Harvesting Organization Linked Data from the Web

Zhongguang Zheng, Yingju Xia, Lu Fang, Yao Meng and Jun Sun
Fujitsu R&D Center CO., LTD., Beijing, China

Keywords: Linked Data, Web, Property Value Extraction.

Abstract: In this paper, we describe our approach of automatically extracting property-value pairs from the Web for organizations when only the name and address information are known. In order to explore the enormous knowledge from the Web, we first retrieve the Web pages containing organization properties by search engine, and then automatically extract the property-value pairs regardless of heterogeneous Web page structures. Our method does not require any training data or human-made template. We have constructed an organization knowledge base containing 3 million entities extracted from the Web for 4.2 million organizations which only have name and address information. The experiment shows that our approach makes it possible and effective for people to construct their own knowledge base.

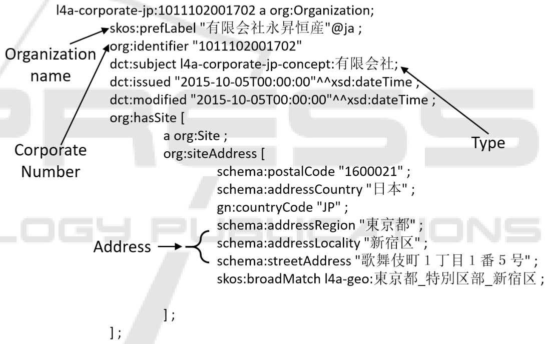
1 INTRODUCTION

Nowadays knowledge base (KB) plays an important role in many applications such as question answering, semantic disambiguation and information retrieval. There are some well-known KBs such as DBpedia and Yago which contain millions of entities describing different concepts, such as persons, places, organizations, etc. However, the quantity is still insufficient under certain contexts.

For instance, we have a task to build a KB for all the Japanese organizations. According to the data set published by the government[1], there are about 4.2 million organizations registered in Japan. The released data set mainly contains four types of information: the organization's names, types, addresses and corporate numbers. Figure 1 shows an example of the official data.

It is known that DBpedia contains only 0.24 million organizations in the English version[2] and much less organizations in the Japanese version. Similarly, other KBs fail to provide enough data to cover the 4.2 million Japanese organizations. As a result, it is impossible for us to meet the requirements with those existing KBs. Similar problem could be posed when people need to construct KBs for products, persons or other things.

Considering that a huge amount of knowledge in not tapped on the Web, it is sensible to extract the information from the Web (WebIE). A lot of research has been done to build effective methods for WebIE in the past years. However, most of them focused on learning wrappers from certain Web pages (training set), and afterwards they use the well-tuned wrapper to analyze other Web pages that have similar structures of the training set. Such methods generate reliable results when the target data is covered in only several similar websites.



```
l4a-corporate-jp:1011102001702 a org:Organization;
        skos:prefLabel "有限会社永昇恒産"@ja ;
        org:identifier "1011102001702"
        dct:subject l4a-corporate-jp-concept:有限会社;
        dct:issued "2015-10-05T00:00:00"^^xsd:dateTime ;
        dct:modified "2015-10-05T00:00:00"^^xsd:dateTime ;
        org:hasSite [
                a org:Site ;
                org:siteAddress [
                        schema:postalCode "1600021" ;
                        schema:addressCountry "日本" ;
                        gn:countryCode "JP" ;
                        schema:addressRegion "東京都" ;
                        schema:addressLocality "新宿区" ;
                        schema:streetAddress "歌舞伎町 1 丁目 1 番 5 号" ;
                        skos:broadMatch l4a-geo:東京都_特別区部_新宿区 ;

                ] ;
        ] ;
```

Figure 1: Example of the Japanese organization entity.

In our task, however, the detail information of one organization is likely available in its homepage. Furthermore, the structures of different homepages are in a wide variety of forms. Thus it is difficult to learn wrappers when the Web pages have little in common. In order to solve the above problems, we propose a novel method that features two points significantly different from the previous work.

