Ontology-Learning-Based Focused Crawling for Online Service Advertising Information Discovery and Classification

Hai Dong[1], Farookh Khadeer Hussain[2], and Elizabeth Chang[1]

[1] School of Information Systems, Curtin University of Technology, Australia
[2] School of Software, University of Technology, Sydney, Australia

Abstract. Online advertising has become increasingly popular among SMEs in service industries, and thousands of service advertisements are published on the Internet every day. However, there is a huge barrier between service-provider-oriented service information publishing and service-customer-oriented service information discovery, which causes that service consumers hardly retrieve the published service advertising information from the Internet. This issue is partly resulted from the ubiquitous, heterogeneous, and ambiguous service advertising information and the open and shoreless Web environment. The existing research, nevertheless, rarely focuses on this research problem. In this paper, we propose an ontology-learning-based focused crawling approach, enabling Web-crawler-based online service advertising information discovery and classification in the Web environment, by taking into account the characteristics of service advertising information. This approach integrates an ontology-based focused crawling framework, a vocabulary-based ontology learning framework, and a hybrid mathematical model for service advertising information similarity computation.

1 Introduction

It is well recognized that the information technology has a profound effect on the conduct of the business, and the Internet has become the largest marketplace in the world. Innovative business professionals have realized the commercial applications of the Internet for their customers and strategic partners. They therefore turn the Internet into an enormous shopping mall and a huge catalogue [1]. In the service industry, Internet advertising is also popular among small and medium enterprises, due to the advantages of low cost, high flexibility, and ease of publishing. Nevertheless, many service consumers find it difficult to quickly and precisely retrieve service advertising information from the Internet, not only owing to the lack of specialized service information registration and retrieval platforms, but also because of the following features of service advertising information.

Ubiquity. Service advertisements can be registered by service providers through various business information registries [2]. These business information registries are geographically distributed over the Internet, yet there is no particular approach or application being designed to quickly and precisely locate the service information from these registries.

C. Liu et al. (Eds.): ICSOC 2012, LNCS 7636, pp. 591–598, 2012.

Heterogeneity. Given the diversity of services in the real world, many schemes have been proposed to classify services from various perspectives. Nevertheless, there is not a publicly agreed scheme available for classifying service advertising information over the Internet.

Ambiguity. Most of service advertising information does not retain a consistent format or standard. They are described by natural languages and embedded in vast Web information, the content of which is sometimes ambiguous for service consumers to understand.

Service (information) discovery is not a fresh topic in the academia. Many theories and applications have been developed so far. Nevertheless, at present few studies have been carried out in the research area of service advertising information discovery, by taking into account the above features of service advertising information.

In order to address this research issue, in this paper, we propose a novel ontology-learning-based focused crawling approach for service advertising information discovery and classification. The proposed approach is the integration of a semantic focused crawling framework, an ontology-learning framework, and a hybrid service advertising information similarity model. The semantic focused crawling framework is to address the issues of service advertising information for service information discovery; the ontology learning framework is to solve the limitations of ontology-based focused crawling; and the hybrid model is to measure the relatedness of service advertising information from the perspectives of text similarity and statistics.

2 Related Work

A semantic focused crawler is a software agent that is able to traverse the Web, and retrieve as well as download related Web information for specific topics, by means of semantic Web technologies [3], [4]. The goal of semantic focused crawlers is to precisely and efficiently retrieve and download relevant Web information by understanding the semantics underlying the Web information and the semantics underlying the predefined topics. According to a survey conducted by Dong et al. [5], the limitation of the semantic focused crawlers is that their crawling performance crucially depends on the quality of ontologies. This eventual consequence of this problem could be reflected in the gradually descending curves in the performance of semantic focused crawlers.

In order to solve the above issue, researchers start to pay their attention to enhancing semantic focused crawling technologies by integrating them with ontology learning technologies. The goal of ontology leaning is to semi-automatically extract facts or patterns from corpus or data and turn facts and patterns into machine-readable ontologies [6]. A few studies have been conducted in this field as follows:

Zheng et al. [7] proposed a supervised ontology-learning-based focused crawler that aims to maintain the harvest rate of the crawler in the crawling process.

