Acta Cybernetica **20** (2012) 419–440.

# Person Attribute Extraction from the Textual Parts of Web Pages

István Nagy T.[*]

**Abstract**

We present a web mining system that clusters persons sharing the same name and also extracts bibliographical information about them. The input of our system is the result of web search engine queries in English or in Hungarian. For system evaluation in English, our system (RGAI) participated in the third Web People Search Task challenge [1]. The chief characteristics of our approach compared to the others are that we focus on the raw textual parts of the web pages instead of the structured parts, we group similar attribute classes together and we explicitly handle their interdependencies. The RGAI system achieved top results on the person attribute extraction subtask, and average results on the person clustering subtask. Following the shared task annotation principles, we also manually constructed a Hungarian person disambiguation corpus and adapted our system from English to Hungarian. We present experimental results on this as well.

**Keywords:** natural language processing, information extraction, web content mining, person attribute extraction, document clustering

## 1 Introduction

Personal names are among the most frequently searched items in web search engines. At the same time, these types of search results ignore the fact that a name may be associated with more than one person. Sometimes person names are highly ambiguous (see Figure 1).

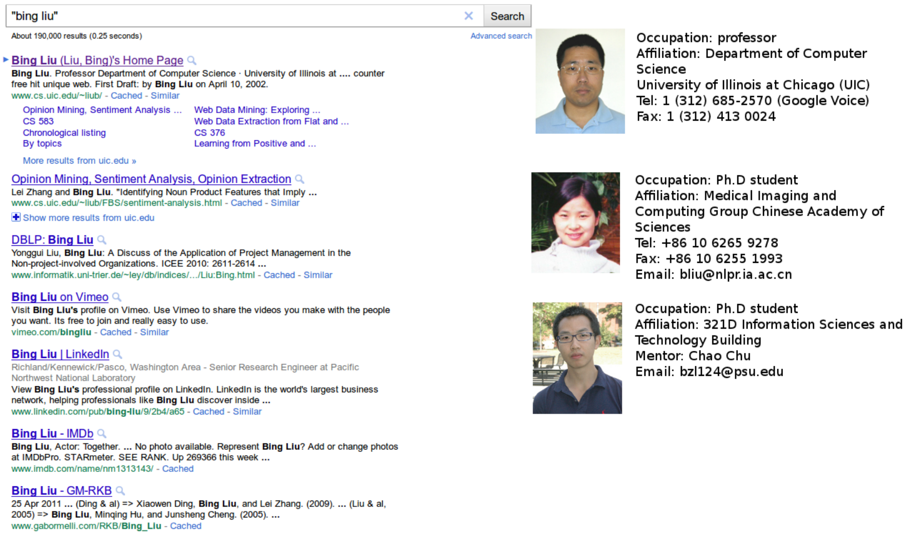
The first Web People Search challenge (WePS) [2] organized in 2007 focused on this disambiguation problem. As input, the participating systems got web pages retrieved from a web search engine using a given person name as a query. The aim of the task was to find all the different people among the results and assign a corresponding document to each person. The second WePS challenge [3] organized in 2009 contained a new subtask. The attribute extraction subtask [30] was to identify 16 different bibliographical attributes from personal web pages such as the

[*] University of Szeged, Department of Informatics, 6720 Szeged, Árpád tér 2., Hungary E-mail: nistvan@inf.u-szeged.hu

DOI: 10.14232/actacyb.20.3.2012.4

birth date, affiliation, and occupation. This subtask proved very difficult and the best system only achieved an F-score of 12.2.

The third WePS shared task [1] introduced a novel subtask which sought to mine attributes for persons, i.e. rather than recognizing attributes in web pages, the task was to assign them to people (the clusters of pages belonging to each given person). This attribute extraction task is more difficult than the previous one because aggregation of recognised text spans has to be carried out as well (people may use synonyms, spelling variants etc. when writing the same content on different pages).



About 190,000 results (0.25 seconds)

**Bing Liu** (Liu, Bing)'s Home Page
**Bing Liu**. Professor Department of Computer Science · University of Illinois at **....** counter free hit unique web. First Draft: by **Bing Liu** on April 10, 2002.
www.cs.uic.edu/~liub/ - Cached - Similar

Opinion Mining, Sentiment Analysis ...
CS 583
Chronological listing
By topics

Web Data Mining: Exploring ...
Web Data Extraction from Flat and ...
CS 376
Learning from Positive and ...

More results from uic.edu »

Opinion Mining, Sentiment Analysis, Opinion Extraction
Lei Zhang and **Bing Liu**. "Identifying Noun Product Features that Imply **...**
www.cs.uic.edu/~liub/FBS/sentiment-analysis.html - Cached - Similar

Show more results from uic.edu

DBLP: **Bing Liu**
Yonggui Liu, **Bing Liu**: A Discuss of the Application of Project Management in the Non-project-involved Organizations. ICEE 2010: 2611-2614 **...**
www.informatik.uni-trier.de/~ley/db/indices/.../Liu:Bing.html - Cached - Similar

**Bing Liu** on Vimeo
Visit **Bing Liu's** profile on Vimeo. Use Vimeo to share the videos you make with the people you want. Its free to join and really easy to use.
vimeo.com/**bingliu** - Cached - Similar

**Bing Liu** | LinkedIn
Richland/Kennewick/Pasco, Washington Area - Senior Research Engineer at Pacific Northwest National Laboratory
View **Bing Liu's** professional profile on LinkedIn. LinkedIn is the world's largest business network, helping professionals like **Bing Liu** discover inside **...**
www.linkedin.com/pub/**bing-liu**/9/2b4/a65 - Cached - Similar

**Bing Liu** - IMDb
**Bing Liu**, Actor: Together. **...** No photo available. Represent **Bing Liu**? Add or change photos at IMDbPro. STARmeter. SEE RANK. Up 269366 this week **...**
www.imdb.com/name/nm1313143/ - Cached

**Bing Liu** - GM-RKB
25 Apr 2011 **...** (Ding & al) => Xiaowen Ding, **Bing Liu**, and Lei Zhang. (2009). **...** (Liu & al, 2005) => **Bing Liu**, Minqing Hu, and Junsheng Cheng. (2005). **...**
www.gabormelli.com/RKB/**Bing_Liu** - Cached

Occupation: professor
Affiliation: Department of Computer Science
University of Illinois at Chicago (UIC)
Tel: 1 (312) 685-2570 (Google Voice)
Fax: 1 (312) 413 0024

Occupation: Ph.D student
Affiliation: Medical Imaging and Computing Group Chinese Academy of Sciences
Tel: +86 10 6265 9278
Fax: +86 10 6255 1993
Email: bliu@nlpr.ia.ac.cn

Occupation: Ph.D student
Affiliation: 321D Information Sciences and Technology Building
Mentor: Chao Chu
Email: bzl124@psu.edu

Figure 1: Personal Name Disambiguation Problem.