- We retrieve the information of the organizations

[1] http://www.houjin-bangou.nta.go.jp/download/zenken/

[2] http://wiki.dbpedia.org/about

Zheng, Z., Xia, Y., Fang, L., Meng, Y. and Sun, J.
Harvesting Organization Linked Data from the Web.
DOI: 10.5220/0006888101590166
In *Proceedings of the 10th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2018) - Volume 2: KEOD*, pages 159-166
ISBN: 978-989-758-330-8

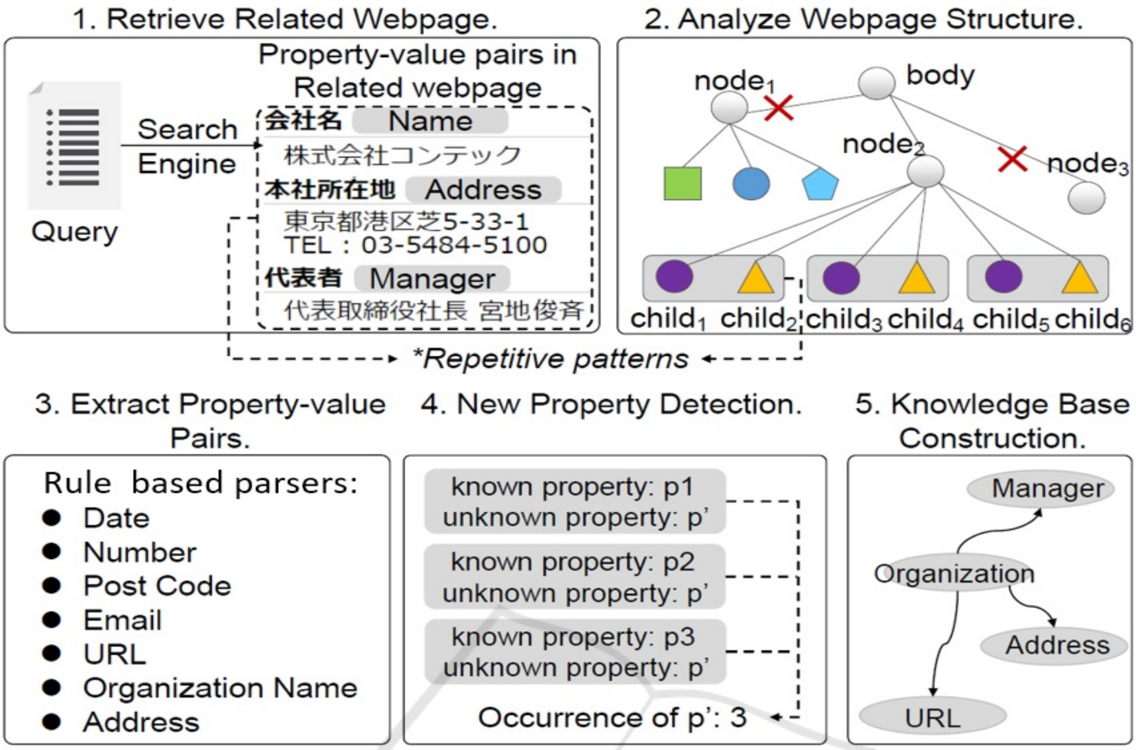


Figure 2: Overview of our method.

through the search engine. If one organization has its own homepage, the search engine is about to retrieve the homepage according to the proper queries, and otherwise, the search engine would retrieve other websites that probably contain the information of that organization which ensures the coverage of our method. In the experiment, we extracted about 3 million organization entities and 2.2 million of them are matched with the official data.

- Considering that the Web pages retrieved by the search engine are heterogeneous, it is difficult to learn wrappers to find common Web page structures. In order to solve this problem, we propose an algorithm to analyze the Web page dynamically without any training process or human-made templates.

The rest of this paper is organized as follows. Section 2 introduces the related work. In Section 3, we describe our method. Experiment is shown in Section 4, followed by a discussion in Section 5. The conclusion and future work are presented in Section 6.

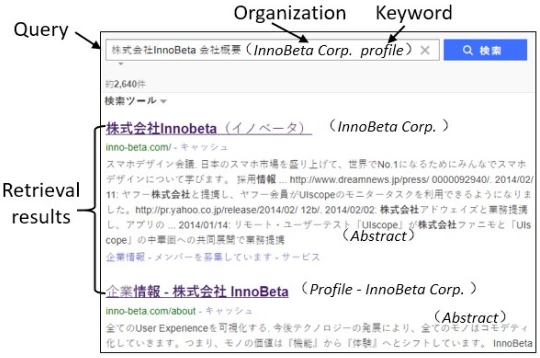
2 RELATED WORK

Conventionally we call the programmatical data extraction from the Web page "*wrapper*". There has been studies on developing various wrappers in the past years.