The main idea of this crawler is to construct an artificial neural network (ANN) model to determine the relatedness between a Web page and an ontology.

Su et al. [8] proposed an unsupervised ontology-learning-based focused crawler in order to compute the relevance scores between topics and Web pages. Given a specific domain ontology and a topic represented by a concept in this ontology, the relevance score between a Web page and the topic is the weighted sum of the occurrence frequencies of all the concepts of the ontology in the Web page. This crawler makes use of reinforcement learning, which is a probabilistic framework for learning optimal decision making from rewards or punishments [9], in order to train the weight of each concept. The learning step follows an unsupervised paradigm which uses the crawler to download a number of Web pages and learn statistics based on these Web pages.

From the above survey, we found that none of the two crawlers is able to really evolve ontologies by enriching their contents, namely their vocabularies. When numerous unpredictable new terms outside the scope of the vocabulary of an ontology emerge in Web pages, these approaches cannot determine the relatedness between the new terms and the topic, and cannot make use of the new terms for the relatedness determination, which could result in the decline in their performance.

3 System Functions and Framework

The proposed ontology-learning-based focused crawler primarily consists of three components based on the functionalities, i.e., a storage component - the *service knowledge base*, a processing component - the *crawling and processing* module, and a computing component - the *service advertising information classification and ontology learning* module (Fig. 1).

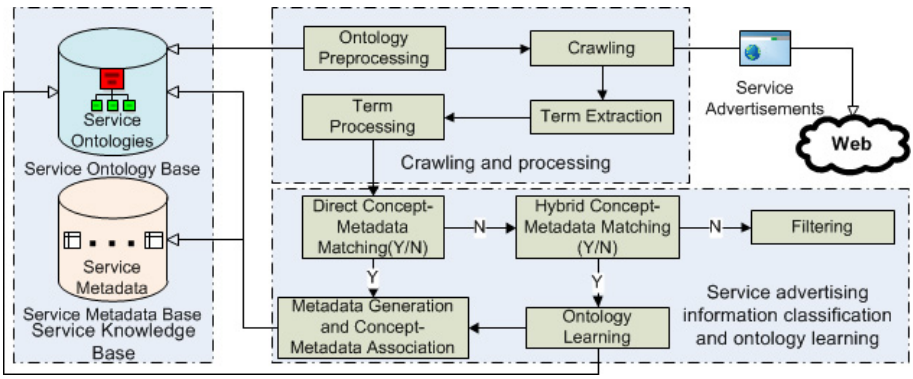

Fig. 1. Framework of the proposed ontology-learning-based focused crawler

3.1 Service Knowledge Base

The *service knowledge base* consists of a *service ontology base* and a *service metadata base*. The former is designed with the purpose of storing the machine-readable representation of domain-specific service knowledge, i.e., service ontologies. The latter is used to store semantically annotated service advertising information, i.e., service metadata.

For the service ontologies stored in the *service ontology base*, it is reasonable to make use of hierarchical ontologies for service advertising information discovery and classification, in which service concepts are linked by the *class/subclass* relationship. We define that each service concept contains the following elementary properties:

- A *conceptDescription* property is used to store the textual descriptions of a service concept, which consists of one or more phrases or sentences. Each phrase or sentence is a description or definition of a service concept, which is defined by domain experts. This property will be used in the process of service metadata classification (Section 3.2).
- A *learnedConceptDescription* property has a similar purpose to the *conceptDescription* property, which is automatically learned from service advertising information through the proposed crawler (Section 3.2).
- A *linkedMetadata* property is used to associate a service concept and a relevant service metadata. This property is used to classify and filter the generated service metadata by means of the concepts in a service ontology.

