Figure 2. A simple example of word-embedding graph

Because Equation (4) is non-convex, we use gradient ascent method to get an approximate optimal solution. We first calculate the partial derivative of \(v_i\) and obtain Equation (6),

\[
\frac{\partial \Phi(v)}{\partial v_i} = \alpha \cdot \left(\frac{1}{|v_i|^2} \cdot \hat{v}_i - \frac{\cosSim(v_i, \hat{v}_i)}{|v_i|^2} \cdot v_i\right) + \sum_{(l,j) \in E} \beta \cdot \left(\frac{1}{|v_l|^2} \cdot \hat{v}_l - \frac{\cosSim(v_l, v_j)}{|v_l|^2} \cdot v_l\right).
\]

According to Equation (6), the iterative formula is given as Equation (7), and the final word embeddings are obtained when a stopping criterion (e.g., iteration count) is satisfied.
In EDSOA, events can be treated as a special type of service requests. In this section, we introduce a model for service matching and services here are described by OWL-SE which mainly contains four aspects of information in service profile, i.e., “input”, “output”, “precondition” and “result”. The “input” and “output” represent essential function features that are generally used for service matching. Therefore, the capability of event recognition and event handling are formulated as services which are presented by Equations (8)-(10) based on description logic. In this specification, events are defined as the outputs of event-recognition services (ERS) and inputs of event-handling services (EHS) respectively. Moreover, effects that mean actions to handle events are denoted as outputs of event-handling services.

\[
v_i = v_i + \eta \cdot [v_i] \cdot \left( \alpha \cdot \hat{v}_i + \sum_{(i, j) \in E} \beta \cdot v_j \right) + \eta \cdot (\alpha \cdot \text{CosSim}(v_i, \hat{v}_i) \cdot v_i)
\]

(7)

The complete training procedure is shown as Algorithm 1, which contains totally three phases, i.e., high-frequency word processing (lines 3-6), low-frequency word processing (lines 7-10) and joint processing (lines 11-13). In the high-frequency processing, word embeddings of high-frequency are obtained by CBOW with corpora. In the low-frequency word processing, word embeddings of low-frequency words are obtained by SGM with semantic relationships extracted from knowledge bases and word embeddings of high-frequency words. In the joint processing, all word embeddings gained in previous two phases are further optimized by CSR. Note that the high-frequency and low-frequency words are classified by a boundary of word frequency which is given according to the size of corpora.

### Algorithm 1: The hybrid learning algorithm for word embeddings

**Input:**
- A corpus \( C \); A knowledge base \( K \);
- A semantic relationship set \( R \);
- The boundary frequency \( \rho \);
- The window size of CBOW model \( c \);
- The dimensionality of word embeddings \( d \);
- The weights for relationships \( \omega_{rk} \);
- The weights \( \alpha, \beta \) and total number of iterations \( T \) for CSR;

**Output:**
- The learned word embedding of word \( w, e(w) \);

1: Extract vocabulary \( V \) from the corpus \( C \);
2: Divide \( V \) into two set by \( \rho \), i.e., the set for high-frequency and low-frequency words \( V_h \) and \( V_l \);
3: Initialize word embeddings of word \( w_t \) in \( V_h, w_t \in \mathbb{R}^d \);
4: For each word \( w_t \) in \( V_h \):
   5: Extract the context of \( w_t \) from \( C \), \( w_t^{i+c} \subseteq V_h \)
   6: \( e(w_t) = \text{CBOW}(w_t, w_t^{i+c}) \)
7: For each word \( w \) in \( V_l \):
   8: For each relationship \( R_K \in R \):
      9: Extract the related word set \( R^K_w \subseteq V_h \) of \( w \) from \( K \);
      10: \( e(w) = SGM(R^K_w, \omega_{rk}, e(w)) \)
11: Extract the set of word pairs \( E \) from \( K, (w_i, w_j) \in E \);
12: Initialize \( v_i, \hat{v}_i \) by the word embedding \( e(w_i) \);
13: \( e(w) = \text{CSR}(E, \alpha, \beta, v_i, \hat{v}_i, T) \)