Our system handles the web page clustering and person-level attribute extraction tasks together. Here, we introduce a novel approach that is primarily based on biographical attribute extraction and it uses this information to determine the clusters of persons. Our system participated in the third WePS challenge and achieved top results on the person attribute extraction subtask, and average results on the person clustering subtask. In addition, we present the first Hungarian Name Disambiguation Corpus, which is based on the WePS3 challenge corpora. We found that the results on the Hungarian corpus are comparable to the English ones.

In Section 2 we present some related work, which is followed by a description of our methods in Section 3. Experimental results are shown in Section 4 and discussed in Section 5, then in Section 6 we draw some conclusions and make some suggestions for future study.

## 2 Related work

Text mining and information extraction approaches can quite effectively solve the problem of automatic attribute extraction from machine-readable documents. Text mining [12] involves the extraction of hidden or non-trivial information from unstructured (or partially structured) electronic text files. It means the automatic extraction of latent and hidden correlations or knowledge from available data sets. Text mining problems are special and require different solutions. Modeling the structure of natural language-written texts syntactically and semantically is essential and an examination of the styles of these texts is also required.

Among the text mining applications, Web-mining [21] makes use of the internet, which is the world's largest and fastest-growing repository. There are four knowledge discovery domains that pertain to web mining. There are Web Content Mining [13, 24], Web Structure Mining [20, 6], Wrapper Induction [37, 22] and Web Usage Mining [26, 31]. The aim of Web Content Mining is to extract useful information from the natural language-written parts of websites. Web Structure Mining is the process of using graph theory to analyze the node and connection structure of a website. The aim of Wrapper Induction is to automatically extract data from structured data (HTML, DOM). It is based on automatically detecting the structure (e.g. webshops, product information pages or similar documents), and extracting relevant information. However, Web Usage Mining focuses on techniques that could predict user behavior while the user interacts with the web. The approaches introduced in this paper fall into the category of Web Content Mining.

The initial classical Web Content Mining attempts appeared around 1998-'99 [13, 24]. These were basically rule-based systems either hand-crafted rules, or supervised learning rules trained on a manually annotated corpus. The next-generation approaches were weakly supervised learning methods. Here, the input is a seed list of target information pairs and the goal is to gather pairs which are related to each other in the same way as seed pairs. These pairs may contain related entities like country - capital city [15] and celebrity partnerships [9] or form an entity-attribute pair like Nobel Prize recipient - year [16] or may be concerned with retrieving all available attributes for entities [5, 28]. These systems are usually downloaded from the websites which include the current pairs, and syntactic/semantic rules are learned from their sentences. Next, a web corpus is used in the pre-learned patterns to obtain a new pair. These seed-based systems exploit the redundancy of the WWW. They are based on the hypothesis that useful information is available in several places and in several forms on the Internet, hence some very accurate rules for boosting information extraction efficiency were required. Their goal is to find and recognize at least one occurrence of the target item and not to find every occurrence on the web. But it is not the case in biographical attribute extraction from related person's pages. Here, we must capture each mention of the item of information.

A good example of a web-mining application is the portal for job seekers developed by FlipDog.com. That is, job vacancies are automatically collected from 60,000 company websites. Furthermore, they publish an analysis each month about their database trends and changes. The services of this website are used by many organizations because similar comprehensive and timely reviews are unavailable elsewhere. The automatically extracted information about jobs proved to be sufficient to achieve good accuracy, while data from less reliable web pages can be checked manually [25].

One task for text-mining systems is identifying Named Entities (person, organization, location names) in documents, as they usually play an important role. Named Entity recognition was a task in the framework of the Message Understanding Conference MUC-7 [10]. Participants had to identify personal names, geographical names, organizations, and other names categorization and time, quantity, and descriptive terms. In 2003, The Conference on Computational Natural Language Learning (CoNNL) [29] was announced by the open tournaments. The aim was the construction of a Named Entity Recognition (NER) model which could handle English and German texts. One of the most successful and most widely used approaches for sequence labeling is the Conditional Random Fields (CRF) approach [23]. For the Hungarian language, there are existing rule-based [11] and machine learning Named Entity Recognition tools [32, 35]. The statistical systems are based on the Hungarian Named Entity Corpus of Business Newswire Texts [32]. Our system contains Conditional Random Fields-based NER modules that are required for information processing and extraction.

For the person clustering subtasks, the majority of the shared task participants used some pre-processing step before the individual documents could be represented, then some general clustering algorithm was applied [3]. One of the main problems with the person disambiguation problem is deciding how to determine the number of entities present. The number of clusters was estimated by different approaches during the WePS challenges such as cross-validation, permutation tests, resampling and panelized likelihood [8, 14]. However, these methods only work well for clusters that are well separated and thus they all proved inefficient. Some participant systems used hierarchical clustering approaches [17, 7], so they did not have to estimate the number of clusters.

Rule-based approaches [7] achieved the best results for the attribute extraction task of WePS2. However, the two-step methods [36, 19] also did well. These approaches first mark the potential values of attributes, and then in the second step, validators have to decide about the attributes. Other participants used manual and automatic pattern constructions [4].

# 3 Our method

We shall focus on the raw text parts of the web pages because we found empirically that more pages express content in textual form than in structured form. The first step of information extraction may be to construct a good section selection module. When handling the problem, we first extract the candidate attributes from the relevant sections of web pages, then we cluster the pages by merging clusters having common person attributes and aggregate attributes with the persons identified.

## 3.1 Preprocessing

The input of each participant's system was a set of pages retrieved from a web search engine using a given person's name as a query. We assumed that useful information was available in the natural language-written part of websites and tables [27]. This is why we concentrated on the natural language-written part of websites and tables, and we discarded a lot of noisy and misleading elements from pages (e.g. menu elements). These elements can seriously hinder the proper functioning of Natural Language Processing (NLP) tools.

In order to identify textual paragraphs, we applied the Stanford POS tagger [34] to each section of the DOM tree of the HTML files. We assumed that one piece of text was a textual paragraph if it was longer than 60 characters and it contained more than one verb. We extracted several attributes with our own Named Entity Recognition [33] system which was trained on CoNLL-2003 training data sets. When we used this model on the entire set of paragraphs, the accuracy score obtained was low. To handle this problem we developed attribute-specific, relevant section selection modules. Then we looked for the occurrences of all gold standard attributes using simple string matching in each extracted paragraph. In this way, we created a database with positive and negative paragraphs for the actual attribute. Then we created a set of positive words with the most frequently occurring words taken from the positive paragraphs. If a paragraph from the prediction document set contained at least one word from the actual positive list, we marked it as a positive paragraph and we only extracted attributes from these paragraphs. This approach was used to find the occupation, affiliation, award and school attributes.

