A Document Analysis System for Linking Cross-document Entities

Manabu Ohta
Okayama University
3-1-1 Tsushima-naka, Kita-ku, Okayama, Japan
ohta@de.cs.okayam-u.ac.jp

Atsuhiro Takasu
National Institute of Informatics
2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo, Japan
takasu@nii.ac.jp

***Abstract*—This paper proposes an entity extraction and matching system for digital documents. Digital documents usually contain many links to their relevant information, but they do not cover all the links. Entity extraction and matching systems are used to detect such implicit links. They usually consist of several steps such as parsing, dictionary matching, and classification. Some of these steps, however, inevitably cause errors, which must be managed properly so that the process of subsequent steps is not degraded. We have therefore been developing an entity extraction and matching system focusing on managing the errors incurred at each step. This paper overviews the system and explains some techniques we have developed to improve the quality of entity extraction and matching because the system can be a key solution to content management for institutional repositories and academic societies as well as digital libraries.**

***Keywords*-digital library; information extraction; CRF.**

I. Introduction

Progress in information and communication technology is changing the style of users when they read documents. Digital documents are augmented with multimedia content such as video and sounds. Texts themselves tend to be decomposed into small portions and linked to one another as in dictionaries and encyclopedias. Users read such *networked documents* by following links according to their preferred order. In other words, documents are organized by readers according to their purposes and interests. Hence, their organization differs depending on readers' contexts.

Traditional documents such as books and articles are also provided in the same cyberspace with the networked documents. These documents are usually written by a single author or a small group of authors, and readers are expected to read them according to the authors' context. The readability of traditional documents in cyberspace is improved by linking them to ones that are related to them like networked documents. For example, by linking technical terms appearing in a document to the corresponding dictionary pages on the Internet, readers can check the meaning of the terms efficiently and effectively. Linkages are also especially useful for named entities such as people and places.

Papers on computer science and related research areas often contain descriptions of software tools such as support vector machines (SVMs) and conditional random fields (CRFs). If the papers are linked to the download pages of such software tools, readers can easily repeat similar experiments described in the papers. Furthermore, if a system can give a list of tools providing the same functions along with their evaluations such as their processing efficiency, which has been reported in various papers, it would help readers to choose proper software tools.

Because documents often contain references, it would be convenient if we could obtain cited papers without having to search for them. Some researchers and publishers are indeed trying to build systems that provide direct access to cited articles.

In this paper, we use the term *entity* for objects to be linked, such as technical terms, software tools, figures, and tables. The two main functions of linking entities are:

- To extract entities and
- To match entities with related portions in networked documents.

Machine learning techniques are often used for both information extraction and matching. Some researchers have applied sequence labeling techniques to extract entities. For example, Xin et al. proposed that CRFs be applied to extract information from conference home pages [1]. Entity extraction has also been studied as a problem in document layout analysis in the pattern recognition community. Nagy et al. proposed a layout analysis system that extracted bibliographic components such as authors and titles from academic articles [2] in early studies. Story et al. developed a digital library system for academic articles where they extracted various entities from scanned documents [3]. Advanced layout analysis techniques have recently been examined for extracting information from books [4]. We link entities after they have been extracted. Various kinds of machine learning techniques have also been applied to entity matching problems [5]. For example, Bilenko et al. applied approximate string matching and classification methods to this problem [6]. Shu et al. proposed the use of latent topics to improve the accuracy of entity resolution [7].

Because processing errors are unavoidable in these techniques, the main concern of our study is how to control the quality of the entity linkages that resulted. The accuracy of entity linkages is improved by

- Improving the accuracy of all processing modules and

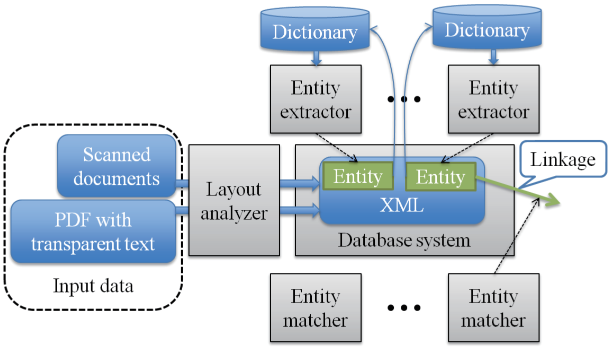


Figure 1. Outline of entity linkage system.

