RENS – Enabling a Robot to Identify a Person

Xin Yan[1], Sabina Jeschke[2], Amit Dubey[4], Marc Wilke[1], and Hinrich Schütze[3]

[1] Institute for IT Service Technologies, University of Stuttgart
[2] Center for Learning and Knowledge Management, RWTH Aachen University
[3] Institute for Natural Language Processing, University of Stuttgart
[4] Institute for Communicating and Collaborative Systems, University of Edinburgh

Abstract. We outline a web personal information mining system that enables robots or devices like mobile phones which possess a visual perception system to discover a person's identity and his personal information (such as phone number, email, address, etc.) by using NLP methods based on the result of the visual perception. At the core of the system lies a rule based personal information extraction algorithm that does not require any supervision or manual annotation, and can easily be applied to other domains such as travel or books. This first implementation was used as a proof of concept and experimental results showed that our annotation-free method is promising and compares favorably to supervised approaches.

1 Introduction

This paper outlines our RENS personal information mining system. RENS was originally inspired by the scenario of a receptionist robot who determines the identity of a person by using a facial recognition process coupled to a web search process. To be able to focus on the aspect of personal information mining, we assume the existence of a functioning facial recognition system capable of collecting, for a given target person, *URL-image pairs*: pairs of a URL and an image such that the URL's page contains at least one *identifying image*, i.e., an image of the person's face. A URL-image pair indicates the mapping between a web page and its containing identifying image and therefore enables us to combine web mining and facial recognition technologies. After analyzing these URL-image pairs and their retrieved web pages, the system searches the web for the person's information and generates a business card for him as output.

The RENS system attempts to address the following three issues:

1. How to ascertain an unknown person's identity using given URL-image pairs
2. How to extract personal information from an HTML page
3. How to select the right personal information for a particular person

For the purposes of our system, the *identity* of a person is defined to include his name and his organization. The *personal information* of a person is defined by the typical information shown on a business card, including address, email, telephone, fax number, title and position. We define a *personal information record*

M. Xie et al. (Eds.): ICIRA 2009, LNAI 5928, pp. 43–54, 2009.

(or simply, *record*) as an area with high density of personal information about a particular person on a web page.

The RENS system has three main components: Personal Identity Ascertainment (henceforth referred to as *PIA*), Records Extraction (*RE*), and Records Selection (*RS*). Each of the three components addresses one of the problems listed above. The PIA component ascertains a person's name and organization by applying a named entity recognizer to the relevant text contents of web pages that contain identifying images. The RE component moves through the DOM tree of a web page and extracts records by applying a rule-based algorithm. The RS component selects the best records matching a particular person by calculating and sorting the records' confidence scores based on cosine similarity.

Empirical evaluations were conducted on web pages related to people working in academia as their personal information is often freely available online. Experimental results show that the methods proposed in this paper are promising. Our two main contributions are: (1) an investigation of the concept of personal identity ascertainment with given URL-image pairs and (2) the development of a simple but powerful rule-based records extraction algorithm. This paper is organized as follows: Sec. 2 reviews related work. A description of the RENS system is given in Sec. 3. Results of the evaluation are discussed in Sec. 4. Sec. 5 provides conclusions and an outlook on future research.

2 Related Work

Prominent work in personal information mining includes the work of Tang et al.[1] and [2]. Their system ArnetMiner aims at extracting and mining academic social networks. The ArnetMiner system focuses on extracting researcher profiles from the Web automatically, integrating the publication data from existing digital libraries, modeling the entire academic network and providing search services on this academic network. Although extracting researcher profiles is only a component of ArnetMiner, it does similar tasks as the RENS system using a different approach. It first collects and identifies a person's homepage from the Web, then uses a unified approach to extract the profile properties from the identified document [1]. As it supports search for experts, which is similar to search for persons, it is taken as the baseline system in the evaluation of RENS. Yu et al.[4] discuss extracting personal information from résumés in a two step process: first, segmenting a résumé into different types of blocks and, second, extracting detailed information such as address and email from the identified blocks.

