CASIANED: People Attribute Extraction based on Information Extraction

Xianpei Han
National Laboratory of Pattern Recognition
Zhongguancun East Road 95
HaiDian District, Beijing, China, 100190
+86 010 82614468
xphan@nlpr.ia.ac.cn

Jun Zhao
National Laboratory of Pattern Recognition
Zhongguancun East Road 95
HaiDian District, Beijing, China, 100190
+86 010 82614505
jzhao@nlpr.ia.ac.cn

ABSTRACT
In this paper, we describe the people attribute extraction system of the CASIANED team for the second Web People search evaluation (WePS-2). We develop an attribute extraction system based on information extraction. Firstly the attribute candidates for every attribute class are extracted using several different information extraction techniques; then these candidates are verified through classification. The system achieves F-measure 0.309 on the develop dataset and F-measure 0.117 on the final test dataset.

Categories and Subject Descriptors
H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing; I.2.7 [**Artificial Intelligence**]: Natural Language Processing

General Terms
Algorithms, Experimentation

Keywords
Information Extraction, Attribute Extraction, Knowledge Acquisition, Knowledge Base Population

1. INTRODUCTION
The World Wide Web is a vast and rapidly growing repository of information. There are various kinds of objects, such as products, people, movies, etc., embedded in various web pages. However, such object information usually only human readable, and can only be accessed through keyword based search. Thus, extracting object information from the Web is of great importance for Web data management and information search, such as web person search and knowledge base population.

The goal of the Attribute Extract task, a subtask of Web People Search task (Artiles et al. 2007, Satoshi Sekine and Javier Artiles. 2009), is to extract 18 kinds of "attribute values" of target individuals whose names appear on each of the provided Web pages, including *Date of birth*, *Birth place*, *School*, etc. Here, the attribution extraction system receives Web pages associated with the given individual name, and outputs all attribute values within these Web pages.

The previous research in attribute extraction mainly focuses on product's attribute extraction. The product information usually embeds in dynamically generated Web pages. The product Web pages within the same web site usually are homogeneous, for example, all detailed web pages about book in Amazon are nearly the same structure. Thus, the Wrapper technique can be used to extract attribute in a single product web site with a few labeled instances (Yanhong Zhai, Bing Liu, 2005). Taken Web page as plain text, text mining technique also can be used in attribute extraction (Rayid Ghani, et al. 2006). However, compared with the plain text, Web pages usually contain rich additional structural information, which can be helpful in attribute extraction. The 2D CRF (Jun Zhu, et al. 2005) and Hierarchical CRF (Jun Zhu, et al. 2006) have been used to model the structure information of Web page in attribute extraction. Sujith Ravi and Marius Pasca detect attributes of a special kind entity using Web pages' structured text, such as the table header. The structure information can also be combined with text information. A graph model is designed to extract and normalize product attribute information from multi-web sites using both the structure and text information (Tak-Lam Wong et al. 2008).

In this paper, we aim at extract attributes about people in different Web pages. Unlike the Web pages about product, little structure homogeneity exists between different Web pages about the same individual, so the wrapper technique and the structure homogeneity based techniques are hard to apply. We develop an attribute extraction system based on information extraction. The attribute candidates for every attribute class are extracted firstly using several different information extraction techniques; then these candidates are verified through classification.

This paper proceeds as follows. Section 2 describes our proposed system in detail. Section 3 includes the performance results and discussions. This paper concludes with a review of summary and future directions.

2. Our Method
In this section, we describe our people attribute extraction system. A graphical diagram presenting our method is shown in Figure 1. The method is essentially composed of two function modules: the attribute candidate generation module and the attribute candidate verification module. For a web page, the attribute candidate generation module extracts all attribute candidates for every attribute class through recognizing different named entities and noun phases which can be used as attributes, then the attribute candidate verification module verifies these candidates through classification. A more detailed discussion for each module will be presented in the following subsections.

WWW 2009, April 20-24, 2009, Madrid, Spain.

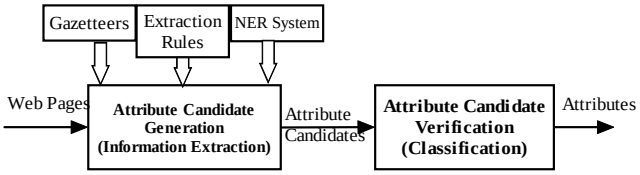

Figure 1. An abstract diagram presenting the main components of our method

2.1 Attribute Candidate Generation

The attribute candidate generation module extracts all attribute candidates in a given web page through information extraction.