Table 1: Definition of attributes of Person for the WePS attribute extraction task

| **Attribute Class** | **Examples of Attribute Value** |
|---|---|
| Date of birth | 4 February 1888 |
| Birth place | Brooklyn, Massachusetts |
| Other name | JFK |
| Occupation | Politician |
| Affiliation | University of California, Los Angeles |
| Award | Pulitzer Prize |
| School | Stanford University |
| Major | Mathematics |
| Degree | Ph.D. |
| Mentor | Tony Visconti |
| Nationality | American |
| Relatives | Jacqueline Bouvier |
| Phone | +1 (111) 111-1111 |
| Fax | (111) 111-1111 |
| Email | xxx@yyy.com |
| Web site | http://rgai.inf.u-szeged.hu/ |

### 3.2 Attribute extraction

This subtask involves extracting 16 kinds of "attribute values" for target individuals whose names appear on each of the web pages provided. The attribute types are listed in Table 1.

Our attribute extraction system consists of two main parts, namely a candidate attributes extraction module and an attribute verification module. Based on this approach, we first mark potential attribute values in a paragraph. Then we find out which candidate values have been found.

When handling this subtask of attribute extraction, it seems necessary to classify the attribute classes. Hence, we aggregated similar attributes into logical groups. For instance, the name group contains the *other name*, *relatives* and *mentor* attribute classes. One advantage of this typology scheme is that we can apply the same approach when we extract the same type of attributes. Another is that we can assume subordinate relations among the coherent attributes. For example, we only marked a candidate name as *mentor* if it was not *relatives* or *other name.*

Table 2: Attribute typologies

| **Name** | **Availability** | **Organization** |
|---|---:|---:|
| other name | web page | award |
| mentor | phone number | affiliation |
| | fax | |

Next, we will elaborate on the extraction procedure for each of the attributes.
**Date of birth:** If a paragraph contains *born*, *birth* or *birthday* phrases, we find candidate dates with a date validator within a window of the word. This validator works with 9 different regular expression rules, and can identify dates written in different formats in the span of text.
**Birth place:** When a paragraph contains *born*, *birth*, *birthplace*, *hometown* and *native* phrases, we use the location markups given by the NER tool [33] trained on the locations class of the CoNLL-2003 training data set to identify candidate locations for the birthplace. We accept a location as a birthplace if a birthplace validator validates it.
**Occupation:** According to the WePS2 results, it was one of the most difficult, ambiguous and frequent attribute classes, which is due to the abstract nature of this attribute. Hence we avoided using lists. It was not available in any NER model or training database. So we created a training database by matching all gold annotations to paragraphs. We used simple string matching and we did not know where the actual attribute occurred. However, the resulting data set was very noisy. We trained our NER tool [33] on this training database, and we applied it on the candidate occupation paragraphs.
**Organizations** (*school*, *award*, *affiliation*): We found that these types of attributes were names of organizations so we grouped them together. We also used an NER

tool [33] here trained on the organization class of the CoNLL-2003 training data to identify candidate organization mentions only in affiliation-candidate paragraphs. When the NER model marks a candidate organization phrase, we first search for the school attribute. Then a potential candidate organization is marked as a school if it appears near some cue phrases such as *graduate*, *degree*, *attend*, *education* and *science*. Next, we defined a school validator that uses the MIVTU [36] school word frequency list with *School*, *High*, *Academy*, *Christian*, *HS*, *Central* and *Senior*. We extended this list with *University*, *College*, *Elementary*, *New*, *State*, *Saint*, *Institute* phrases. First letter capitalized sequences, except for some stopwords like *of* and *at* which contain at least one of these words, were marked as a school by a validator. If the school validator did not validate the potential candidate organization, we looked for the award attribute. When candidate sequences appear near cue phrases such as *award*, *win*, *won*, *receive* and *prize*, we assumed an expression with *award* was an attribute. We also defined an award validator that validates a first letter capitalized sequence except for some stopword like *at* or *of*, if it contains at least one element of the *award*, *prize*, *medal*, *order*, *year*, *player* and *best* phrases. When the candidate string is not a valid *school* and *award*, we tag it to the *affiliation* attribute.

**Degree:** A list of degrees complied manually which contains 62 items. When we found one element from these lists in a paragraph, we marked it as a degree attribute. We assumed that the degree attribute might be located far from the name in a CV-type web page.

**Names** (*relatives*, *other name*, *mentor*): These types of attributes are person names so we found that they occur together. To identify name attributes we used an NER tool [33] trained on the person names of the CoNLL training data. A model extracts name phrases as relatives if they appear in the immediate context of the candidate that indicates various relationships like *father*, *son*, *daughter* and so on. Cue phrases were the same as in the MIVTU [36] system used in WePS2 and are also available in Wikipedia. Sometimes we did not mark the potential candidate sequence for *relatives*, but looked for *other name* attributes instead. We hypothesized that a person does not write his or her name using the same number of tokens; at the same time *other name* has to contain at least a part of the original name. For instance when the original name was *Helen Thomas*, we did not accept the candidate string *Helen McCumber*, but we accepted the *Helen M. Thomas* sequence. This hypothesis may not be true for nicknames. If a name was not marked as *relatives* or *other name*, we analyzed the potential candidates for a *mentor* name. If it appeared near cue phrases such as *study with*, *work with*, *coach*, *train*, *advis*, *mentor*, *supervisor*, *principal*, *manager* and *promote* we marked the potential candidate sequence as a mentor attribute.

**Nationality:** We created a list of nationalities that contained 371 elements. It has multiple entries for certain nationalities. Once we had found one element from this list in a paragraph or table, we assumed it was a potential nationality attribute. Then we selected the most frequent potential nationality attribute of the web pages.

When extracting *availability* attribute classes we did not focus just on textual paragraphs, but examined the whole text of web pages as these types of attributes may occur in other parts as well.

**Phone:** When a text contains *tel*, *telephone*, *ph:*, *phone*, *mobile*, *call*, *reached at*, *office*, *cell* or *contact* words or a part of the original name, we applied the following regular expression:

```
(((?[0-9+(][.()0-9s/-]4,[0-9])((?(s?x|s?ext|s?hart).?)?  d1,5)?)
```

It is a permitted regular expression for potential phone numbers. We defined a phone number validator that validated the sequence determined by the regular expression.

**Fax:** We use the same method as that for phone numbers, except for that we look for *fax*, *telfax* and *telefax* phrases.

**E-mail:** We assumed that if somebody offers their e-mail address, it is also a link. Therefore, we examined links that contained the mailto tag. Moreover, we assumed that every mail address contains the original name or one part of the original name. Hence we defined an e-mail address validator that validates email addresses. We generate all character trigrams from the original name and when an e-mail address contains at least one of them, the validator accepts it. We defined a stopword list as well. This list contains words such as *wiki*, *support* and *webmaster*. Should a candidate e-mail address contain one from the stopword list, the validator does not accept it. Next we extracted the domain from all accepted e-mail addresses, which we used for the website attribute.

**Website:** We assumed that when somebody displays a web address on a website, it is also a link, so a web address is a link at the same time. In this case we only extract a website attribute from links. We marked a potential candidate attribute as a website when it contained the original name or one part of the original extracted domain name from the e-mail attribute.