In addition to work directly concerning personal information mining, it is worth discussing research related to the underlying techniques used by RENS. RENS extracts information by walking through the nodes of a DOM tree. Such a DOM tree based extraction approach was first introduced by Gupta et al.[3]. Their basic idea was to use the links-to-text ratio and remove nodes with a high ratio in order to extract general web content from different domains. Prasad et al.[5] used a similar DOM-based heuristic applied to news stories. Kim et al.[6] suggested extracting information from DOM trees using tree edit distance.

Another aspect of the RENS system is its use of *wrappers*. A wrapper extracts data (including unstructured, semi-structured and structured data) from web pages and turns the data into a self-described structured representation for further processing. Liu et al.[9] proposed a novel algorithm that is able to discover noncontinuous data records and uses partial tree alignment to integrate data records. In another paper, Liu et al.[8] also proposed a two-step extraction approach by first identifying data records without extracting data and then aligning data items to a schema.

3 System Description

Figure 1 shows an overview of RENS. In addition to the PIA, RE and RS components mentioned in Section 1, there are 3 smaller components: preprocessing, the information source builder (*ISB*) and card generation. We describe these 6 components in turn.

3.1 Preprocessing

The preprocessing step takes a file containing several URL-image pairs as input, and retrieves the corresponding web pages. We call these pages "seed pages", which we use as a starting point to discover the identity and relevant personal information of a person.

3.2 Personal Identity Ascertainment (PIA)

The PIA component determines a person's name and information related to his organization by applying the STANFORD NAMED ENTITY RECOGNIZER(SNER) to the relevant text contents on each seed page respectively. The texts of a seed page are relevant when they describe the page's identifying image, like the image name, the text adjacent to the image, the image's alt text, the page meta information, title and so on. Because SNER is case-sensitive, espacially with person names, and because the texts online are often informally edited, we need to ensure the capitalization of letters is correct in order to increase the accuracy of the named entity recognition. We used a very simple heuristic: if a token is found having a capitalized first letter, then all the occurences of this token will be enforced to have their first letter capitalized. SNER tags proper nouns as PERSON, ORGANIZATION or LOCATION. Anything tagged as a PERSON could possibly be the person name we are interested in. We found that simply taking the most frequently occurring PERSON name resulted in poor results. In our case, a name having occurred 10 times but only in one document(here a seed page) is often less important than another name with an occurence of one time in 7 documents respectively. The high document frequency suggests a high global accordance to the initial information provided by seed pages. Thus, we picked the entity with the highest score according to the formula: $0.99 \times df + 0.01 \times cf$ where *df* refers to the document frequency (where a document is defined as