Wrapper Induction. Wrapper induction is to learn the structure of Web pages so that the wrapper is used to analyze other Web pages and the information is extracted accordingly. Some early studies focused on developing wrappers based on the priori knowledge (e.g., labeled data set, templates or heuristics). Kushmerick (1997) builds wrappers from a set of labeled example Web pages. J. Hammer & Crespo (1997) develops a tool for extracting semistructured data from Web pages which needs human configuration. In order to reduce human work, recent studies learn wrappers automatically from Web pages aiming to discover common structures. V. Crescenzi & Merialdo (2001) develops wrappers by evaluating the similarities and differences between Web pages without any priori knowledge. Freitag & Kushmerick (2000) and Carlson & Schafer (2008) adopts boost learning method to learn wrappers. Wong & Lam (2010) presented a Bayesian learning framework trained from source Web pages for new unseen Web pages. N. Dalvi & Sha (2009) and Q. Hao & Zhang (2011) adopt other statistical models for wrapper induction. Those supervised methods still depend on manually labeled data set to train the wrappers. Moreover, the inducted wrappers fail to produce satisfactory results when dealing with very different structured Web pages. This makes the effectiveness of the wrapper limited to a fraction of websites. Anna Lisa Gentile & Ciravegna (2013) proposes a method for learning wrappers without any training materials. However, their method depends on existing Linked Data knowledge bases (e.g., DBpeida) to match the content in the Web page, and also induces wrappers from homogeneous Web pages. In our task, there is no such existing knowledge base that covers enough organization entities and our goal is to analyze any given Web pages.

Wrapper for Unstructured Web Pages. There has been studies focused on unstructured information extraction. Szekely & Craig A. Knoblock (2015) describes a system that constructs knowledge graph by analyzing online advertisement to fight the human trafficking. They first adopt Apache Nutch for the data acquisition, and then use their DIG system to extract the properties. The DIG system is a semi-structured page extractor that identifies elements based on several regular expressions. The system needs a small size labeled data set for training the property extractor. Michele Banko (2007) introduces an open information extraction method that extracts relation tuples from the Web corpus. Aliaksandr Talaika & Suchanek (2015) proposes an extraction method of the unique identifiers and names for products from the Web pages. Their method is independent of the Web page structure. Although those methods are workable to handle any given Web page and are free of training set (or need a small set) since they are insensitive to the Web page structure, their methods are confined to either extraction scope in the Web page or the property numbers of the target data. Szekely & Craig A. Knoblock (2015) focuses on the sentences containing suspicious information while Aliaksandr Talaika & Suchanek (2015) is only interested in the identifier and the product name. A lot of information on the Web is organized in the tabular or list layout. Their methods fail to handle such complex Web structures. As a result they can not extract multiple property-value pairs from the Web page.

In this paper, we propose a novel approach that efficiently extracts property-value pairs for organizations from the Web pages. Our method differs from the previous studies in that we automatically detect the area that containing useful information in the Web page without any training process. Moreover, the goalof our method is to extract all the property-value pairs describing the same organization entity from one Web page. Our method is scalable for extracting information for other kinds of entities.



株式会社InnoBeta 会社概要（InnoBeta Corp. profile）

約2,640件

株式会社Innobeta（イノベータ）

inno-beta.com/ - キャッシュ

スマホデザイン会議. 日本のスマホ市場を盛り上げて、世界でNo.1になるためにみんなでスマホデザインについて学びます。 採用情報 ... http://www.dreamnews.jp/press/ 0000092940/. 2014/02/11: ヤフー株式会社と提携し、ヤフー会員がUIscopeのモニタータスクを利用できるようになりました。http://pr.yahoo.co.jp/release/2014/02/ 12b/. 2014/02/02: 株式会社アドウェイズと業務提携し、アプリの ... 2014/01/14: リモート・ユーザーテスト「UIscope」が株式会社ファニモと「UIscope」の中華圏への共同展開で業務提携

企業情報 - メンバーを募集しています - サービス

企業情報 - 株式会社 InnoBeta

inno-beta.com/about - キャッシュ

全てのUser Experienceを可視化する. 今後テクノロジーの発展により、全てのモノはコモデティ化していきます。つまり、モノの価値は『機能』から『体験』へとシフトしています。 InnoBeta

Figure 3: Example of retrieved results.