- Preventing errors from being propagated to succeeding processes.

Many researchers have focused on the former, but our concerns are on the latter as well as improvements to all modules.

We occasionally need complex rules and large amounts of training data to acquire accurate extraction and matching of entities. However, it is very labor-intensive to prepare training data. Furthermore, if rules become more complex, we need more training data. Therefore, it is important to reduce the human cost while retaining the quality of linkages, which is another focal point of our study. This paper, therefore, describes our ongoing efforts to develop a document analysis system with these features.

The rest of this paper is organized as follows. Section II overviews the system we are developing. Sections III and IV explain our bibliographic entity extractors for research papers and our entity matchers for generating links between technical terms and their corresponding Web content. Section V summarizes the paper and mentions future work.

II. System Overview

This section overviews our system. As seen in Figure 1, it consists of four modules: a layout analyzer, entity extractors, entity matchers, and a database system. The input of our system is an XML file of OCRed text with bounding rectangles for its characters, words, and lines, as shown in Figure 2. The output of the system is also an XML file where entities are marked up.

A. Input Data and Layout Analyzer

We handle both scanned document images and PDF files. Given a scanned document, we applied commercial OCR to obtain recognized text and the positions of characters. However, we plan to obtain characters and their positions also from PDF with transparent text. The position was represented by a bounding rectangle for each character located on a page. Figure 2 shows part of the fictitious input data for the title page given in Figure 3. The bounding rectangles for lines, words, and characters are tagged by “line”, “word”, and “char”. Attributes “x” and “y” denote the coordinates of the upper left corner of the bounding rectangle whereas “w” and “h” denote the width and height of the rectangle.

Words and lines obtained from scanned documents sometimes contain errors in layout analysis. For example, two lines in different columns may be incorrectly merged into one line. Therefore, we applied layout analysis to correct errors using rules designed for individual journal formats [8].

B. Entity Extractor

Documents contain various kinds of entities such as technical terms, the names of software tools, figures, and tables. We are developing multiple extractors each of which is designed for an accurate extraction of each specific entity. We currently have extractors developed for technical terms and bibliographic components that appear in title pages and reference sections. In addition, we plan to develop extractors for figures and tables.

Information extraction has been studied in natural language processing and machine learning communities [9], where only textual information is utilized. Document image analysis researchers, on the other hand, have developed various methods of layout analysis [10]. We believe that the combination of these techniques will be effective to improve the accuracy of extraction for some entities. For example, the font size and spaces around a bounding rectangle are important features to extract the article title from the title page shown in Figure 3. We discuss the effectiveness of such layout information for entity extraction in Section III.

Another way of improving the accuracy of extraction is to use dictionaries such as authority files. Let us consider the task of extracting bibliographic components from the academic papers shown in Figure 3. If we have an authority file for authors, this helps us find authors on a title page by comparing the words that appear on the page with the authors’ names in the authority file.

We obtain entries for dictionaries by entity extraction. We can enrich entries in the dictionary and then increase the accuracy of extraction by adding them to a corresponding dictionary. The key to enabling this positive feedback is the quality of entity extraction. We are presently trying to solve this problem with two approaches:

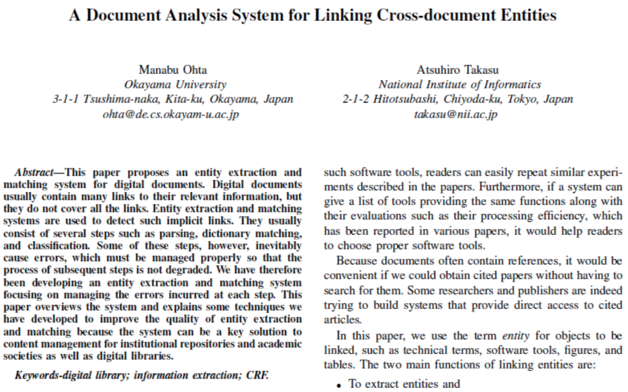
- Manual correction of extraction errors and
- Robust extraction against noise in dictionaries.