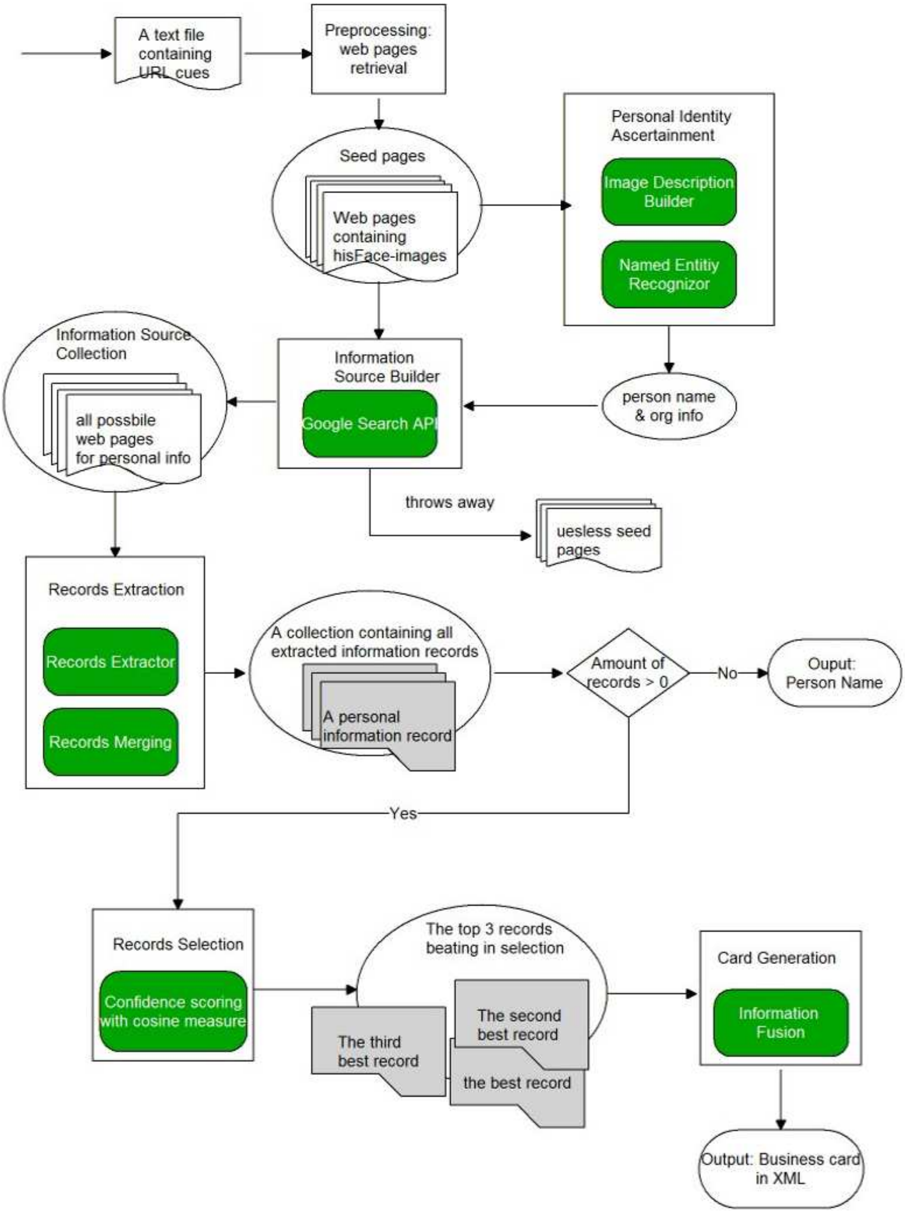

A text file containing URL cues
Preprocessing: web pages retrieval
Seed pages
Web pages containing hisFace-images
Personal Identity Ascertainment
Image Description Builder
Named Entitiy Recognizor
person name & org info
Information Source Builder
Google Search API
Information Source Collection
all possbile web pages for personal info
throws away
uesless seed pages
Records Extraction
Records Extractor
Records Merging
A collection containing all extracted information records
A personal information record
Amount of records > 0
No
Ouput: Person Name
Yes
Records Selection
Confidence scoring with cosine measure
The top 3 records beating in selection
The third best record
The second best record
the best record
Card Generation
Information Fusion
Output: Business card in XML

Fig. 1. RENS System Architecture. The 6 main components are highlighted in blue. If there is no record found, system output is the identity of the person. If records are found, the system outputs a business card in XML format.

the text content of a seed page), or number of documents in which a name occurred, and cf refers to collection frequency, or the number of times a name occurs among all the documents. Having attained the person name, one can use pointwise mutual information to find the best matching organization phrase corresponding to the person name. Organization-type named entities are indexed as bigram and trigram phrases; we do this because in a unigram representation, the weighting formula would give too much weight to stop words like "of" and "for" (which are semantically empty, but occur frequently in ORG names). In addition, the key words of organizational names are often phrases of length 2 or 3. The mutual information between person and organization phrases is calculated in the following manner:

$$MI(p, o) = \log \frac{P(p, o)}{P(p)P(o)} \tag{1}$$

where p stands for person name and o for organization phrase. $P(p)$ is the normalized document frequency of p, $P(o)$ is the normalized document frequency of o, and $P(p, o)$ is normalized document frequency of joint occurrences of p and o.