### 3.3 Person disambiguation

Our chief hypothesis in the person disambiguation subtask was that it can be effectively solved by using extracted person attributes. We defined a weighting of attribute classes. The most useful attribute classes were *web address*, *e-mail*, *telephone*, *fax number* and *other name* and they got a weight of 3. In addition, we weighted *birth date* as 2, while *birth place*, *mentor*, *affiliation*, *occupation*, *nationality*, *relatives*, *school*, and *award* each got a weight of 1. Then every document was represented by a vector with extracted person attribute values.

To define a document similarity measure, we needed normalize the attribute to values. We developed individual normalization rules for each attribute class. For example, the birth place of *United States of America* could be referred to as *USA*, *U.S.A.*, *United States*, *Federal United States* and so on. We created a synonym dictionary based on the re-direct links of the English Wikipedia. We developed regular expressions or rules based on normalization procedures for other attribute classes.

As a first approach for web page clustering, a bottom-up heuristic clustering was performed based on this similarity measure. Here, the starting clusters consist of the individual web pages and then the clusters are merged iteratively until a stopping criterion is reached. For each step of this procedure the most similar clusters are merged (the union of their attributes formed the attribute set of the resulting cluster), where the similarity measure of the weighted size of the intersection of the cluster attribute sets was employed. The stopping criterion was defined to be a similarity value threshold of 2, i.e. if the similarity value of the closest clusters is less than 2, the procedure is terminated (`RGAI(A+H)` with attributes as features and hierarchial clustering).

Besides this heuristic bottom-up clustering, we employed the Xmeans algorithm from the WEKA Java package [18] as well. The advantage of this approach is that it is not necessary to define the number of clusters, but we can define the minimum number of clusters. We used the final number of clusters obtained by heuristic clustering as the minimum number of clusters for Xmeans with (`RGAI(A+X)` attributes as features and Xmeans).

We compared our person attribute approach against standard document clustering methods. With the results of `RGAI(S+X)` (snippet as features), we only used the search engine snippet data. These types of representation compress the most important pieces of information. We represented the data with the tf-idf vector space model, where each document is represented by a vector of its token uni- and bigrams and employed Xmeans for clustering. Next, `RGAI(AS+X)` is a hybrid method of the above two approaches, i.e. the feature sets of the person-based attribute and the snippet-based clustering were merged.

We evaluated our attribute-based approach presented above on the Hungarian Name Disambiguation Corpus as well. To compare our method against the document clustering approach, we employed the KMeans algorithm in the WEKA Java package, as we have more information about the test corpus in this case. This algorithm, like other clustering algorithms, requires fixing the number of clusters beforehand. Since this value is not known in this task, we used various heuristics to estimate it. First (`HNum`), we used the final cluster number of the bottom-up clustering output as the number of clusters for KMeans. In the second case (`Simple`), the average number of documents associated with one name was employed (7 on the corpus). After, we applied the gold standard number of clusters (as determined by human annotators) related to each name in the corpus (`Oracle`).

Lastly, we used the two simple clustering baseline methods, which are shown in Figure 2. In the first case, all the documents are assigned to a single cluster and in the second case, each document is assigned to a different cluster.

### 3.4 Attribute aggregation

We had to aggregate those attributes that occurred in web pages and were found in a cluster, i.e. belonged to a particular person. The official evaluation metric of the challenge required only one attribute from each class. As we extracted more than one potential attribute value for each class, we had to choose one (e.g. a person

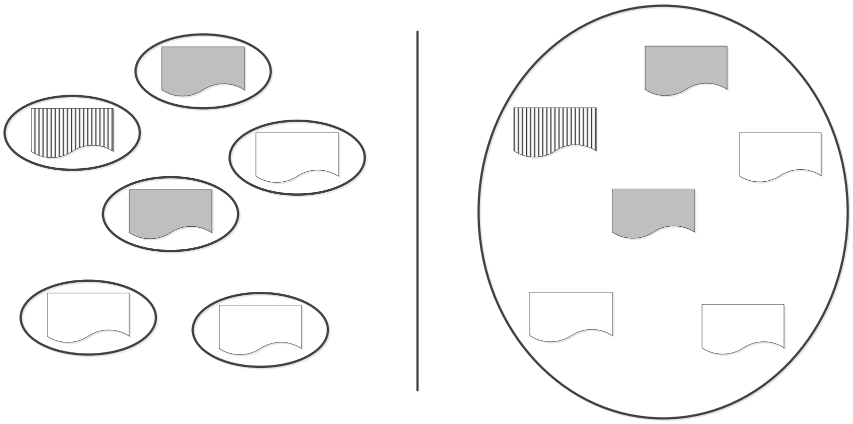


Figure 2: Baseline methods.

may mention several of his affiliations). In the end we chose the most frequent element per person from each attribute class. When some attribute frequencies were the same, we just chose one at random.

# 4 Results

In this section we describe the results achieved on the Hungarian Name Disambiguation Corpus and the third WePS campaign test corpus.

## 4.1 Evaluation issues

In the first WePS, the standard clustering metrics, Purity and Inverse Purity [2] was used to evaluate participant systems against a manually annotated test corpus. During this first evaluation campaign, the organizers concluded that it was possible to cheat using these metrics. The problem is caused by the overlap of the clusters. If a page referred to more than one person, the organizers were allowed to include a document in several clusters. Exploiting this issue, higher scores could be obtained by simply forming one cluster for each document along with one cluster containing all the documents. In this way, a maximal value of Inverse Purity could be obtained together with a high score of Purity. Figure 3 shows an example of the difference between the cheating approach and a correct approach. Since each document is a separate cluster, the cheating system will achieve a maximum value for Inverse Purity. However, it may result in a low Purity score. Since a page can refer to more than one person, it can specify a maximum value of the Purity cluster (for each item). In this way, the average Purity value can be higher.

To overcome of these shortcomings, the organisers of the campaign chose the extended B-Cubed metric for the clustering evaluation metric. We employed this B-cubed metric to evaluate the clustering results of the Hungarian dataset. In this case, we also investigated the precision and recall. However, it is necessary to extend the correctness when a document is classified in several clusters. This is why we defined the multiple version for accuracy and coverage by:

$$\textit{Multiple recall(e, e')} = \frac{Min(|C(e) \cap C(e')|, |L(e) \cap L(e')|)}{|L(e) \cap L(e')|}$$

$$\textit{Multiple precision(e, e')} = \frac{Min(|C(e) \cap C(e')|, |L(e) \cap L(e')|)}{|C(e) \cap C(e')|}$$

where $e$ and $e'$ are two different elements, $L(e)$ is the set of categories and $C(e)$ is the set of clusters associated with $e$. Multiple precision can only be used if e and $e'$ share some cluster, and multiple recall can be used when $e$ and $e'$ share some category. The previous value is maximal (1) when the number of shared categories is lower than or equal to the number of shared clusters, and it is minimal (0) when the two items do not share any category. Here, the multiple recall is maximal when the number of shared clusters is lower or equal to the number of shared categories, and it is minimal when the two items do not share any cluster.