C. Entity Matcher

The same entity appears in different documents. For example, an author’s name appears in multiple papers as well as on his/her own home page. An entity matcher detects identical entities that appear in different documents and links them. We need to develop multiple entity matchers each of

```
<line x="626" y="752" w="2580" h="83">
 <word x="626" y="753" w="68" h="77">
  <char x="626" y="753" w="68" h="77"> A </char>
 </word>
 <word x="720" y="753" w="502" h="82">
  <char x="720" y="753" w="60" h="81"> D </char>
  <char x="783" y="770" w="55" h="53"> o </char>
  <char x="839" y="770" w="56" h="53"> c </char>
  <char x="898" y="770" w="58" h="56"> u </char>
  <char x="958" y="770" w="67" h="55"> m </char>
                   ....
 </word>
                   ....
</line>
```

Figure 2. Example of input data.



A Document Analysis System for Linking Cross-document Entities

Manabu Ohta
Okayama University
3-1-1 Tsushima-naka, Kita-ku, Okayama, Japan
ohta@de.cs.okayam-u.ac.jp

Atsuhiro Takasu
National Institute of Informatics
2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo, Japan
takasu@nii.ac.jp

***Abstract*—This paper proposes an entity extraction and matching system for digital documents. Digital documents usually contain many links to their relevant information, but they do not cover all the links. Entity extraction and matching systems are used to detect such implicit links. They usually consist of several steps such as parsing, dictionary matching, and classification. Some of these steps, however, inevitably cause errors, which must be managed properly so that the process of subsequent steps is not degraded. We have therefore been developing an entity extraction and matching system focusing on managing the errors incurred at each step. This paper overviews the system and explains some techniques we have developed to improve the quality of entity extraction and matching because the system can be a key solution to content management for institutional repositories and academic societies as well as digital libraries.**

Keywords-digital library; information extraction; CRF.

such software tools, readers can easily repeat similar experiments described in the papers. Furthermore, if a system can give a list of tools providing the same functions along with their evaluations such as their processing efficiency, which has been reported in various papers, it would help readers to choose proper software tools.

Because documents often contain references, it would be convenient if we could obtain cited papers without having to search for them. Some researchers and publishers are indeed trying to build systems that provide direct access to cited articles.

In this paper, we use the term *entity* for objects to be linked, such as technical terms, software tools, figures, and tables. The two main functions of linking entities are:

- To extract entities and

Figure 3. Title page of this paper.

which is designed for a specific type of entity, as in the entity extractor.

The entity matcher solves the problem with linking records, detecting duplicate records [5], and mining links [11]. The main feature of the entity matcher is its robustness against errors caused by preceding steps. An entity is usually represented by multiple attributes. For example, a person is represented by his/her names, affiliations, and titles. When entities are matched, we first calculate a similarity measure between corresponding attributes, and then integrate the attribute similarity measures into an entity similarity measure (e.g., [6]). Finally, we determine the identity of entities according to their overall similarity. If a document is obtained through OCR, we need to overcome errors in OCR recognition in addition to notational discrepancies. We are trying to handle this problem by approximate string matching that is learnable [12], [13]. We describe a way of handling OCR errors in entity matching in Section IV.

D. Database System

The database system provides functions for managing XML files. We are currently designing a system that focuses on two features. First, the database system cooperates with multiple entity extractors and they are dynamically added, deleted, and modified. As a result, the database system needs to manage dynamically updated tags. Some tags are correlated with one another. For example, one entity extractor detects a person's name as a single entity, whereas another detects its name as a combination of its first and last names. As a result, the database contains tags for full names, first names, and last names. The system needs to appropriately correlate these tags with one another.

Not only OCR but also entity extractors and matchers inevitably make errors. To handle the errors in a succeeding process, we plan to design a markup system that contains information on the quality of data. We are currently considering encoding two kinds of information into the resulting tags: candidates and confidence values. For example, some OCR software tools output candidate characters with confidence values and some CRF tools also generate candidate labels for tokens in a sequence with their confidence values. We plan to encode these kinds of information into the tags of the resulting XML files.

III. Bibliographic Entity Extraction

We are developing an automatic method of extracting bibliographies from a title page of academic articles scanned with OCR markup. The method uses CRFs [14] to label serially OCRed text lines in an article's title page as appropriate bibliographic entity names. Although we achieved excellent extraction accuracies for some Japanese academic journals [15], we needed a substantial amount of training data that had to be obtained by manually extracting bibliographies from printed documents, which was costly. Therefore, we applied some active sampling techniques to the CRF-based extraction of bibliographies to reduce the amount of training data [16]. We achieved favorable experimental results where a sampling strategy using the proposed criteria to select samples could reduce the amount of training data to less than half or even a third that for the two journals used in the experiment. However, later manual correction was still required since extraction errors were unavoidable. Therefore, we also plan to address the problem of detecting such extraction errors as precisely as possible to minimize costly manual corrections.