3.3 Information Source Builder (ISB)

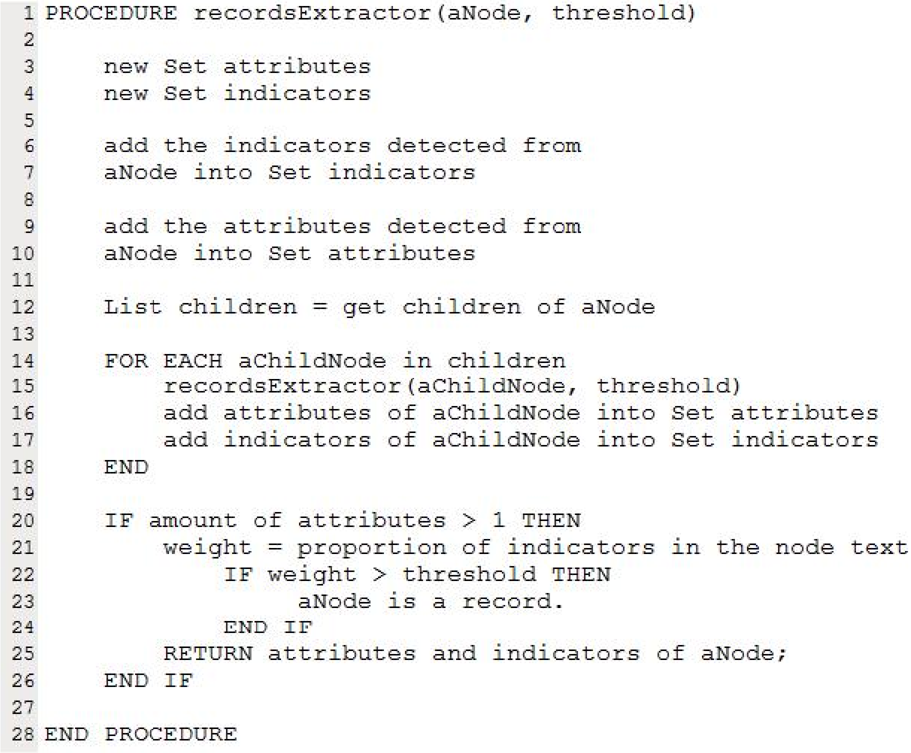
This component has two purposes. First, it uses two search queries (one is the person's name, where possible the full name, another is the combination of the person's name and the organizational association that was also found in the PIA component) and the Google AJAX search API to get the top 10 ranked pages respectively for each query. The seed pages and new pages found during the search represent the information source collection. Second, the ISB removes repetitive or useless (not containing the person name found in the PIA) web pages from the collection.

3.4 Records Extraction (RE)

This component uses the information source collection, namely the output of the ISB, as input. For each page in the collection, it traverses the corresponding DOM tree, annotating all nodes with the personal information features of a particular language, like the ones for US. english mentioned in Table 1. It then uses a local extraction strategy to extract personal information records. The output of this component is a collection of all possible records that are detected from the pages in the information source collection. A recursive bottom-up extraction algorithm we developed in this project is given in Figure 2. The extraction process starts from the `<html>` node which is the root of an HTML DOM tree, and computes a weight for each node inside the DOM tree. This weight is the ratio of the text which can be classified as personal information compared to all the text in the node. When this ratio exceeds a predetermined threshold (empirically determined to be 0.13 for our task), we classify the node as being a potential personal information record.

Table 1. Personal information features: Attributes and Indicators of US. English

Attributes	**Indicators**
Email:	email, netmail e-mail, mailto ···
Telephone:	telephone, tel, call, mobile, phone, cellphone ···
Fax:	fax, telexfax, facsimile ···
Address:	department, avenue, Ave., building, room, office ···
Website:	homepage, website, url ···
Title:	professor, Ph.d ···
Position:	CEO, CFO, dean, chief, coach ···



```
PROCEDURE recordsExtractor(aNode, threshold)

    new Set attributes
    new Set indicators

    add the indicators detected from
    aNode into Set indicators

    add the attributes detected from
    aNode into Set attributes

    List children = get children of aNode

    FOR EACH aChildNode in children
        recordsExtractor(aChildNode, threshold)
        add attributes of aChildNode into Set attributes
        add indicators of aChildNode into Set indicators
    END

    IF amount of attributes > 1 THEN
        weight = proportion of indicators in the node text
            IF weight > threshold THEN
                 aNode is a record.
            END IF
        RETURN attributes and indicators of aNode;
    END IF

END PROCEDURE
```

Fig. 2. The algorithm of the personal information records extractor

Computing this ratio therefore depends upon being able to classify text as containing personal information or not. Recall that personal information is the typical information shown on a business card. A unique property of this kind of information is that there are often obvious words and patterns which strongly indicate its presence. We manually developed a set of these 'indicator words' for US English, a subset of which is listed in Table 1.