### B. SEMANTIC SERVICE MATCHING FOR EVENTS

In EDSOA, events can be treated as a special type of service requests. In this section, we introduce a model for service matching and services here are described by OWL-SE which mainly contains four aspects of information in service profile, i.e., “input”, “output”, “precondition” and “result”. The “input” and “output” represent essential function features that are generally used for service matching. Therefore, the capability of event recognition and event handling are formulated as services which are presented by Equations (8)-(10) based on description logic. In this specification, events are defined as the outputs of event-recognition services (ERS) and inputs of event-handling services (EHS) respectively. Moreover, effects that mean actions to handle events are denoted as outputs of event-handling services.

\[
\text{Event} \sqsubseteq \text{Event\_Recognition\_Service}\_\text{hasInput} \quad \text{(8)}
\]

\[
\text{Effect} \sqsubseteq \text{Event\_handling\_Service}\_\text{hasOutput} \quad \text{(10)}
\]

According to Equations (8)-(10), the service matching model for events is represented as follows:

\[
\text{match}(ERS, EHS) = \begin{cases} 1, & \text{Sim}(E_r, E_h) > \tau \\ 0, & \text{otherwise} \end{cases}
\]

(11)

where \( \tau \) is a given threshold, \( E_r \) and \( E_h \) denote the output event of ERS and input event of EHS respectively, and \( \text{Sim}(E_r, E_h) \) is the similarity of events. When \( \text{Sim}(E_r, E_h) \) is larger than \( \tau \), events \( E_r \) and \( E_h \) are treated as similar, and ERS and EHS are considered to be successfully matched.

With the consideration of event attributes, such as object
and location, the similarity assessment of events is computed by Equation (12).

\[
Sim(E_r, E_h) = \sum_{a \in \text{attr}(E_r)} W_a \cdot Sim_a(E_r^a, E_h^a), \tag{12}
\]

where \(\text{attr}(E_r)\) is the attribute set of \(E_r\), 
\(W_a = \frac{1}{|\text{attr}(E_r)|}\)

and 
\(Sim_a(E_r^a, E_h^a)\)

\[
= \begin{cases} 
1, & E_h = \emptyset \\
0, & \max \{\text{sim}(E_r^a, E_h^a) \mid i \in \text{attr}(E_h)\}, \\
\end{cases} \tag{13}
\]

where \(\text{sim}(E_r^a, E_h^a)\) is the similarity of event attributes which is calculated by the cosine similarity of word embeddings.

IV. EXPERIMENTS AND ANALYSIS

A. WORD SIMILARITY COMPARISON

In this study, we apply a public corpus Wikipedia\(^1\) which has a large amount of vocabulary about 8.87 million and 2 billion words. The lexicon WordNet is used here and three types of relationships in it are considered, i.e., the synset, hypernym and hyponym. The pre-processing of corpora and implementation of CBOW are on the foundation of a publicly available toolkit, word2vec\(^2\). Experiments are conducted on a server with 32G main memory and 8-core CPU, running Ubuntu 14.04. To evaluate the quality of word embeddings, the commonly used task of word similarity comparison is implemented. The goal of this task is to check how closely the similarity of words are judged by learned word embeddings and human. The experiments are conducted on four benchmark datasets, i.e., RG-65 [34], WS-353 [35], WS-sim and WS-rel [36]. WS-sim and WS-rel are both the subset of WS-353. Specifically, WS-sim is the union of similar and unrelated pairs, while WS-rel is the union of related and unrelated pairs. Each dataset contains a certain number of word pairs with the similarity scores assessed by human. Two evaluation indexes, Spearman rank correlation coefficient and Pearson correlation coefficient, are utilized to compare the proposed methods and existing works. Main criteria are set: \(\rho = 3000, \omega_k = 0.33, \alpha = \beta = 0.5\) and \(T = 10\).

| TABLE I |
| THE PERFORMANCE OF CBOW, MSF, HYBRID-R AND HYBRID-CSR WITH RESPECT TO PEARSON CORRELATION COEFFICIENT WITH RG-65, WS-353, WS-REL AND WS-SIM DATABASES |
|---|---|---|---|
| RG-65 | WS-353 | WS-rel | WS-sim |
| CBOW | 0.781 | 0.633 | 0.548 | 0.748 |
| MSF | 0.834 | 0.553 | 0.370 | 0.737 |
| Hybrid-R | 0.825 | 0.635 | 0.501 | 0.784 |
| Hybrid-CSR | 0.830 | 0.641 | 0.510 | 0.793 |