A service metadata is the semantic descriptions of a service entity, which consists of the following elementary properties:

- A *serviceDescription* property stores the textual description, e.g., a phrase or a sentence, regarding a service entity. The content of this property is automatically extracted from the crawled service advertising information by the proposed crawler (Section 3.2).
- A *linkedConcept* property is the inverse property of the *linkedMetadata* property. This property stores the URIs of the relevant service concepts of the service metadata. The service metadata and the service concepts can have a many-to-many relationship.

3.2 System Workflow of the Modules

In this section, we introduce the functionalities of the *crawling and processing* module and the *service advertising information classification and ontology learning* module in Fig. 1.

The *crawling and processing* module is designed for crawling service advertising information and processing the contents of the downloaded information and service ontologies for forthcoming computation. The first process in this module is *preprocessing*, which processes the contents of the *conceptDescription* property of each concept in a service ontology, before the crawler starts crawling. This process is realized by using Java WordNet Library[1] to implement tokenization,

[1] `http://sourceforge.net/projects/jwordnet/`

part-of-speech tagging, nonsense word filtering, stemming, synonym searching, and term weighting. The term weighting is to measure the particularity of each term in the service ontology. Here we make use of the inverse document frequency (IDF) model for the weight calculation. For a term (t) in a concept description ($CD_{j,h}$) of a service ontology (O), the weight of the term is

$$W(t) = log \frac{\{|C||\forall C \in O\}}{\{|C_\alpha||[t \in \delta(CD_\beta)] \bigcap (\exists CD_\beta \in C_\alpha) \bigcap (\forall C_\alpha \in O)\}} \quad (1)$$

where $|C|$ is the number of concepts in the ontology, $|C_\alpha|$ is the number of the concepts that contain the term, and $\delta(CD_\beta)$ is the set of sysnoyms of the terms in a concept description.

The missions of the *crawling* and *term extraction* processes are to download a service advertisement from the Internet at one time, and to extract the required service advertising information from the downloaded advertisement, according to the service metadata schema defined in Section 3.1, in order to prepare the properties to generate service metadata and service provider metadata. These two processes are realized by the semantic focused crawlers designed in our previous work [3], [4], in which the extraction rules and templates are defined by observing common patterns in HTML codes.

The *term processing* process is to process the contents of the *serviceDescription* property of the service metadata, which is similar to the *preprocessing* process. The major difference is that the former does not need the function of synonym searching. Similarly, the terms in the *serviceDescription* property also need a weight to indicate their particularity. Here, a term matching function is designed for passing the weights of ontological terms obtained in the *preprocessing* process, in order to reduce the computing cost in this real-time process. If no term in a service ontology matches a term in the *serviceDescription* property, the term will be regarded as a new term and assigned the maximum valid weight for its particularity, i.e., *log (number of concepts in the ontology)*.

The procedure of the *service advertising information classification and ontology learning* module is described as follows: first, the *direct string matching* process examines whether or not the content of the *serviceDescription* property of a service metadata is included in the *conceptDescription* and *learnedConceptDescription* properties of a service concept. If the answer is yes, then the concept and the metadata are considered as relevant. By means of the *etadata generation and association* process, the metadata can then be generated and stored in the *service metadata base* as well as associated to the concept. If the answer is no, a *hybrid concept-metadata matching* process will be invoked to check the relatedness between the metadata and the concept (Section 4). If the *serviceDescription* property of the metadata is related to any phrases in the *conceptDescription* property of the concept, the metadata and the concept are considered as relevant, and the contents of the *serviceDescription* property of the metadata can be regarded as a new phrase for the *learnedConceptDescription* property of the concept; otherwise the metadata is deemed as non-relevant to the concept. The above process is repeated until all the concepts in the service

ontology are compared to the metadata. If none of the concepts is relevant to the metadata, this metadata is then regarded as non-relevant to the service domain represented by the ontology and will be filtered.

4 Hybrid Concept-Metadata Matching Models

In this *hybrid concept-metadata matching* process, the extents of relatedness between the service description and the concept descriptions are assessed by a text-based concept-metadata matching (TCM) model and a probability-based concept-metadata matching (PCM) model. The results of the two models are then aggregated by a trained support vector machine (SVM) model. The eventual output of the hybrid model is the binary relatedness (relevant/non-relevant) between the service description and the concept descriptions.