A. Problem Definition

The automatic extraction of bibliographic entities from a title page of research papers is defined to label each text line on the title page as an appropriate bibliographic entity. Bibliographic entities include titles, authors, abstracts, and whatever other components we find on the title pages of research papers. Note that a bibliographic entity includes at least one text line and is often comprised of several lines.

Figure 3 shows a title page of this paper as an example of a title page of research papers, which starts with a title followed by authors' names and affiliations and continues with an abstract and keywords. Since the title page in Figure 3 includes a title, authors' names and affiliations, an abstract, and keywords, we can generate an XML file where bibliographic markups for these are inserted into the original XML file.

Our OCR was developed in collaboration with an OCR vendor to analyze page layouts and achieve character recognition. Since Japanese articles contain both Japanese and English words, the OCR was equipped with both Japanese and English OCR engines and it automatically selected one of them according to the dominant language of the article. The OCR not only produced recognized text for scanned pages, but also XML markups indicating the bounding rectangles for characters, words, lines, and blocks. The target of bibliography labeling was text lines composed of one or more words. Moreover, these XML elements had the layout attributes of x, y, w, and h shown in Figure 2, and we therefore knew where the text blocks, lines, words, or characters were located on the page and how large they were.

B. CRF-based Bibliography Extraction

We adopt a linear-chain CRF to label text lines. That is, we define the conditional probability of a label sequence, $\boldsymbol{y} = y_1, ..., y_n$, given an input-token sequence, $\boldsymbol{x} = x_1, ..., x_n$ as:

$$p(\boldsymbol{y} \mid \boldsymbol{x}) = \frac{1}{Z(\boldsymbol{x})} \exp \left\{ \sum_{i=1}^{n} \sum_{k=1}^{K} \lambda_k f_k(y_{i-1}, y_i, \boldsymbol{x}) \right\}, \quad (1)$$

where $Z(\boldsymbol{x})$ is the normalization constant, $f_k(y_{i-1}, y_i, \boldsymbol{x})$ is an arbitrary feature function, and λ_k is a learned weight associated with the feature function, f_k.

The CRF assigns the label sequence, $\boldsymbol{y}^*$, to the given-token sequence, $\boldsymbol{x}$, that maximizes Eq. (1), i.e

$$\boldsymbol{y}^* := \operatorname*{argmax}_{\boldsymbol{y}} p(\boldsymbol{y} \mid \boldsymbol{x}). \quad (2)$$

Note that the input-token sequence, $\boldsymbol{x}$, is the sequence of text lines, while the label sequence, $\boldsymbol{y}$, is the sequence of names of bibliographic entities such as the title, authors, and abstract.

We prepared ten kinds of labels for the bibliographic entities listed in Table I to extract these from three target Japanese academic journals. The "type" is that of the article specifically defined for one of the journals. We did not extract them all but a subset of the ten bibliographic entities from articles in a given journal since different journals have slightly different bibliographic entities on their title pages.

We took into account the line's location and size, the gap between lines, and the size and number of characters constituting each line for visual features used for CRF-based labeling. The visual features reflected the layout information of the title pages. The linguistic features for CRF-based labeling were also important, which reflected the textual information of text lines identified through OCR.

Table I
BIBLIOGRAPHIC ENTITIES

Bibliographic entity label	Description
j/e-title	Title in Japanese/English
j/e-authors	Authors' names in Japanese/English
j/e-abstract	Abstract in Japanese/English
j/e-keywords	Keywords in Japanese/English
type	Article type
other	Other text lines

We adopted the proportions of several kinds of characters in the text lines: alphanumerics, kanji, hiragana, and symbols. We also used the appearance of keywords that seemed to be correlated with specific bibliographic entities, e.g., "university" was often found in the author's affiliations. The experiments indicated that more than 98% of the bibliographic entities were correctly extracted from a Japanese academic journal [15]. The experiments also revealed that both visual and textual features were indispensable to the CRF-based labeling of bibliographies.