3 APPROACH

3.1 Overview

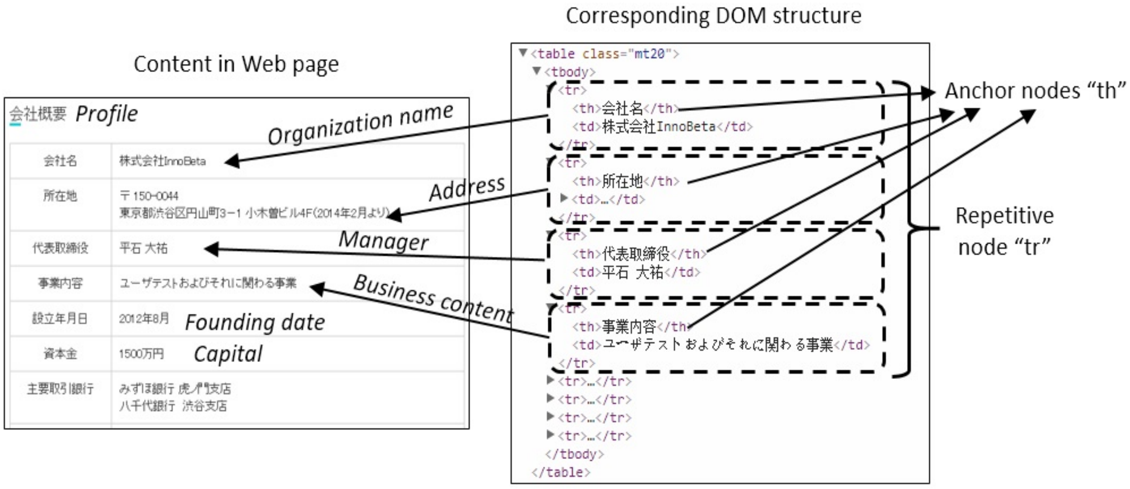
Our goal is to find the Web page which contains the information of the given organization, and then extract all the property-values pairs from the page. In order to find the Web page related to the given organization, we adopt the search engine as a helper. After retrieving the Web pages, our wrapper will automatically analyze the pages and extract all the property-value pairs by some rule-based parsers, and afterwards we will discover new properties so as to complement the extraction results. The whole process is depicted in Figure 2.

3.2 Related Web Pages Retrieval

Since there is no available knowledge base which contains enough information for our task, we turn to the search engine to locate the Web pages containing the organization information, e.g., the homepage. In order to make the retrieval result as precise as possible, we construct each query with the organization names, cities and other keywords such like “会社概要(profile)”. In this paper, Yahoo! Japan[3] is selected as our final search engine.

Figure 3 shows an example of the query “株式会社InnoBeta (InnoBeta Corporation), 会社概要(profile)”. The retrieval result contains about 10 records per page and the top 2 records are listed in Figure 3. Conventionally the topmost record should be considered as the closest result to the query. However, when examining the related Web pages, we find that sometimes the search engine fails to return satisfactory results by using the same set of keywords

[3] yahoo.jp



会社概要

会社名	株式会社InnoBeta
所在地	〒150-0044 東京都渋谷区円山町3−1 小木曽ビル4F(2014年2月より)
代表取締役	平石 大祐
事業内容	ユーザテストおよびそれに関わる事業
設立年月日	2012年8月
資本金	1500万円
主要取引銀行	みずほ銀行 虎ノ門支店 八千代銀行 渋谷支店

```
▼<table class="mt20">
  ▼<tbody>
    ▼<tr>
        <th>会社名</th>
        <td>株式会社InnoBeta</td>
     </tr>
    ▼<tr>
        <th>所在地</th>
      ▶<td>…</td>
     </tr>
    ▼<tr>
        <th>代表取締役</th>
        <td>平石 大祐</td>
     </tr>
    ▼<tr>
        <th>事業内容</th>
        <td>ユーザテストおよびそれに関わる事業</td>
     </tr>
    ▶<tr>…</tr>
    ▶<tr>…</tr>
    ▶<tr>…</tr>
    ▶<tr>…</tr>
    </tbody>
  </table>
```

Figure 4: Example of property-value pairs.

since different Web pages may contain different keywords.