For the attribute extraction subtask, the attributes were only extracted from documents that grouped person A or person B in the clustering annotation. Evaluation metrics were computed as follows:

**Precision:** For a given person, this is the number of correct attribute/value pairs divided by the total number of attribute/value pairs extracted.

**Recall:** For a given person, this is the number of attributes having at least one correct value divided by the total number of attributes for which a correct value has been found by at least one of the systems.

**F-score:** 1 / (alpha * 1/prec + (1-alpha) * 1/rec), where alpha was 0.5.

The above-defined given person is taken from the evaluation for the clustering subtask. Here, every cluster has 2 person names. During the evaluation process for each person the most resampled cluster is defined as the cluster with the higher F-score or with the highest recall, where

**Precision:** The number of documents in the cluster that refer to the person / number of documents in cluster.

**Recall:** The number of documents in the cluster that refer to the person / number of documents that refer to the person.

The organizers found that these methods often missed the cluster with more relevant attributes, resulting in extremely low evaluation results. Hence they defined another metric: they choose the cluster with the best recall of attributes for a person. Next, the organizers defined two different interpretations of the manual annotations, which were combined with the other two clustering evaluation options.

**Strict evaluation:** We count as correct all attribute - value pairs judged as correct by a majority of annotators and as incorrect otherwise.

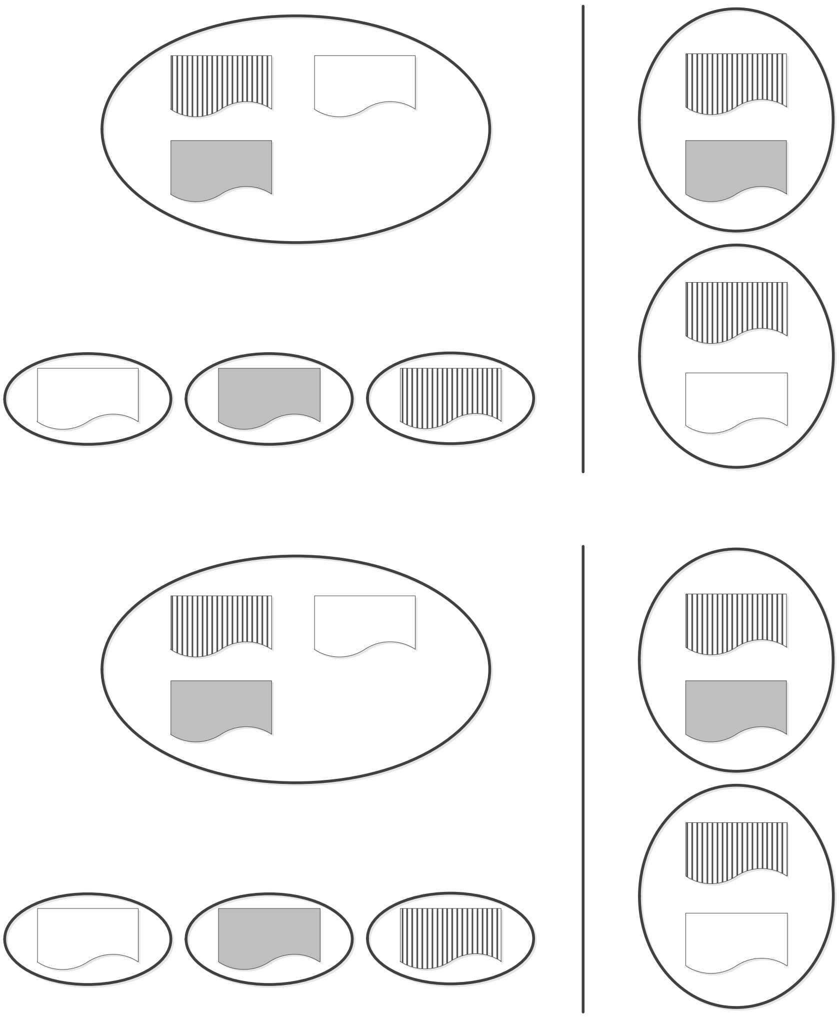


Figure 3: Output of cheat distribution vs. correct solution.

**Lenient evaluation:** We count as correct all attribute - value pairs judged as correct or inexact by a majority of annotators, and as incorrect otherwise.
On the attribute extraction subtask, the official evaluation metric was the attribute recall based clustering with lenient manual annotation. These results are shown in Table 5, where `RGAI(A+X)` achieved the top F-score.

## 4.2 The Hungarian Test Corpus

To evaluate our system, we created the Hungarian Name Disambiguation Corpus with manually annotated web pages. It consists of 10 different Hungarian names and it is available [1] at `http://www.inf.u-szeged.hu/rgai/disambiguation`.

The name list included several well-known figures like *Sándor Csányi* (president of the largest bank in Hungary and a famous Hungarian actor) and the most common names in Hungary like *István Kovács* and *Zsófia Szabó*. When we selected common names, we looked for names of famous people. As in the first case, it could be a famous boxer or in the second case a popular actress. The list included Hungarian President *Pál Schmitt*, a former Olympic champion, who has held several top governmental positions over the years. Tracking down *Pál Schmitt*-related websites can be a hard task because websites write about him in different areas of life.

We downloaded 100 web pages related to each name using the Yahoo search engine[2]. During the annotation process, the human annotators could assign 570 pages to a specific person. So an average of 57 pages was related to each name. However, there were big differences for the various names. For example, in the case of *Miklós Zrínyi*, a historical Hungarian person, the results are mostly related to an institute bearing his name or in the case *Pál Schmitt* the pages focused on just only one person. In total, 120 different individuals were identified by the annotators. In this case there were big differences again for the various names. Over 30 different individuals had the common Hungarian name *Istvács Kovács*, while in the *Pál Schmitt* case the results were mostly related to one person.

## 4.3 Results on the Hungarian Test Corpus

The results achieved by different cluster number initializations are shown in Table 3. As can be seen, our attribute-based document clustering approach achieved the best results on the Hungarian Name Disambiguation Corpus. When we got the correct number of clusters (*Oracle*), it achieved the best results among the clustering methods. The other two approaches yielded the same F-score with different precision and recall values.

## 4.4 The English Test Corpus

The WePS3 datasets containing 300 names were used for testing, in contrast to WePS2 where the test database consisted only of 30 names. The person names were obtained from a US Census, Wikipedia, Computer Science PC lists and names which contained at least one person who is a lawyer, corporate executive or estate agent. For each name the top 200 web search results from the Yahoo! API were downloaded and archived with their corresponding search metadata, like search snippet, title, URL and position in the results ranking.