For a given attribute class, the attribute value usually is noun phrase of some special types. For example, the value of *Date of birth* attribute must be date; the value of *Birth place* must be location, etc. We category the attribute classes of people into three different categories according to their value type, as shown in Table 1. For different attribute value types, special methods are used to extract attribute candidates.

Table 1. The Value Type of Different Attribute Class and the Corresponding Candidate Extraction Method

Value Type	Detailed Value Type	Attribute Class	Candidate Extraction Method
Traditional Named Entity	People Location Organization Date	Date of birth, Birth place, Other name, Affiliation, School, Mentor, Relatives	Named Entity Recognition tools
Special Type Named Entity	Email Phone Number URL	Award, Nationality, Phone, FAX, Email, Web site	Extraction Rule(Regular Expressions)
Special Type Noun phase	Occupation Degree Major Award Nationality	Occupation, Work, Major, Degree	Gazetteer based Matching

The first value type of some attribute classes is the traditional named entities (People, etc.), for example, the value type of *Date of birth* is date, the value type of *Affiliation*, *School* is organization, etc. The existing approach has been able to achieve good results in traditional named entity recognition (Nadeau, David et al. 2007). We extract the named entities from the web page using the OpenNLP[1] Named Entity Detection tools. Then the extracted named entities are considered to be the candidates of corresponding attribute classes.

The second value type of some attribute classes is the named entity of special types. Currently, there is no tool which can be used to extract such named entities instantly. Fortunately, there are some patterns existed on Email, Phone Number and URL. We can extract them using some extraction rules. In our system, we build several regular expressions to extract Email, Phone Number and URL from Web pages.

The third value type is noun phase of specific types. For example, the noun phase about occupation, such as professor, artist, actor, etc. We use a gazetteer based matching method to extract the attribute candidates for this value type. For each noun phase type, we build a gazetteer from the Wikipedia[2], the largest online encyclopedia which contains millions of concepts. For example, we build the occupation gazetteer through extracting all occupations listed in the *List of occupations* entry in Wikipedia. Based on these gazetteers, we can extract the attribute candidates by finding all word appearances of the corresponding gazetteer in Web pages.

After the attribute candidate generation, a list of attribute candidates is obtained for every attribute class.

2.2 Attribute Candidate Verification through Classification

The attribute candidates obtained using the method described in previous section need to be verified. This is because a candidate may be the attribute of other persons or not attribute at all. For example, a phone number candidate may not the phone of the target individual, or it is actually a Fax.

Our system verifies the attribute candidates through classification. For every attribute class, a classifier is trained to identify whether an attribute candidate is the real attribute of the target individual. For a given person and a given attribute class, the classifier classifies an attribute candidate into two targets: attribute or not attribute.

We build the training corpus for each attribute class's classifier as follows: firstly, we generate all attribute candidates of the given attribute class in the develop dataset, where an attribute candidate is represented as (value, context text); secondly, we label the attribute candidates whose value can match one true attribute as positive training instance, and all left candidates are taken as negative training instances.

We need to extract the representation of an attribute candidate as the input of the classification process. An attribute candidate is represented as a vector of features as follows.

Context Token. The context words near an attribute candidate usually provide helpful information. For example, the word *Fax* within *Fax +43 (1) 58801 – 18392* provides the evidence that *+43 (1) 58801 – 18392* is a Fax. We firstly extract the words within a window size 5 as context words, then stem them using the Porter stemmer[3], stop words are filtered. Each word retained is used as a feature.

Value Pattern. For a specific attribute class, the value usually shows some patterns. For example, some patterns existed in the *School* attribute value, such as *School_of_Location*, *Location_State_College*, etc. We extract the pattern features of value through the following steps: firstly we recognize and label the named entity within the attribute candidate, for *University of Cape Town* the label result is *University of <LOCATION>Cape Town<LOCATION>*; then we replace the named entity with the

[1] http://opennlp.sourceforge.net/

[2] http://www.wikipedia.org

[3] http://tartarus.org/~martin/PorterStemmer/

label, for the above example, the result after replacing is *University of LOCATION*; then we extract all the unigrams, bigrams and the total strings as pattern features, we also filter the features which are too general. For the above example, the extracted features are *University*, *University of*, *University of LOCATION*.