Thus we create a keyword list by hand denoted as $l = [(k_0, p_0), (k_1, p_1), ..., (k_n, p_n)]$, where k_i denotes a keyword, such as "会社概要(prfile)", "プロフィール(profile)", "社名(name)" and "本社住所(address)", and p_i denotes the priority of the keyword which is assigned manually. We use an iterative mechanism to retrieve the related Web pages depicted as follows.

```
sort l by priority
i = 0
while iter < max_iter_num:
     select l[i,i+n] as keywords
     construct query with the keywords
     call the search engine
     extract information from retrieved results
     if success
          go to the next query
     else
          iter += 1
          i += n
```

Where l denotes the keyword list. For each organization name, we define *max_iter_num* as the maximum iteration number when the search engine fails to retrieve satisfactory pages.

3.3 Web Page Analysis

3.3.1 Anchor Nodes Detection

Figure 4 shows an example of the tabular structure in the Web page containing property-value pairs of the organization "株式会社InnoBeta (InnoBeta Corporation)" together with its corresponding DOM tree structure. First, we identify all the leaf nodes which contain property names. Thus we create a property name dictionary manually beforehand. Note that the expressions of the same property name are unpredictable and vary a lot in different Web pages. For instance, the property "organization name" may be in terms of "企业名", "商号" or "名称". Besides, it is impossible to cover all types of properties in advance. As a result, it is difficult to create a dictionary which covers all the expressions or properties. Therefore we propose a mechanism introduced in Section 3.4 to enlarge the dictionary when extracting property-value pairs simultaneously.

For each leaf node in the DOM tree, if the text of a node n_i is matched in the dictionary, we consider n_i as an *anchor node*. Those anchor nodes roughly locate the positions of the property-value pairs.

3.3.2 Property-value Pair Extraction

After the observation of many Web pages, we find that despite Web pages have heterogeneous HTML structures, the useful information is always displayed in well-organized layouts such as table or list. As shown in Figure 4, the property-value pairs are located in the node "*table*". Furthermore, each property-value pair is under the node "*tr*". In this paper, we call nodessuch as "*table*" *root nodes*, and its descendant nodes such as "*tr*" *pattern nodes*. Note that the "table or list" layouts may be composed of any HTML elements, such as "*table*", "*div*", "*span*", "*ul*", etc. Thus it is impossible to use templates to predict the location of property-value pairs.

The pattern nodes have two characteristics. Firstly, they cover anchor nodes. Secondly, they repetitively appear under one *root nodes*. These two characteristics ensure that each property-value pair is covered in just one pattern node, e.g., each "*tr*" node covers only one property-value pair in Figure 4. This

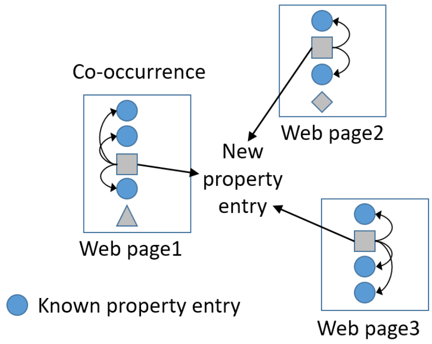


Figure 5: New property name detection.

is an import clue that helps us to confine the boundary of the property value. As the example in Figure 4, the value of property “会社名(name)” can be only extracted from node “*td*” which is the child node of the first pattern node “*tr*”. Hence our problem is to find those pattern nodes. As mentioned above, the pattern nodes may be composed of multiple nodes which could be any HTML element, we propose a novel method to detect pattern nodes automatically.

Root Node Detection. In order to find the pattern nodes, we first find the root node. If the root is detected, we could extract pattern nodes from its descendant nodes.