[1] under CC license

[2] www.yahoo.com

Table 3: Person disambiguation results on the Hungarian corpus

| | average B-Cubed | | |
|---|---|---|---|
| **Approach** | **precision** | **recall** | **F-score** |
| **A+H** | **0.59** | **0.64** | **0.59** |
| All_In_One | 0.43 | 0.84 | 0.50 |
| Oracle | 0.59 | 0.37 | 0.43 |
| Simple | 0.52 | 0.38 | 0.36 |
| HNum | 0.69 | 0.28 | 0.36 |
| One_in_One | 0.93 | 0.24 | 0.35 |

During the annotation process, only two-person-related websites were labeled by the annotators. In this way, the annotation effort was radically reduced. Consequently, large amounts of human resources and time were saved. Clearly, the gold standard used was not perfect. When the Hungarian Name Disambiguation Corpus was created we avoided this problem and the annotators had to identify each individual person.

The attribute extraction subtasks were not manually annotated in the gold standard test database of WePS3. They were only manually evaluated for the participant systems' outputs. Only the clustering subtask of finding person-related attributes was evaluated. Then an annotator got a website with the ten most common attributes - value pairs according to the participant systems, and he or she had to decide which attribute belonged to which of the following categories:

- Correct: the attribute appears in the website and it is related to the actual person.
- Incorrect for any reason other than being too long or too short. For instance, the type of attribute is incorrect (e.g. a number is incorrectly identified as a telephone number); the attribute is not related to the actual person (e.g. the attribute describes some other person described on the page); or the attribute simply did not appear in the text.
- The attribute is correct, but it is too long or too short. So the attribute has one of the following problems:
  - It is too short. The attribute is incomplete (e.g. *director* when it should say *director of marketing*).
  - It is too long. The attribute contains a correct value but includes irrelevant information (e.g. *CEO in 1982* when it should say just *CEO* ).
- Cannot decide, because the web page is unreadable for some reason.
- The web page is readable, but the specified person is not on this page.

## 4.5 Results on the English Test Corpus

Here, we describe the results obtained on the WePS3 challenge.

### Clustering results on the English Test Corpus

During the evaluation of the clustering subtask the organizers used the extended versions of BCubed Precision and Recall, which was the official evaluation metric with alpha set to 0.5. They evaluated the clustering of documents for each query, just focusing on two different people, except for 50 names, where only documents about one person were considered. The official results on the clustering task of the RGAI systems, other participants and the two baselines results are shown in Table 4. Here, our `RGAI(S+X)` system achieved the best scores.

Table 4: Document clustering results on the English corpus

| | | **average B-Cubed** | | |
|---|---|---|---|---|
| **rank** | **System** | **preceison** | **recall** | **F-score** |
| 1 | YHBJ_unofficial | 0.61 | 0.60 | 0.55 |
| 2 | AXIS | 0.69 | 0.46 | 0.50 |
| 3 | TALP | 0.40 | 0.66 | 0.44 |
| 4 | **RGAI(S+X)** | **0.38** | **0.61** | **0.40** |
| 5 | WOLVES | 0.31 | 0.80 | 0.40 |
| 6 | RGAI(S+X) | 0.38 | 0.61 | 0.40 |
| 7 | RGAI(A+H) | 0.40 | 0.57 | 0.40 |
| 8 | DEADELUS | 0.29 | 0.84 | 0.39 |
| 9 | BYU | 0.52 | 0.39 | 0.38 |
| 10 | RGAI(A+X) | 0.47 | 0.43 | 0.38 |
| 11 | RGAI(AS+X) | 0.36 | 0.55 | 0.38 |
| | one_in_one_baseline | 1.00 | 0.23 | 0.35 |
| 12 | HITSGS | 0.26 | 0.81 | 0.35 |
| | All_in_one_baseline | 0.22 | 1.00 | 0.32 |

### Attribute extraction results on the English corpus

The WePS2 train and test sets were used for the training set, which contained 5,122 websites with 187,032 paragraphs. We found 2,781 affiliations, 3,419 occupations and 2,092 biographical paragraphs. For the location, organization and names, markups given by the NER tool [33] trained on the CoNLL-2003 training data set, it achieved F-scores of 89.94% on names, 87.06% on locations and 76.37% on organizations evaluated on the CoNLL-2003 evaluation set [29].

As Table 6 shows, the results of the RGAI systems when the clustering resemblance was the recall approach and the manual annotation was lenient. Our

Table 5: Lenient annotation and attribute recall based clustering

| **System** | **precision** | **recall** | **F-score** |
|---|---|---|---|
| **RGAI(A+X)** | **21.53** | **24.53** | **18.46** |
| RGAI(S+X) | 17.99 | 19.46 | 14.72 |
| Intelius_AE_UNOFFICIAL | 16.48 | 17.18 | 13.15 |
| RGAI(AS+X) | 14.95 | 16.46 | 12.52 |
| RGAI(S+X) | 16.37 | 15.09 | 12.45 |
| RGAI(A+H) | 15.46 | 15.33 | 12.19 |
| BYU | 11.11 | 13.94 | 9.86 |
| WOLVES_AE_1 | 18.49 | 8.83 | 9.60 |

best result was achieved by the `RGAI(A+X)` system, but the Intelius system was outstanding.

Table 6: Lenient annotation with recall based clustering

| **System** | **precision** | **recall** | **F-score** |
|---|---|---|---|
| Intelius_AE_UNOFFICIAL | 10.66 | 14.55 | 9.93 |
| **RGAI(A+X)** | **5.81** | **8.99** | **5.93** |
| WOLVES_AE_2 | 5.42 | 5.99 | 4.55 |
| RGAI(AS+X) | 3.99 | 7.56 | 4.38 |
| RGAI(S+X) | 3.84 | 7.51 | 4.37 |
| WOLVES_AE_1 | 4.96 | 4.77 | 4.14 |
| RGAI(A+H) | 3.76 | 5.15 | 3.67 |
| RGAI(S+X) | 3.82 | 5.27 | 3.55 |
| BYU | 2.86 | 5.16 | 3.03 |

However, when we used the lenient annotation interpretation and the clustering approach based on the F-score, our `RGAI(A+X)` system achieved significantly better results.

When the clustering method was attribute recall based and the annotation was strict, the `RGAI(A+X)` system gave the top F-score.

When we used the strict annotation and recall-based clustering approach, the results of the Intelius system were noticeably better than those of the other systems. It was able to cluster the documents better (see Table 9).

Next, Table 10 shows the results got when the clustering approach was based on the F-score and the annotation was strict. `RGAI(A+X)` achieved the best results, but the other systems performed fairly well.