Indicators are not limited to words, but also include regular expressions that identify personal information including email addresses, zip codes, telephone and fax numbers. These regular expressions and indicators are not only used to compute the above-mentioned ratio, but also to annotate a node as containing a particular kind of personal information. If an indicator occurs in a node (including its children), the node is annotated with this indicator and the indicator's corresponding attribute. The weight of a node is thus formalized to eq. 2. t is a node inside a DOM tree.

$$weight(t) = \frac{||\text{indicators in t }||}{||\text{ words in t }||} \tag{2}$$

In order to prevent a node with very high weight but only one attribute from being taken as a record, we require that a record have at least 2 attributes. However, not all attributes strongly indicate a personal information record. Attributes like position or title can occur within any node in a DOM tree, because their indicators like "Professor" or "CEO" could be mentioned anywhere on a page with a person name. On the other hand, particular email, fax, telephone or address patterns are very suggestive (in particular, the ZIP code pattern), so they are good attributes to identify a record. Thus, besides the weight, another precondition to be a record is that a node should have at least 2 good attributes. In addition, we count the attributes and indicators in a boolean model which means no attribute gains any additional weight beyond its first occurrence. We do this to dampen the weight of large nodes like `<html>` which may have more than one record as child nodes. If nodes that have 2 good attributes and exceed a predefined threshold are all chosen for records, we would have much redundancy, due to the nested structure of HTML pages. By eliminating the nested records, we finally attain the mutually exclusive personal information records on a page.

3.5 Records Selection (RS)

The records selection component calculates and sorts the confidence scores of the records, and outputs them in sorted order. Scoring uses cosine similarity:

$$\textsc{sim}(R, P) = \frac{\boldsymbol{r} \cdot \boldsymbol{p}}{|\boldsymbol{r}||\boldsymbol{p}|} \tag{3}$$

In eq. 3, the vectors $\boldsymbol{r}$ and $\boldsymbol{p}$ are TF-IDF representations of the record R and the seed page P, respectively. The cosine measures the similarity between $\boldsymbol{r}$ and $\boldsymbol{p}$, and therefore shows how likely the record R relates to the person described on the seed page P. As there are multiple seed pages $(P_1, P_2, P_3, \ldots, P_n)$, the confidence score of the record R is the overall similarity and is calculated using:

$$score(R) = \sum_{i=1}^{n} \textsc{sim}(R, P_i) \tag{4}$$

3.6 Business Card Generation (BCG)

The card generation component takes records as input and generates business cards in XML format as output. We predefined a business card template that is coposed of 5 slots:

person name, fax, telephone, address, academic title or position.

The person name slot is filled with the name we have found in personal identity ascertainment component. For the other slots, we use pattern matching and heuristic methods as annotating a DOM tree node mentioned in Section 3.4. In the end, the business cards are generated in XML format with JDOM.

3.7 Assumptions and Preconsiderations

As the system does not include an actual face recognition system, we have to set certain limitations on the test set used as input. Existing face recognition techniques are not perfect. To account for this deficiency and to simulate a real world scenario, we assume there are a few misleading URL-image pairs, containing information of "wrong" persons. Thus, our first, arbitrary assumption is that the error rate of the input URL-image pairs is 30%. The experiments are performed on web pages related to academics, a useful limitation as their personal information can be easily found online. To reduce complexity, we experimented only on web pages, not including files of other formats like pdf or MS Word. In future , there will be more investigations on these types of files. Finally, we assume that the person we are searching for has only one unique social identity. It is still unclear how to deal with people who have multiple social identities (a mathematician can also be a musician), and who have different personal information during different periods of time. We left further discussions of this problem to future work.

4 Empirical Evaluations

The evaluation consists of 3 tests:

1. *Records Extraction Test*: Given a web page, RENS decides whether the page contains records. If the page contains at least one record, it extracts all detected records from the page.
2. *Personal Identity Ascertainment Test*: Given a set of URL-image pairs, RENS ascertains the name of the person who the set points to. If no full name exists, it finds the first or last name.
3. *Evaluation of the* RENS *System*: Given a set of URL-image pairs, RENS finds the records best matching the target person.