Given a Web page, we first resolve the Web page into its DOM tree structure. Then we preserve all the anchor nodes from the leaf nodes. For each anchor node, we define its pattern as $pat\{tag, class\}$, where tag is the DOM node type such as “*div*” or “*table*”, while $class$ is the class attribute of that node. For example, the node “*<table class=‘mt20’>...</table>*” in Figure 4 has the pattern $pat\{\text{‘}table\text{’}, \text{‘}mt20\text{’}\}$. For each non-leaf node, we select a root node $node_{root}$ when

- The node covers all the anchor nodes which have the same pat.
- There is no other root nodes in its descendants.

Pattern Node Detection. For each $node_{root}$, if it has more than n children, we will extract pattern nodes from its children denoted as $patset[subpat_{1,l},...,subpat_{n,1}]$, where $subpat_{i,l}\{pat_i, nodeset\}$ presents node sets that have the same pat_i and l is the length of pat_i, while $nodeset[nodes_1,...,nodes_m]$ (m>$minmum\_repeat\_count$) presents the node sets. $minmum\_repeat\_count$ is a constraint that depicts the minimum occurrence of pat_i. Each $nodes_i$ contains DOM tree nodes $[child_1,...,child_l]$, where $child_i$ is a descendant node of $node_{root}$. For one $subpat_{i,l}\{pat_i, nodeset\}$, if $l \geq valid\_pattern\_length$, we will consider $pat_{i,l}$ as a valid pattern and then extract property-value pairs from $nodeset$.

As the example in Figure 4, the node “*table*” is not a root node since it has only one child “*tbody*”. The node “*tbody*” is a root node since we can extract $patset$ from its eight children nodes “*tr*”, that is $patset[subpat\{tr, nodeset[0,1,...,7]\}]$, and then we can extract property-value pairs from each “*tr*” node.

Rule based Value Parser. Parsing values from the free text is task specific. In our task, we developed several rule-based parsers (shown in Figure 2) for different types of organization properties such as date, number, address and organization name, etc. Note that the previous procedure depicted in this section is independent of subjects or languages (except the handcrafted dictionary).

There is a problem that the extracted property-value pairs may describe multiple organizations. In this paper, we simply split the property-value pairs into different groups ensuring that each group has only one property representing the organization's name.

Entity Linkage. After extracting the property-value pairs, we will link the pairs with the official data set. Since the organization name and address are already known (see Figure 3), we can link extracted property-value pairs with an organization entity by matching their name and address properties. Those property-value pairs failed to match the source organizations will be grouped by matching their name and address properties with each other, and the grouped pairs will form new organization entities.

3.4 New Property Name Detection

As mentioned in Section 3.3.1, the handcraft dictionary fails to cover all the properties on the Web. Some unknown properties may occur when handling different Web pages. Moreover, even a known property may have various expressions in different Web pages, for example, property “会社名(organization name)” has other synonymous expressions like “企业名”,“商号”, “名称”, etc. It is an important work to discover more properties and expressions to enlarge the dictionary.

Initially we craft the dictionary with some basic properties, such as “社名(organization name)”, “住址(address)”, “取缔役(manager)”, etc. Then during the step described in Section 3.3.2, for $nodes_i$ $[child_1,...,child_l]$, if $child_h$ is in the property dictionary, we will record its pattern $path_h$. If there is

Table 1: Overall experimental result.

Source entity	Extracted entity	Linked entity
4.27M	2.99M	2.33M (77.9%)

a node $child_k$ in $nodes_j$ $[child_1, ..., child_l]$ ($nodes_i \in nodeset_m, nodes_j \in nodeset_m$) has the same pattern with $path_h$ but $text(child_k)$ is not in the dictionary, we will count the occurrence of $text(child_k)$. A new property will be discovered if its occurrence reaches a threshold.

This process is shown in Figure 5, where the circle nodes are the known properties. We consider the rectangle node as the new property when it co-occurs with the known properties and they have the same pattern. We have discovered more than 240 new property entries comparing with the initial dictionary that only contains about 120 property entries. Finally we can construct our organization knowledge base with those extracted property-value pairs.

Table 2: Information of our organization KB.

Entity	Triple	Property
2.99M	31.22M	42

Table 3: Result of listed organizations.