The above tables indicate that our approach achieved an F-score slightly above 10 on the F-score-based clustering. Compared to the WePS2 results – where the best system achieved about an F-score of twelve – these results are competitive as

Table 7: Lenient annotation with F-score based clustering

| **System** | **precision** | **recall** | **F-score** |
|---|---|---|---|
| **RGAI(A+X)** | **13.22** | **15.48** | **11.67** |
| RGAI(S+X) | 12.14 | 13.23 | 10.06 |
| RGAI(AS+X) | 11.26 | 11.89 | 9.38 |
| RGAI(S+X) | 11.58 | 10.42 | 8.71 |
| WOLVES_AE_1 | 15.17 | 7.09 | 8.14 |
| RGAI(A+H) | 9.97 | 9.37 | 7.57 |
| Intelius_AE_UNOFFICIAL | 7.44 | 7.23 | 6.39 |
| BYU | 5.99 | 7.88 | 5.59 |
| WOLVES_AE_2 | 7.46 | 6.42 | 5.35 |

Table 8: Strict annotation and attribute recall based clustering

| **System** | **precision** | **recall** | **F-score** |
|---|---|---|---|
| **RGAI(A+X)** | **21.38** | **24.48** | **18.40** |
| RGAI(S+X) | 17.94 | 19.46 | 14.69 |
| Intelius_AE_UNOFFICIAL | 16.41 | 17.19 | 13.13 |
| RGAI(AS+X) | 14.88 | 16.41 | 12.47 |
| RGAI(S+X) | 16.37 | 15.11 | 12.47 |
| RGAI(A+H) | 15.43 | 15.34 | 12.20 |
| BYU | 11.08 | 13.98 | 9.86 |
| WOLVES_AE_1 | 18.60 | 8.83 | 9.62 |
| WOLVES_AE_2 | 7.74 | 7.14 | 5.79 |

Table 9: Strict anotation with recall based clustering

| **System** | **precision** | **recall** | **F-score** |
|---|---|---|---|
| Intelius_AE_UNOFFICIAL | 9.81 | 14.37 | 9.60 |
| **RGAI(A+X)** | **5.81** | **8.83** | **5.89** |
| RGAI(S+X) | 4.20 | 7.98 | 4.76 |
| WOLVES_AE_2 | 5.40 | 5.99 | 4.54 |
| RGAI(AS+X) | 3.97 | 7.53 | 4.36 |
| WOLVES_AE_1 | 4.94 | 4.75 | 4.12 |
| RGAI(A+H) | 3.76 | 5.16 | 3.67 |
| RGAI(S+X) | 3.82 | 5.28 | 3.56 |
| BYU | 2.81 | 5.14 | 2.99 |

Table 10: Strict annotation with F-score based clustering

| System | precision | recall | F-score |
|---|---|---|---|
| **RGAI(A+X)** | **13.00** | **15.80** | **11.72** |
| RGAI(S+X) | 12.25 | 13.28 | 10.13 |
| RGAI(AS+X) | 10.90 | 11.35 | 8.98 |
| RGAI(S+X) | 11.48 | 10.52 | 8.79 |
| WOLVES_AE_1 | 15.61 | 7.14 | 8.20 |
| RGAI(A+H) | 9.97 | 9.64 | 7.80 |
| Intelius_AE_UNOFFICIAL | 7.28 | 7.10 | 6.10 |
| BYU | 6.05 | 8.11 | 5.68 |
| WOLVES_AE_2 | 7.44 | 6.42 | 5.34 |

we solved a more complex problem here. Nevertheless, the recall-based results tell us that our clustering approach needs to be improved.

## 5 Discussion

On the Web People Search Clustering Task our system managed to achieve a B-Cubed F score of 40, which is largely due to the annotation process of the test corpus. Although the participants' systems were designed to create the best possible clustering of the whole corpus, during the annotation process only two of the websites related to each person were manually labeled by the annotators. In this way, the organizers spared a large amount of human resources and time, but the gold standard was not perfect. When the Hungarian Name Disambiguation Corpus was created, it did not address this problem and the annotators had to identify each website belonging to the given person. The annotation process did not require large resources because it was a much smaller corpus than the WePS gold standard. Our system achieved a B-Cubed F score of 59 on this corpus, so the development of the gold standard had a beneficial effect on the results. However, a comparison of our results on corpora with different languages suggests that our method is roughly language independent.

The attribute extraction task proved very difficult as the best system only achieved an F-score of 12.2 on the second WePS. The third WePS shared task introduced a novel, harder subtask which sought to mine attributes for a person. In this case it was a major challenge to decide which extracted attribute was related to the actual person and which attribute was related to someone else. We found that for some of the attribute classes (like *occupation*), rule-based approaches outperform machine learning approaches in extracting for the given attribute because these attribute classes are too general. However, the rule-based methods were not perfect, and they are not language independent. We think that an adequate accuracy can be attained just by employing procedures which incorporate external semantic knowledge.

We were able to extract some attributes better (like *mentor*, *other name*, or *relatives*) via machine learning approaches when we placed the attribute classes into different logical groups and we could assume subordinate relations among the coherent attributes.

## 6 Summary and Conclusions

In this article, we presented a joint approach for the person name disambiguation and attribute extraction tasks. Our method is based on the useful biographical attributes obtained from the natural language-written parts of websites. We defined 16 different attribute classes, which were extracted by automatic tools. Our method yielded a B-Cubed F-score of 59 on the first Hungarian Name Disambiguation Corpus, while on the WePS3 clustering task it achieved a B-Cubed F-score of 40. These differences can be partly explained by the annotation differences. For the attribute extraction task, our method efficiently extracted the different types of attributes from web pages and we achieved top results on the WePS3 challenge. We think that the two main reasons for the success of our attribute extractor are the following. First, our approach groups attribute classes and introduces rules which efficiently handle the interdependencies among these classes. Second, we focused on the textual parts of the web pages using NLP tools, which demonstrates that raw text parts of person web pages should be analyzed along with the structured parts of the pages.

Although our results seem satisfactory, there are several possible directions for further improvements of the system. One is that the system could be extended with useful biographical information from the table parts of websites. Another is that the validators could be extended by performing a deeper syntactic analysis. Yet another is that an identification of the subjects of the paragraphs seems necessary. Some of the errors that arose were due to the hypothesis that all extracted attributes are related to the actual person, which is not always the case, hence it should be reconsidered.

## 7 Acknowledgments

This work was supported by the project BELAMI financed by the National Innovation Office of the Hungarian government. The author would like to thank Richárd Farkas for his useful comments and suggestions on a draft version of this paper.

## References

[1] Artiles, Javier, Borthwick, Andrew, Gonzalo, Julio, Sekine, Satoshi, and Amigó, Enrique. WePS-3 Evaluation Campaign: Overview of the Web People

Search Clustering and Attribute Extraction Tasks. In *Conference on Multilingual and Multimodal Information Access Evaluation (CLEF)*, 2010.

[2] Artiles, Javier, Gonzalo, Julio, and Sekine, Satoshi. The SemEval-2007 WePS Evaluation: Establishing a benchmark for the Web People Search Task. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*. ACL, 2007.

[3] Artiles, Javier, Gonzalo, Julio, and Sekine, Satoshi. WePS 2 Evaluation Campaign: overview of the Web People Search Clustering Task. In *2nd Web People Search Evaluation Workshop (WePS 2009), 18th WWW Conference*, 2009.

[4] Balog, Krisztian, He, Jiyin, Hofmann, Katja, Jijkoun, Valentin, Monz, Christof, Tsagkias, Manos, Weerkamp, Wouter, and de Rijke, Maarten. The University of Amsterdam at WePS2. In *Second Web People Search Evaluation Workshop (WEPS 2009)*, 2009.