The evaluation was designed to measure the accuracy of the RENS system in the framework of the tests defined above.

The records extraction test was performed on 815 web pages. 15 of these do not contain any records directly but have links to contact pages containing records. If RENS detects the records from the contact page of such a test page, we score this instance 1, else 0. In the other 800 web pages that do not have a linked contact page, 500 of them contained at least one record (most of them containing exactly one record). If records of a page are returned, we assign a score of 1 else 0. The other 300 test pages do not contain records. In this case, when the RENS system (correctly) returns no records, we score the instance with 1, else with 0.

For the personal identity ascertainment test and the RENS system test, we used 100 test sets. Each set is composed of 9 URL-image pairs, 3 of which are related to wrong persons, according to the 30% input error rate. In the personal identity ascertainment test, we checked manually whether the output person's name corresponds to the target person. If correct, accuracy is 1, else 0. In the last test, the RENS system test, we check how accurately RENS could find personal information on that particular person. The evaluation metric for this test is given below.

4.1 Metrics

The metrics used to evaluate our system are fine-grained accuracy and coarse-grained accuracy, both of which take a value between 0.0 and 1.0. The coarse-grained accuracy is computed by taking the ceiling of the fine-grained accuracy. The fine-grained accuracy is computed as follows:

- **Case 1** there are information records available for a particular person. If the best record is returned at the first place, the fine-grained accuracy is 1.0, second place 0.8, third 0.6, fourth 0.4, fifth 0.2. After 5th place, the fine-grained accuracy is scored 0.0.
- **Case 2** no information record is provided for a particular person. If no cards are returned by the RENS system, fine-grained accuracy is 1.0, else 0.0.

4.2 Baseline

As a personal information mining system, the RENS system was compared with the ArnetMiner system's expert search component. The ArnetMiner implements the process in three steps: relevant page identification, preprocessing, and extraction. Given a researcher name, they get a list of web pages by a search engine (we use the Google API) and then identify the homepage/introducing page of the researcher and in the end they extract personal information by using machine learning methods[1].

4.3 Results

Records Extraction Test. For the 500 pages containing records, RENS has an accuracy of 91.2%, for the 300 pages without records 93.33% and for the 15 embedded contact pages 80%. It reaches an average accuracy of 91.4% on the 815 test pages. *Personal Identity Ascertainment Test* The test result showed an accuracy of 96% for the PIA component. RENS *System Test* If the best record could be found in one of the seed pages, the result of the RENS system is exceptional with a fine-grained accuracy of 89.6% and a coarse-grained accuracy of 92%. As a comparison, the result of ArnetMiner is 81.6% and 92.0% respectively. If no record of the person is given in the seed pages, the performance of RENS drops down to a fine-grained accuracy of 72.0% and coarse-grained accuracy of 80.0%. In this case, ArnetMiner has 93.6% fine-grained accuracy and 96.0% coarse-accuracy. The average performance of RENS is 80.8% of fine-grained accuracy and 86.0% of coarse-grained accuracy, while ArnetMiner has 87.6% and 94% respectively.

4.4 Discussion

Records Extraction. The extraction test failed for 8.8% of the 500 pages that contain records, mainly because the shortcomings of the local extraction strategy cause false negatives. If a node contains many other text elements besides all the right personal information we need, its weight becomes too small to pass the threshold test. As a result, this node will not be classified as a record. A possible remedy for this weakness is to take the change rate of personal information into account. Inside a node, when entering the area that contains personal information, the number of indicators increases very quickly; upon leaving, the increase rate slows down and eventually approaches 0.

Of the 300 pages that did not contain any records, 7.67% were classified incorrectly. These errors were often numbers with a pattern identical to phone and fax numbers. This is a direct result of the use of regular expressions in the annotation of personal information. Additionally, some people have a separate contact information page that is linked from the main page and contains most of the personal information. To address this problem, RENS uses Google search to acquire additional information beyond the seed pages. The pages found in this manner usually contain the required information or, at worst, link directly to them. In the second case we could use simple regular expressions to extract the contact links. The test result was 12 out of 15 contact pages detected and extracted correctly with an accuracy of 80%. Our approach of records extraction needs many improvements to get a better performance. In many cases, it can not extract all the personal information at one time but requires post processing steps. However, the test result still indicates it is a simple but reasonable way to extract personal information.