The key idea of the TCM model is to measure the text similarity between a concept description of a service concept ($CD_{j,h}$) and a service description (SD_i) of a service metadata, by means of a weighted Dice's coefficient model and WordNet. The weighted Dice's coefficient model is mathematically expressed as follows:

$$sim_T(CD_{j,h}, SD_i) = \frac{\sum_{\forall u \in SD_i : u \in \delta(CD_{j,h})} w(u) + \sum_{\forall v \in CD_{j,h} : \exists \delta(v) \in SD_i} w(v)}{\sum_{\forall s \in SD_i} w(s) + \sum_{\forall t \in CD_{j,h}} w(t)} \tag{2}$$

where $\delta(CD_{j,h})$is the set of synonyms of the terms in the concept description, and $\delta(v)$ is the set of synonyms of a single term in the concept description.

The PCM model is a complementary solution for measuring the relevance between a concept description ($CD_{j,h}$) and a service description (SD_i) by measuring their co-occurrence frequencies in the crawled service advertisements, based on a probabilistic model. The PCM model is mathematically expressed as follows:

$$\begin{aligned} maxSim_P(CD_{j,h}, SD_i) &= \max_{CD_{j,\theta} \in C_j} [P(CD_{j,\theta}|CD_{j,h}) \cdot P(CD_{j,\theta}|SD_i)] \\ &= \max_{CD_{j,\theta} \in C_j} [\frac{n_{j,h}^{j,\theta}}{n_{j,h}} \cdot \frac{n_i^{j,\theta}}{n_i}] \end{aligned} \tag{3}$$

where $CD_{j,\theta}$ is a concept description of C_j, $n_{j,h}^{j,\theta}$ is the number of service advertisements that contain both $CD_{j,\theta}$ and $CD_{j,h}$, $n_{j,h}$ is the number of service advertisements that contain $CD_{j,h}$, $n_i^{j,\theta}$ is the number of service advertisements that contain both $CD_{j,\theta}$ and SD_i, and n_i is the number of service advertisements that contain SD_i.

The SVM classifier for each concept is designed to best aggregate the results of TCM model and the PCM model in order to decide on the semantic relatedness between a concept description and a service description, through a supervised training paradigm. This classifier provides a binary classification function (*relevant/non-relevant*), which is characterized by a hyperplane in a given feature space. For more details on SVM, we refer interested readers to the examples in [10].

5 System Implementation and Evaluation

We implement a prototype of the proposed ontology-learning-based focused crawler, and compare the performance of this crawler with the existing work reviewed in Section 2, i.e., Zheng et al.'s and Su et al.'s crawler, in the context of service advertising information discovery and classification, based on harvest rate, precision, and recall.

The overall framework of this crawler is built in Java within the platform of Eclipse 3.7.1[2]. The libSVM[3] Java library is utilized for the implementation of the SVM classifiers. For the purpose of comparatively analyzing our work and the two crawlers, we implement a prototype for each crawler in Java, in which the ANN model used by Zheng et al.'s crawler is built in Encog[4]. Next, we use a previously designed transport service ontology, which represents the domain knowledge in the transport service domain. The details of the transport service ontology can be referenced from [3].

In order to evaluate our crawler and the two crawlers in an open and heterogeneous Web environment, we choose two mainstream transport service advertising websites - Australian Yellowpages[5] (abbreviated as Yellowpages below) and Australian Kompass[6] (abbreviated as Kompass below), as the experimental data source. There are around 4400 downloadable transport-related service or product advertisements registered in Yellowpages, and around 10000 similar advertisements registered in Kompass, all of which are published in English. Since Zheng et al.'s crawler and the proposed crawler both need a supervised training process, and Su et al.'s crawler needs an unsupervised training process, we label the advertisements from Yellowpages, and use these advertisements as the training data set for all of these crawlers. Subsequently, we use the unlabeled advertisements from Kompass as the test data source.