Our CRF-based labeling was applied to another setting to extract bibliographic entities by adaptively changing the granularity of its target. That is, we applied it to automatically extracting author's names from identified "authors" text lines in scanned academic articles [17]. The experimental results indicated that more than 99% of author-name strings were correctly extracted by using the CRF-based labeling for characters that constituted the "authors" lines.

C. Active Sampling

We achieved more than 98% accuracy in extracting bibliographic entities from a Japanese academic journal; however, we needed 280 articles to train CRF [15]. Therefore, we tried a few active sampling techniques to reduce the amount of training data because such data can only be obtained through manual labeling of bibliographies, which is costly.

We first proposed two confidence measures for selecting samples, both of which reflected the confidence of labeling for thus far manually unlabeled training data [16]. One of these was the normalized likelihood obtained through dividing $\log(p(\boldsymbol{y}^* \mid \boldsymbol{x}))$ shown in Eq. (1) by the length of the input-token sequence, $|\boldsymbol{x}|$. We first calculated these confidence measures for automatically labeled token sequences and then ordered them in ascending order on the basis of these confidence measures. We manually assigned labels to the top-ranked token sequences and added them to the training data. We then obtained CRF using the enriched training data.

The experiments revealed that both confidence measures could reduce the amount of training data to less than half that in random sampling for two Japanese academic journals. However, we observed no significant improvements for one Japanese journal where the accuracy of assigning labels was much lower than those for the other two journals. Apart from active sampling, we also investigated the effect of using pseudo-training data labeled with the current CRF with high confidence in addition to manually labeled training data. Although we observed improvements for one journal with the pseudo-training data, there were no significant differences in the other two journals, which suggests the need for further investigations.

D. Future Perspectives

Although the CRF-based labeling of bibliographies achieved excellent extraction accuracies for some Japanese journals, extraction errors were inevitable. Therefore, we plan to detect such labeling errors automatically to pass them onto manual labeling. This involves the problem of balancing human costs against the quality of extracted bibliographic entities. We expect that the confidence measures proposed for active sampling can also be used for detecting labeling errors since less-confident samples are more likely to be erroneous than more-confident ones. We also expect that eliminating some bibliographic label transitions in CRF will improve labeling accuracy because bibliographic entities have syntactic constraints such that the space for authors typically follows that of the title and is followed by that of the abstract, although different journals have slightly different syntax.

IV. Linkage between Terms and Documents

With more appropriate linkage of digital libraries to Web resources, online-browsing of research papers would be much more comfortable since many digital libraries for research papers are online and accessible from the Web. Such linkages are accomplished in practice by linking terms that appear in papers stored in digital libraries with corresponding Web content. We implemented a prototype system to support the online-browsing of research papers by using the OCRed text of scanned Japanese academic articles [18]. The prototype system extracted technical terms from the OCRed text, and searched Wikipedia and the Web for the best explanatory descriptions of the terms and for related software download pages to generate links to them. We also enhanced the prototype system to further use the extracted technical terms for recommending papers related to browsed papers [19]. We achieved excellent accuracies for extracting technical terms from research papers and reasonable precision in retrieved Web content such as Wikipedia articles, explanatory Web pages, and related software download pages. Moreover, we also attained favorable experimental results where the prototype system could recommend more relevant papers than other methods of recommendation that were implemented for comparison.

A. Online-Browsing Support for Research Papers

The proposed browsing support system i) extracted technical terms from XML files with OCR markups, ii) searched Wikipedia and the Web for the extracted terms to generate links to the retrieved Wikipedia articles and Web pages, and iii) searched the Web for the download pages of software tools related to the technical terms to generate links to them.

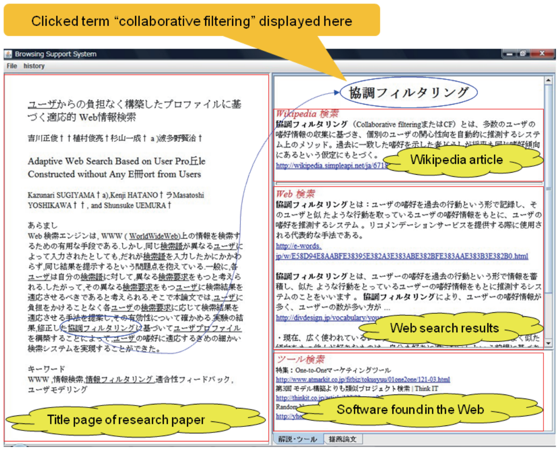
The technical terms were those whose explanations would be helpful to novice researchers and were extracted from research papers by using a Japanese morphological analyzer and some heuristics [18]. Since we handled OCRed text containing recognition errors, we utilized the query correction function of the Yahoo!JAPAN search engine [20], i.e., “Did you mean: *guessed-corrected-term*”. This correction was not always effective especially for short acronyms with various meanings. Therefore, it was only applied if there were a small number of search results for the original term.