Personal Identity Ascertainment. The error rate of 4% proves the high performance of the Stanford named entity recognizer and also the efficiency of our method. The two exceptions that were not found by our method are both Italian names. For the Asian names within our test set, the Stanford named entity recognizer shows a very high accuracy of 100% in recognition.

Rens System. If the seed pages contained the best record already, RENS had a slightly better result in fine-grained accuracy. In some cases, ArnetMiner does not find any correctpersonal information, mainly because its strategy is based on finding a person's homepage or profiling first and then extracting his information. On those people who do not have a valid homepage or never published their personal information right on their homepage or whose personal information is in an embedded contact page, ArnetMiner does not perform very well. rens, in contrast, does not select the homepages as its only source for extraction, thus performing better in the same situation.

However, rens's performance was lower when the right personal information record was not included in the seed pages. This is probably because the search term is not good enough, or in many cases, false positive. If a person does not have his personal information available online, but a related person does, the personal information of this related person will be returned. We have not found a satisfying solution to this problem yet. ArnetMiner performed very well in this case. As a mature academic search engine project, it receives its search term by user input, providing an advantage at the level of search terms and its machine learning extraction approach is often more accurate on a large scale corpus. In the future, we can also apply our automatic annotation methods to prepare a corpus for machine learning approach. Although ArnetMiner has an advantage in search term correctness, in contrast to rens it requires name disambiguation as a large-scale academic search engine. Thus we consider our comparison to be fair.

5 Conclusion

We have presented a methodology for combining facial recognition and web mining technologies enabling a robot to determine a person's identity and his personal information based on visual perception. We have also implemented a simple, yet modular algorithm to extract data records like personal information from web pages. We have tested and compared the resulting, fully automatic system based on heuristics against ArnetMiner, which uses a machine learning approach and needs large labeled training sets. Our simple rule-based approach has shortcomings in accuracy, but delivers a good approximation and shows that our proof of concept is successful. There are a lot of potential future directions of this work. Name disambiguation is crucial to the performance of the system for large scale mining. Further interesting avenues for research are the discovery and interaction of different social contexts, like a person's information as a mathematician vs. as a musician and ensuring that the information mined is up to date.

References

1. Tang, J., Hong, M., Zhang, J., Liang, B., Li, J.: ArnetMiner: Extraction and Mining of Academic Social Networks. In: Proceedings of the Fourteenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD 2008), pp. 990–998 (2008)
2. Tang, J., Hong, M., Zhang, J., Liang, B., Li, J.: A New Approach to Personal Network Search based on Information Extraction. Demo paper. In: Proc. of ASWC 2006 (2006)

3. Gupta, S., Kaiser, G., Grimm, P., Chiang, M., Starren, J.: Automating Content Extraction of HTML Documents, pp. 179–224. Kluwer Academic Publishers, Dordrecht (2004)
4. Yu, K., Guan, G., Zhou, M.: Resume information extraction with cascaded hybrid model. In: IACL 2005: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics, pp. 499–506 (2005)
5. Prasad, J., Paepcke, A.: Coreex: content extraction from online news articles. In: CIKM 2008: Proceeding of the 17th ACM conference on Information and knowledge management, pp. 1391–1392 (2004)
6. Kim, Y., Park, J., Kim, T., Choi, J.: ArnetMiner: Web Information Extraction by HTML Tree Edit Distance Matching. In: International Conference on Convergence Information Technology, pp. 2455–2460 (2007)
7. Gomez, C., Puertas: Named Entity Recognition for Web Content Filtering. Natural Language Processing and Information Systems, 286–297 (2005)
8. Zhai, Y., Liu, B.: Mining data records in web pages. In: ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), pp. 601–606 (2003)
9. Zhai, Y., Liu, B., Grossman, R.: Mining web pages for data records. IEEE Intell. Syst., 49–55 (November/December 2004)