[5] Bellare, Kedar, Talukdar, Partha Pratim, Kumaran, Giridhar, Pereira, Fernando, Liberman, Mark, McCallum, Andrew, and Dredze, Mark. Lightly Supervised Attribute Extraction for Web Search. In *Proceedings of Machine Learning for Web Search Workshop, NIPS 2007.*

[6] Chakrabarti, Soumen, Dom, Byron E., Gibson, David, Kleinberg, Jon, Kumar, Ravi, Raghavan, Prabhakar, Rajagopalan, Sridhar, and Tomkins, Andrew. Mining the Link Structure of the World Wide Web. *IEEE Computer*, 32:60–67, 1999.

[7] Chen, Ying, Lee, Sophia Yat Mei, and Huang, Chu-Ren. PolyUHK: A Robust Information Extraction System for Web Personal Names. In *2nd Web People Search Evaluation Workshop (WePS 2009), 18th WWW Conference*, 2009.

[8] Chen, Ying and Martin, James H. CU-COMSEM: Exploring Rich Features for Unsupervised Web Personal Name Disambiguation. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 125–128, Prague, Czech Republic, June 2007. Association for Computational Linguistics.

[9] Cheng, Xiwen, Adolphs, Peter, Xu, Feiyu, Uszkoreit, Hans, and Li, Hong. Gossip Galore - A Self-Learning Agent for Exchanging Pop Trivia. In *EACL (Demos)*, pages 13–16. The Association for Computer Linguistics, 2009.

[10] Chinchor, Nancy. MUC-7 Named Entity Task Definition. *Proceedings of Seventh Message Understanding Conference*, 1998.

[11] Chiticariu, Laura, Krishnamurthy, Rajasekar, Li, Yunyao, Reiss, Frederick, and Vaithyanathan, Shivakumar. Domain adaptation of rule-based annotators for named-entity recognition tasks. In *Proceedings of EMNLP 2010*, pages 1002–1012, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[12] Cohen, K. Bretonnel and Hunter, Lawrence. Getting Started in Text Mining. *PLoS Comput Biol*, 4(1):e20, 01 2008.

[13] Cooley, R., Srivastava, J., and Mobasher, B. Web mining: Information and pattern discovery on the world wide web, 1997.

[14] Elmacioglu, Ergin, Tan, Yee Fan, Yan, Su, Kan, Min-Yen, and Lee, Dongwon. PSNUS: Web People Name Disambiguation by Simple Clustering with Rich Features. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 268–271, Prague, Czech Republic, June 2007. Association for Computational Linguistics.

[15] Etzioni, Oren, Cafarella, Michael, Downey, Doug, Popescu, Ana-Maria, Shaked, Tal, Soderl, Stephen, Weld, Daniel S., and Yates, Er. Unsupervised Named-Entity Extraction from the Web: An Experimental Study. *Artificial Intelligence*, 165:91–134, 2005.

[16] Feiyu Xu and Hans Uszkoreit and Hong Li. A seed-driven bottom-up machine learning framework for extracting relations of various complexity. In *Proceedings of ACL07, 584–591*, 2007.

[17] Gong, Jun and Oard, Douglas. Determine the Entity Number in Hierarchical Clustering for Web Personal Name Disambiguation. In *2nd Web People Search Evaluation Workshop (WePS 2009), 18th WWW Conference*, 2009.

[18] Hall, Mark, Frank, Eibe, Holmes, Geoffrey, Pfahringer, Bernhard, Reutemann, Peter, and H.Witten, Ian. The WEKA Data Mining Software: An Update;. In *SIGKDD Explorations,Volume 11, Issue 1*, 2009.

[19] Han, Xianpei and Zhao, Jun. CASIANED: People Attribute Extraction based on Information Extraction.

[20] Kautz, Henry, Selman, Bart, and Shah, Mehul. The hidden web. *Artificial Intelligence Magazine*, 18(2):27–36, 1997.

[21] Kosala, Raymond and Blockeel, Hendrik. Web Mining Research: A Survey. *SIGKDD Explorations*, 2:1–15, 2000.

[22] Kushmerick, Nicholas. Wrapper Induction: Efficiency and Expressiveness. *Artificial Intelligence*, 118(1-2):15–68, April 2000.

[23] Lafferty, John, McCallum, Andrew, and Pereira, Fernando. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of ICML-01, 18th Int. Conf. on Machine Learning*, pages 282–289. Morgan Kaufmann, 2001.

[24] Liu, Bing and Chang, Kevin Chen Chuan. Editorial: special issue on web content mining. *SIGKDD Explor. Newsl.*, 6(2):1–4, December 2004.

[25] McCallum, Andrew. Information Extraction: Distilling Structured Data from Unstructured Text. *Queue*, 3(9):48–57, 2005.

[26] Mobasher, B. Web Usage Mining. In *Encyclopedia of Data Warehousing and Mining*, 2006.

[27] Nagy, István T. and Farkas, Richárd. Person attribute extraction from the textual parts of web pages. In *Third Web People Search Evaluation Forum (WePS-3), CLEF 2010*, 2010.

[28] Pasca, Marius. Outclassing Wikipedia in open-domain information extraction: weakly-supervised acquisition of attributes over conceptual hierarchies. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, EACL '09, pages 639–647, Morristown, NJ, USA, 2009. Association for Computational Linguistics.

[29] Sang, Erik F. Tjong Kim and Meulder, Fien De. Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition. In *Proceedings of CoNLL-2003*, pages 142–147, 2003.

[30] Sekine, Satoshi and Artiles, Javier. 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*, 2009.

[31] Srivastava, Jaideep and Cooley, Robert. Web usage mining: Discovery and applications of usage patterns from web data. *SIGKDD Explorations*, 1:12–23, 2000.

[32] Szarvas, György, Farkas, Richárd, Felföldi, László, Kocsor, András, and Csirik, János. A highly accurate Named Entity corpus for Hungarian. In *Proceedings of International Conference on Language Resources and Evaluation*, 2006.

[33] Szarvas, György, Farkas, Richárd, and Kocsor, András. A Multilingual Named Entity Recognition System Using Boosting and C4.5 Decision Tree Learning Algorithms. Discovery Science 2006, 2006.

[34] Toutanova, Kristina, Klein, Dan, Manning, Christopher D., and Singer, Yoram. Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency Network. In *Proceedings of HLT-NAACL*, pages 252–259, 2003.

[35] Varga, Dániel and Simon, Eszter. Hungarian named entity recognition with a maximum entropy approach. *Acta Cybernetica*, 18(2):293–301, 2007.

[36] Watanabe, Keigo and Bollegala, Danushka. MIVTU: A Two-Step Approach to Extracting Attributes for People on the Web. In *Proceedings 2nd Web People Search Evaluation Workshop (WePS 2009), 18th WWW Conference*, 2009.

[37] Yi, Lan. Web page cleaning for web mining through feature weighting. In *Intl. Joint Conf. on Artificial Intelligence (IJCAI)*, pages 43–50, 2003.

*Received 20th June 2011*