1) Matching Technical Terms against Explanatory Content: Once technical terms had been extracted, we searched Wikipedia [21] for explanatory articles on all terms in the first place. Since we only retrieved one article whose head exactly matched the extracted technical term, each technical term had at most one link to a Wikipedia article. We also obtained a summary of the article as a brief explanation of the term, and displayed it near the links on our prototype browser, as shown in Figure 4.

Although retrieved Wikipedia articles usually provide good explanations of technical terms, such articles are not always found in Wikipedia and sometimes may be inadequate. Therefore, we also searched the Web for explanations of terms by using the Web search API provided by Yahoo!JAPAN. We first searched the Web for phrases such as “*technical-term* is defined as” in Japanese and then searched the retrieved Web pages for explanatory sentences that matched 23 different explanatory expression templates we predefined. We extracted the explanatory sentences and their two succeeding sentences as explanations of the technical terms. The prototype browser used these sentences as descriptions of the generated links. The prototype browser indicated at most three links to the explanatory Web pages for each technical term (Figure 4).

2) Matching Technical Terms against Software Download Pages: The download pages of software tools related to technical terms, say, SVM and CRF, were also considered to be helpful to readers of research papers. Therefore, we also searched the Web for such download pages. We used a specifically designed query and some characteristic keywords that often appear on download pages such as “tools”, “software”, and “downloads” to retrieve them. Links to the software download pages were also displayed on the prototype browser, as shown in Figure 4.

3) Recommendation of Related Papers: We also proposed another mode to support the online-browsing of research papers by further utilizing extracted technical terms. That is, we added a function to our prototype browser that recommended papers that were related to a paper that was browsed. Concretely, we first generated a bipartite graph that consisted of papers retrieved by the extracted technical


Clicked term “collaborative filtering” displayed here
Browsing Support System
File history
Title page of research paper
Wikipedia article
Web search results
Software found in the Web
解説・ツール
推薦論文

ユーザからの負担なく構築したプロファイルに基づく適応的 Web情報検索

吉川正俊††植村俊亮†杉山一成† a)波多野賢治†

Adaptive Web Search Based on User Pro丘le Constructed without Any E冊ort from Users

Kazunari SUGIYAMA† a),Kenji HATANO† ラMasatoshi YOSHIKAWA††, and Shunsuke UEMURA†

あらまし
Web 検索エンジンは, WWW (WorldWideWeb)上の情報を検索するための有用な手段である.しかし,同じ検索語が異なるユーザによって入力されたとしても,だれが検索語を入力したかにかかわらず,同じ結果を提示するという問題点を抱えている.一般に,各ユーザは自分の検索語に対して,異なる検索要求をもつと考えられる.したがって,その異なる検索要求をもつユーザに検索結果を適応させるべきであると考えられる.そこで本論文では,ユーザに負担をかけることなく各ユーザの検索要求に応じて検索結果を適応させる手法を提案し,その有効性について確かめる.実験の結果,修正した協調フィルタリングに基づいてユーザプロファイルを構築することによって,ユーザの嗜好に適応するきめの細かい検索システムを実現することができた。

キーワード
WWW ,情報検索,情報フィルタリング,適合性フィードバック,ユーザモデリング

協調フィルタリング

Wikipedia 検索
協調フィルタリング（Collaborative filteringまたはCF）とは、多数のユーザの嗜好情報の収集に基づき、個別のユーザの関心性向を自動的に推測するシステム上のメソッド。過去に一致した嗜好を示した者どうしが将来も同じ嗜好傾向にあるという仮定にもとづく。
http://wikipedia.simpleapi.net/ja/6718