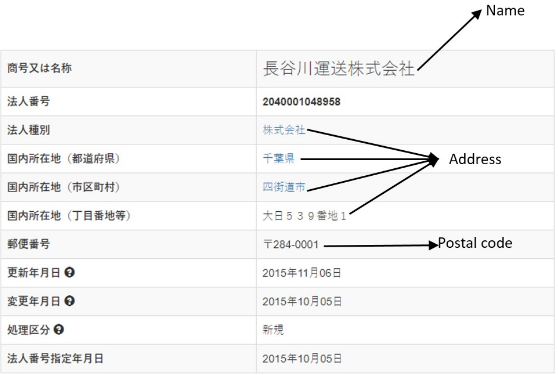
Source entity	Extracted entity	Linked entity
2642	2431	2431 (92%)

4 EXPERIMENT

In this paper, our task is to construct a knowledge base for the Japanese organizations. The source data released by the government contains about 4.2 million organizations (see Figure 1). We first develop a crawler to collect the related Web pages as described in Section 3.2, and then extract property-value pairs for the knowledge base construction by our proposed method.

Table 1 displays the overall result. We extract 2.99 million entities from all the Web pages, and 2.33 million (78%) of them are successfully linked to the source entities. Table 2 depicts the main information of our knowledge base which contains 31.22 million triples and 42 kinds of properties. The discussion of our result is shown in Section 5.

In order to further prove the effectiveness of our approach, we select a subset containing 2.6 thousand organizations listed in the Japanese stock market from the official data set. Considering that it is much easier to obtain the Web pages for the listed organizations, the influence of the search engine will be lowered. This makes it a better test set for our wrapper. Table 3 demonstrates the result on the listed organizations.



商号又は名称	長谷川運送株式会社
法人番号	2040001048958
法人種別	株式会社
国内所在地（都道府県）	千葉県
国内所在地（市区町村）	四街道市
国内所在地（丁目番地等）	大日５３９番地１
郵便番号	〒284-0001
更新年月日 ❓	2015年11月06日
変更年月日 ❓	2015年10月05日
処理区分 ❓	新規
法人番号指定年月日	2015年10月05日

Figure 6: Information from the third-party website.

We can see that 92% of the source entities are covered by our extracted entities. This convincing result reveals the effectiveness of our method.

We list a portion of properties with coverage greater than 5% and their corresponding entries in Table 4. "*Coverage*" presents the percentage of the extracted entities with the property. The discussion of this result is shown in the following section. Moreover, we randomly selected about 100 entities from the listed organizations as a sample which is available here[4]. People could trace the hyperlink of property "dct:source" to the original Web page[5].

5 DISCUSSION

Long Tail Data. In our experiment, there are still a large portion of source entities left without any extracted information. After studying some bad cases, we find that the search engine is set to retrieve unrelated Web pages for some unknown organizations. For example, the retrieval results of the organization "有限会社川壱食品" are all irrelevant Web pages. It is even difficult for human to find the related Web pages through search engine for those long tail data. This is contradictory against our initial thoughts that there would be enough data on the Web. It seems that many organizations do not have homepage or do not put their information available on the Web.

Property Coverage. As shown in Table 4, addresses, postal codes and telephone umbers are the most common properties. The coverage of other properties

[4] http://36.110.45.44:8081/sample.html

[5] Since the most Web pages were crawled in 2016 and 2017, the access to the original Web page may fail due to the possible change of the website.

Table 4: Properties with coverage greater than 5%.

Property	Dictionary entry	Coverage	Property	Dictionary entry	Coverage
Name	会社名 正式名称 店名 名称 事業所名 屋号 店名 事業者名称 企業名 法人名 商号	100%	Foundation date	創業 創立 設立 設立日 法人設立 会社設立 設立年月日 創業年月日	17.9%
Address	住所 本社住所 所在地 本店 所在 運営住所 会社所在地 本社所在地	95.1%	Scale	従業員数 社員数 従業員規模 正社員数	11.5%
Postcode Telephone	本社郵便番号 郵便番号 電話番号 代表電話番号	83%	Classification	主要業種 業種分類 サービス分野 種類 事業紹介	8.9%
Key person	社長 代表取締役 CEO 代表者 常務取締役 役員 理事長 執行役員社長 代表取締役会長 監査役 専務取締役 常務理事	27.5%	Homepage	homepage url サイト 会社hp ホームページ オフィシャル 企業サイト ウェブサイト 公式ホームページ	8%
Corporation number	法人番号 会社法人等番号	25.2%	Related bank	取引銀行 主な取引銀行 取引金融機関 主要取引銀行	6.6%
Fax	fax FAX番号	21.7%	Business hour	受付時間 営業時間	6%
Industry	営業内容 業務内容 事業内容 主要業務 事業 主な事業内容 会社事業内容 主な事業 事業分野	18.8%	Email	e-mail email Eメール メール メールアドレス	5.7%