The performance of the proposed crawler, Su et al.'s crawler, Zheng et al.'s crawler on the metrics of harvest rate, precision, and recall is shown in Table 1. Since Zheng et al.'s crawler does not have the function of classification, we only obtain its performance data on harvest rate.

Table 1. Overall performance of the ontology-learning-based focused crawlers

	Proposed crawler	*Su et al.'s crawler*	*Zheng et al.'s crawler*
Harvest rate	18.00%	6.80%	40.80%
Precision	88.03%	50.51%	*N/A*
Recall	55.55%	23.72%	*N/A*

It can be seen that the proposed crawler outperforms Su et al.'s crawler on all of the three parameters, only falling behind Zheng et al.'s crawler on the harvest

[2] http://www.eclipse.org/
[3] http://www.csie.ntu.edu.tw/~cjlin/libsvm/
[4] http://code.google.com/p/encog-java/
[5] http://www.yellowpages.com.au/
[6] http://au.kompass.com/

rate. However, harvest rate only concerns the capability of crawling Web pages but not the capability of crawling *right* Web pages. It is found that only around 25% of advertisements (Web pages) in the test data source are real transport-service-related advertisements. The harvest rate of the proposed crawler is closer to this ratio than the other two crawlers, which can partly prove the capability of the proposed crawler on crawling *right* service advertising information in a heterogeneous environment.

6 Conlusion

In conclusion, in the above experiments the proposed ontology-learning-based focused crawler shows the competitive performance, in comparison with the existing research work, in a simulated heterogeneous Web environment. This test primarily proves the feasibility of the proposed crawling framework for service advertising information discovery and classification.

References

1. Wang, H., Lee, M.K.O., Wang, C.: Consumer privacy concerns about Internet marketing. Commun. ACM 41, 63–70 (1998)
2. Dong, H., Hussain, F.K., Chang, E.: A service search engine for the industrial digital ecosystems. IEEE Trans. Ind. Electron. 58, 2183–2196 (2011)
3. Dong, H., Hussain, F.K.: Focused crawling for automatic service discovery, annotation, and classification in industrial digital ecosystems. IEEE Trans. Ind. Electron. 58, 2106–2116 (2011)
4. Dong, H., Hussain, F.K., Chang, E.: A framework for discovering and classifying ubiquitous services in digital health ecosystems. J. of Comput. and Syst. Sci. 77, 687–704 (2011)
5. Dong, H., Hussain, F.K., Chang, E.: State of the Art in Semantic Focused Crawlers. In: Gervasi, O., Taniar, D., Murgante, B., Laganà, A., Mun, Y., Gavrilova, M.L. (eds.) ICCSA 2009, Part II. LNCS, vol. 5593, pp. 910–924. Springer, Heidelberg (2009)
6. Wong, W., Liu, W., Bennamoun, M.: Ontology learning from text: A look back and into the future. ACM Computing Surveys X (2011) (to appear)
7. Zheng, H.-T., Kang, B.-Y., Kim, H.-G.: An ontology-based approach to learnable focused crawling. Inform. Sciences 178, 4512–4522 (2008)
8. Su, C., Gao, Y., Yang, J., Luo, B.: An efficient adaptive focused crawler based on ontology learning. In: Proceedings of the Fifth Int. Conf. on Hybrid Intelligent Syst. (HIS 2005), pp. 73–78. IEEE Computer Society, Rio de Janeiro (2005)
9. Rennie, J., McCallum, A.: Using reinforcement learning to spider the Web efficiently. In: Bratko, I., Dzeroski, S. (eds.) Proceedings of the Sixteenth Int. Conf. on Mach. Learning (ICML 1999), pp. 335–343. Morgan Kaufmann Publishers Inc., Bled (1999)
10. Boser, B.E., Guyon, I.M., Vapnik, V.N.: A training algorithm for optimal margin classifiers. In: Proceedings of the Fifth Annual Workshop on Computational Learning Theory, pp. 144–152. ACM, Pittsburgh (1992)