We compare our method with models CBOW and MSF. Hybrid-R denotes the approach that uses the origin retrofitting model [30] in the joint processing phase. Hybrid-CSR presents the method utilizing the improved retrofitting model CSR. Table 1 is the result with respect to Pearson correlation coefficient and Table 2 shows the result in terms of Spearman rank correlation coefficient. As shown in Table 1 and 2, MSF has an advantage with the word pair RG-65 and CBOW outperforms the other models with the word pair WS-rel. Because WS-rel emphasizes relatedness rather than similarity which is mainly considered in Hybrid-CSR. Hybrid-CSR with WS-rel is not as good as with the other benchmark datasets. Obviously, Hybrid-CSR achieves a preferable performance than Hybrid-R with all word pairs, and outperforms the other models with word pairs WS-353 and WS-sim.

B. SEMANTIC SERVICE MATCHING FOR EVENTS

To verify the approach for service matching in event-driven service discovery, we conduct an experiment in the scenario of logistics storage. We consider five objects i.e., temperature, humidity, goods, route and devices. For events about temperature, humidity and route, the consumer can be alarm services. For the event about devices, the consumer might be the alarm services, maintenance service, etc. In this scene, six volunteers with the domain knowledge of logistics have participated in the construction of the service base. As for the description of events, we refer to the event model [37] and consider five attributes, i.e., time, location, observation object, state and consumer. Event-recognition services and event-handling services here are established by Protégé 3.2 (with a plug-in named owl-s editor), an ontology editor. Because the attribute values of services should be represented by XML, Schema or URI, the attribute value about events, except time, are flexibly given by volunteers with various word in “string” type and used for service matching.

Semantic services from the OWL-TC4\(^3\) are related to 9 domains, i.e., simulation, education, medical care, food, travel, communication, economy, weapons, and geography. Because the simulation services are involved in the state change of indoor objects, to make this experiment more convincing, those simulation services are transformed into event-handling services and put into the service base as negative samples. In total, the service base includes 166 event-recognition services (37 about devices, 33 about route, 32 about temperature, 29 about humidity and 35 about goods) and 182 event-handling services.

| TABLE II |
| THE PERFORMANCE OF CBOW, MSF, HYBRID-R AND HYBRID-CSR IN TERM OF SPEARMAN RANK CORRELATION COEFFICIENT WITH RG-65, WS-353, WS-REL AND WS-SIM DATABASES |
|---|---|---|---|
| RG-65 | WS-353 | WS-rel | WS-sim |
| CBOW | 0.793 | 0.646 | 0.542 | 0.745 |
| MSF | 0.818 | 0.548 | 0.352 | 0.724 |
| Hybrid-R | 0.813 | 0.658 | 0.506 | 0.769 |
| Hybrid-CSR | 0.831 | 0.664 | 0.513 | 0.778 |

\(^1\) http://dumps.wikimedia.org/enwiki/20170520/
\(^2\) http://word2vec.googlecode.com/svn/trunk/
\(^3\) http://projects.semwebcentral.org/projects/owl-s-tc/
services. The service profile of an event-recognition service about goods is shown as Figure 3. This service with ID “ERS_025” has an output “Goods_Event025” that is described by 5 attributes in detail.

In this experiment, we apply the proposed approach for service matching with different threshold values for service discovery. It is assumed that all event-recognition services capture physical events once and are triggered to be matched with event-handling services in the service base. Three criteria precision, recall and F-measure are used to evaluate the effectiveness of our approach. Aiming to analyze the influence of the word embeddings on service matching, we conduct the experiment by using word embeddings obtained by CBOW and Hybrid-CSR respectively. The experiment result is shown in Figure 4. With the increase of the threshold, precision increases while recall decreases. F-measure obtain a maximum value when the threshold is set to a proper value. It means that when the threshold value is relatively small, two approaches are both likely to match incorrect services. However, when the threshold value is relatively large, some correct services might be omitted. In practice, the threshold value can be set according to the scale of service bases. That is to say, if the number of event-handling services is large, to match services more precisely, the threshold should be set to a relative large value. On the contrary, when the number of candidate event-handling services is small, to obtain the correct services as much as possible, the threshold should be set to a relative small value.