drops drastically. That is because many entities are extracted from the third-party websites rather than the organization homepages, and the third-party websites are set to provide less information. Figure 6 shows an example from the third-party website. Ideally, the homepage would provide diverse information, however, as discussed above, many long tail organizations do not have homepages. As a result, the third-party websites become the only information resources.

Data Cleansing and Formalizing. It is known that the data on the Web is mostly noisy and informal. It will take a lot of effort to improve the data quality. Meanwhile, some sophisticated technologies should be involved, e.g., named entity recognition (NER). Though we have developed several basic parsers for formalizing certain properties, more parsers are needed to improve the data quality, such as person name or other newly detected properties.

Evaluation. Currently we are still improving our method according to the feedbacks of human check. In our experiment, the linkage rate between the extracted results and the official data set can be seen as the recall value. However, in order to thoroughly evaluate the precision of our method, a test set is needed, although it is not easy to set up.

6 CONCLUSION AND FUTURE WORK

In this paper, we introduced our approach to extract entity property-value pairs through the Web. We first utilize search engine to retrieve the related Web pages, and then we automatically analyze the Web page structure and extract property-value pairs. Furthermore, we studied the technology to discover unknown properties.

Our proposed approach makes it possible for people to build their own knowledge base from a fraction of data. In the future, we will improve our approach to make it robust for more complicated structured Web pages. Moreover, evaluation by precision is another important subject to explore in the future.

REFERENCES

Aliaksandr Talaika, Joanna Biega, A. A. & Suchanek, F. M. (2015), Ibex: Harvesting entities from the web using unique identifiers, *in* 'Proceedings of the 18th International Workshop on Web and Databases', pp. 13–19.

Anna Lisa Gentile, Ziqi Zhang, I. A. & Ciravegna, F. (2013), Unsupervised wrapper induction using linked data, *in* 'Proceedings of the seventh international conference on Knowledge capture', pp. 41–48.

Carlson, A. & Schafer, C. (2008), Bootstrapping information extraction from semi-structured web pages, *in* 'European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases'.

Freitag, D. & Kushmerick, N. (2000), Boosted wrapper induction, *in* 'NCAI'.

J. Hammer, H. Garcia-Molina, J. C. R. A. & Crespo, A. (1997), Extracting semistructured information from the web, *in* 'Proceedings of the Workshop on the Management of Semistructured Data'.

Kushmerick, N. (1997), Wrapper induction for information extraction, *in* 'IJCAI97', pp. 729–735.

Michele Banko, Michael J Cafarella, e. a. (2007), Open information extraction from the web, *in* 'Proceedings of the 20th international joint conference on Artifical intelligence', pp. 2670–2676.

N. Dalvi, P. B. & Sha, F. (2009), Robust web extraction: an approach based on a probabilistic tree-edit model, *in* 'Proceedings of the 35th SIGMOD international conference on Management of data'.

Q. Hao, R. Cai, Y. P. & Zhang, L. (2011), A unified solution for structured web data extraction, *in* 'SIGIR', pp. 775–784.

Szekely, P. & Craig A. Knoblock, e. a. (2015), Building and using a knowledge graph to combat human trafficking, *in* 'Proceedings of the 14th International Semantic Web Conference', Vol. 9367, pp. 205–221.

V. Crescenzi, G. M. & Merialdo, P. (2001), Roadrunner: Towards automatic data extraction from large web sites, *in* 'VLDB'.

Wong, T. & Lam, W. (2010), Learning to adapt web information extraction knowledge and discovering new attributes via a bayesian approach, *in* 'Knowledge and Data Engineering', Vol. 22, pp. 523–536.