Dependency Path. We model the relation between the target individual and the attribute candidate by analyzing the dependency path between them. For example, one dependency path between *Christine Borgman* and the Occupation candidate *professor* is "*the **Christine Borgman**, a **professor** at UCLA*", which provides evidence about the *Christine Borgman* is a *professor*. We extract the features on dependency path through the following steps: firstly we label the dependency path with POS tag and NER tag; then we extract all unigrams, bigrams and trigrams, all grams extracted are used as features. We also add the number of named entity on the dependency path and the length of dependency path as features.

Using the features described above, we can classify whether an attribute candidate is the specific attribute class' value of the target individual. After the attribute candidate verification, all attribute candidates retained will be taken as real attributes.

3. Results and Discussions

To assess the performance of our people attribute extraction system, we apply it on both the develop dataset and the test dataset of the WePS-2 attribute extraction task. Each experiment uses Maxent classifier to verify the attribute candidates with the software package provided in OpenNLP Maxent[4].

In Table 2, we demonstrate our results on develop dataset. Our system can achieve 0.309 with average F-measure. In Table 3, we demonstrate the result on test dataset. Compared with the results on develop dataset, the performance declines significantly on average F-measure, from 30.9% to 11.7%.

Table 2. Results on Develop Dataset

Name	Precision	Recall	F-measure
Alexander Macomb	34.7	30.3	32.4
Allan Hanbury	42.2	31.8	36.2
Andrew Powell	25.1	14.5	18.4
Anita Coleman	35.6	28.8	31.9
Christine Borgman	41.2	25.1	31.2
David Lodge	40.5	34.5	37.2
Donna Harman	48.2	43.8	45.9
Edward Fox	52.6	34.6	41.7
George Clinton	35.6	29.6	32.3
Gregory Crane	32.7	33.0	32.9
Jane Hunter	24.5	26.2	25.3
John Kennedy	30.5	39.0	34.2
Michael Howard	37.8	25.2	30.3
Paul Clough	32.3	21.2	25.6
Paul Collins	25.5	20.8	22.9
Thomas Baker	31.8	23.9	27.3
Tony Abbott	19.8	18.4	19.1
Average	**34.7**	**28.3**	**30.9**

[4] http://maxent.sourceforge.net/

Table 3. Results on Test Dataset

Name	Precision	Recall	F-measure
Amanda Lentz	3.4	8.3	4.8
Benjamin Snyder	4.3	10.3	6.0
Bertram Brooker	12.2	22.7	15.9
Cheng Niu	3.4	22.2	5.9
David Tua	2.7	8.7	4.1
David Weir	12.5	18.5	14.9
Emily Bender	3.4	19.3	5.8
Franz Masereel	8.5	25.8	12.8
Gideon Mann	4.9	15.8	7.5
Hao Zhang	9.4	22.6	13.3
Helen Thomas	10.7	24.6	14.9
Herb Ritts	9.8	23.4	13.8
Hui Fang	7.2	19.9	10.5
Ivan Titov	2.5	13.6	4.2
James Patterson	10.7	29.0	15.7
Janelle Lee	1.6	5.0	2.4
Jason Hart	11.3	15.0	12.9
Jonathan Shaw	14.8	26.4	19.0
Judith Schwartz	9.2	23.0	13.1
Louis Lowe	5.8	16.3	8.5
Mike Robertson	12.5	19.6	15.3
Mirella Lapata	5.8	30	9.8
Nicholas Maw	10.0	23.0	13.9
Otis Lee	3.7	5.8	4.5
Rita Fisher	13.9	22.0	17.0
Sharon Cummings	9.1	15.9	11.6
Susan Jones	14.9	17.7	16.1
Tamer Elsayed	3.8	7.5	5.0
Theodore Smith	10.5	14.6	12.2
Tom Linton	9.5	27.7	14.2
Average	**8.5**	**19.0**	**11.7**

Based on the results shown in Table 2 and Table 3 and the detailed results of different persons, we can make the following observations:

1. The traditional named entity recognition tools do not perform well in the irregular text of Web pages. So it is difficult for our system to find the accurate border of attribute value. For example, the NER tool only identifies the *Egypt* within *the Khedive of Egypt* as a location, which will lead to a false attribute extraction result, for it is the whole phrase *the Khedive of Egypt* which acts as the true attribute value. On the other hand, as the capitalized first letter is a heavy feature in traditional NER system, the NER tool will recognize many error NEs on the navigation bar and title bar, where the words are usually capitalized. A clearing step is needed to identify the main block of Web pages.
2. The need of extract multi-type NEs and Noun phases increases the difficulty of attribute extraction. The traditional named entity recognition research, such as the MUC (Message Understanding Conference), CoNLL (Conference on Computational Natural Language Learning), IEER (Information Extraction-Entity Recognition Evaluation) and ACE (Automatic Content Extraction), mainly focuses on named entity of limited types. However, in order to extract attribute value, we need to extract NEs different from previous types, such as awards, nationality, etc. In this paper, we use the gazetteers generated from large knowledge base to recognize the named entity for the lack of training corpus.
3. It needs large corpora in attribute extraction for the little consistency of context in different datasets. The previous research always based on the assumption that the values of the same attribute class will present some context

consistency. For the Wrapper, it is the structure consistency within the same web site; for the text mining technique, it is the context word and the dependency path consistency. As seen in the results in Table 2 and Table 3, we find the classifier trained on develop dataset performs poorly on the test dataset, which indicates that the little context consistency existed between this two datasets.

4. For many Web pages are multi-topical, there is a need to model the relation between the target individual and the attribute value. In this paper, we model this relation using the dependency path between them. However, we find many dependency paths are too long to provide useful relation information. We also consider that the coreference resolution and the sentence syntactic parsing are possibly needed to provide better dependency analysis.

5. It is difficult to extract attributes only using the evidences in a single Web page. As seen above, the Web pages are noisy, multi-topical and irregular, so it is hard to develop an attribute extraction system that can perform well on all different kinds of Web pages.

4. Conclusion

We describe our CASIANED system that extract attribute values of the given individual from Web pages, as defined in the Web People Search Task's sub task Attribute Extraction Task. We extract attribute based on information extraction: multi-information extraction techniques are used to extract attribute candidates. These candidates are verified through classification.

We observed that it is difficult to extract attributes only using the evidences in a single Web page. However, in many different tasks such as knowledge base population, we only need to extract the attribute of the individual, instead of to identify every attribute appearance in different Web pages. So we can develop systems which can accumulate evidences on a large collection of Web pages to enhance the performance of person attribute extraction.

5. ACKNOWLEDGMENTS

The work is supported by the National Natural Science Foundation of China under Grants no. 60673042, the National Natural Science Foundation of China under Grants no. 60875041 and the National High Technology Development 863 Program of China under Grants no. 2006AA01Z144.

REFERENCES

[1] Javier Artiles, Julio Gonzalo, and Satoshi Sekine. The SemEval-2007 WePS evaluation: Establishing a benchmark for the Web People Search Task. In SemEval, ACL, 2007.

[2] Yanhong Zhai, and Bing Liu. Web Data Extraction Based on Partial Tree Alignment. In Proceeding of WWW, 2005.

[3] Jun Zhu, et al. 2D Conditional Random Fields for Web Information Extraction. In Proceeding of ICML, 2005.

[4] Jun Zhu, et al. Simultaneous Record Detection and Attribute Labeling in Web Data Extraction. In Proceeding of KDD, 2006.

[5] Rayid Ghani, et al. Text Mining for Product Attribute Extraction. In Proceeding of KDD, 2006.

[6] Sujith Ravi and Marius Pasca. Using Structured Text for Large-Scale Attribute Extraction. In Proceeding of CIKM, 2008.

[7] Tak-Lam Wong, Wai Lam and Tik-Shun Wong. An Unsupervised Framework for Extracting and Normalizing Product Attributes from Multiple Web Sites. In Proceeding of SIGIR, 2008.

[8] Nadeau, David et al. A survey of named entity recognition and classification. Linguisticae Investigationes. 30, 1 (January, 2007), 3-26.

[9] Satoshi Sekine and Javier Artiles. 2009. WePS 2 Evaluation Campaign: overview of the Web People Search Attribute Extraction Task. In 2nd Web People Search Evaluation Workshop (WePS 2009), 18th WWW Conference, April.