The comparison result of CBOW and Hybrid-CSR with respect to F-measure (optimal value) is shown as table 3. Conclusively, the proposed service matching method is effective for event-driven service discovery, and Hybrid-CSR has a positive impact on the effectiveness of service matching by improving the quality of word embeddings.

<table>
<thead>
<tr>
<th></th>
<th>CBOW</th>
<th>Hybrid-CSR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Goods</td>
<td>0.559</td>
<td>0.585</td>
</tr>
<tr>
<td>Humidity</td>
<td>0.737</td>
<td>0.742</td>
</tr>
<tr>
<td>Device</td>
<td>0.686</td>
<td>0.691</td>
</tr>
<tr>
<td>Route</td>
<td>0.577</td>
<td>0.621</td>
</tr>
<tr>
<td>Temperature</td>
<td>0.702</td>
<td>0.698</td>
</tr>
<tr>
<td>Total objects</td>
<td>0.633</td>
<td>0.660</td>
</tr>
</tbody>
</table>

TABLE III

THE COMPARISON RESULT OF CBOW AND HYBRID-CSR WITH RESPECT TO F-MEASURE
Events are the significant manifestation of dynamic characteristics in IoT. In this paper, we investigate the problem of service matching in event-driven service discovery. Although there are several methods of service delivery triggered by events, few of them address this issue automatically with semantic service matching. The main contribution of our work consists of two parts: (1) a clearly presented method of training word embeddings by using both corpora and lexica; (2) a newly proposed approach for event-driven semantic service matching based on word embeddings. Implemented on different datasets, the experiments show that our approach provides an effective way to discover services automatically for handling events, and our training method of word embeddings outperform existing approaches and enhances the accuracy of service discovery. In our future work, we shall analyze the performance of the newly proposed approach for event-driven service discovery in large-scale IoT application scenarios. In addition, we plan to train word embeddings based on corpora which are extracted from specific application fields.

V. CONCLUSIONS AND FUTURE WORKS

Events are the significant manifestation of dynamic characteristics in IoT. In this paper, we investigate the problem of service matching in event-driven service discovery. Although there are several methods of service delivery triggered by events, few of them address this issue automatically with semantic service matching. The main contribution of our work consists of two parts: (1) a clearly presented method of training word embeddings by using both corpora and lexica; (2) a newly proposed approach for event-driven semantic service matching based on word embeddings. Implemented on different datasets, the experiments show that our approach provides an effective way to discover services automatically for handling events, and our training method of word embeddings outperform existing approaches and enhances the accuracy of service discovery. In our future work, we shall analyze the performance of the newly proposed approach for event-driven service discovery in large-scale IoT application scenarios. In addition, we plan to train word embeddings based on corpora which are extracted from specific application fields.

REFERENCES


Fagai Liu received the M.S. degree from Beihang University and Ph.D. degree from South China University of Technology in 1991 and 2006 respectively. Currently she is a professor at the School of Computer Science and Engineering, South China University of Technology, Guangzhou, China. Her research interests include service computing, Internet of things, cloud computing and big data.
Dacheng Deng received the B.S. degree from Department of Computer and Electronic Information, Guangxi University, Nanning, China in 2013. He is currently pursuing Ph.D. degree in School of Computer Science and Engineering, South China University of Technology, Guangzhou, China. His research interests include semantic Web, service computing and natural language processing.

Jun Jiang received the B.S. degree in Information Management and Information System from Shihezi University, China in 2017. He is currently working toward the Ph.D. degree at the School of Computer Science and Engineering, South China University of Technology, Guangzhou, China. His research interests include semantic Web, data stream classification, neural networks and fuzzy system.

Quan Tang received the B.S. degree from South China University of Technology, Guangzhou, China, in 2018. He is now working toward the Ph.D. degree at the School of Computer Science and Engineering, South China University of Technology, Guangzhou, China. His research interests include machine learning, semantic reasoning, and approximate reasoning.