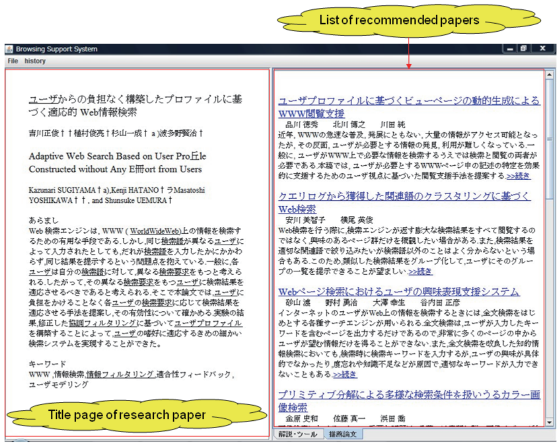
Web 検索
協調フィルタリングとは：ユーザの嗜好を過去の行動という形で記録し、そのユーザと似たような行動を取っているユーザの嗜好情報をもとに、ユーザの嗜好を推測するシステム 。リコメンデーションサービスを提供する際に使用される代表的な手法である。
http://e-words.jp/w/E58D94E8AABFE38395E382A3E383ABE382BFE383AAE383B3E382B0.html

協調フィルタリングとは、ユーザーの嗜好を過去の行動という形で情報を蓄積し、似た ような行動をとっているユーザーの嗜好情報をもとに推測するシステムのことをいいます 。協調フィルタリングにより、ユーザーの嗜好情報が多く、ユーザーの数が多い方が ...
http://divdesign.jp/vocabulary/you

・現在、広く使われている

ツール検索
特集：One-to-Oneマーケティングツール
http://www.atmarkit.co.jp/fitbiz/tokusyuu/01one2one/121-03.html
第3回 モデル構築よりも類似プロジェクト検索 | Think IT
http://thinkit.co.jp/
Random
http://yhe

Figure 4. Prototype browser showing links to explanatory and software download pages.



ユーザからの負担なく構築したプロファイルに基づく適応的 Web情報検索

吉川正俊††植村俊亮†杉山一成† a)波多野賢治†

Adaptive Web Search Based on User Pro丘le Constructed without Any E冊ort from Users

Kazunari SUGIYAMA† a),Kenji HATANO† ラMasatoshi YOSHIKAWA††, and Shunsuke UEMURA†

あらまし
Web 検索エンジンは, WWW (WorldWideWeb)上の情報を検索するための有用な手段である.しかし,同じ検索語が異なるユーザによって入力されたとしても,だれが検索語を入力したかにかかわらず,同じ結果を提示するという問題点を抱えている.一般に,各ユーザは自分の検索語に対して,異なる検索要求をもつと考えられる.したがって,その異なる検索要求をもつユーザに検索結果を適応させるべきであると考えられる.そこで本論文では,ユーザに負担をかけることなく各ユーザの検索要求に応じて検索結果を適応させる手法を提案し,その有効性について確かめる.実験の結果,修正した協調フィルタリングに基づいてユーザプロファイルを構築することによって,ユーザの嗜好に適応するきめの細かい検索システムを実現することができた。

キーワード
WWW ,情報検索,情報フィルタリング,適合性フィードバック,ユーザモデリング

ユーザプロファイルに基づくビューページの動的生成によるWWW閲覧支援
品川 徳秀　北川 博之　川田 純
近年, WWWの急速な普及, 発展にともない, 大量の情報がアクセス可能となったが, その反面, ユーザが必要とする情報の発見, 利用が難しくなっている.一般に, ユーザがWWW上で必要な情報を検索するうえでは検索と閲覧の両者が必要である.本稿では, ユーザが必要とするWWWページ中の記述の特定を効果的に支援するためのユーザ視点に基づいた閲覧支援手法を提案する.>>続き

クエリログから獲得した関連語のクラスタリングに基づくWeb検索
安川 美智子　横尾 英俊
Web検索を行う際に,検索エンジンが返す膨大な検索結果をすべて閲覧するのではなく,興味のあるページ群だけを概観したい場合がある.また,検索結果を適切な関連語で絞り込みたいが検索語以外のことはよく分からないという場合もある.このため,類似した検索結果をグループ化して,ユーザにそのグループの一覧を提示できることが望ましい.>>続き

Webページ検索におけるユーザの興味表現支援システム
砂山 渡　野村 勇治　大澤 幸生　谷内田 正彦
インターネットのユーザがWeb上の情報を検索するときには,全文検索をはじめとする各種サーチエンジンが用いられる.全文検索は,ユーザが入力したキーワードを含むページを出力するだけであるので,非常に多くのページの中からユーザが望む情報だけを得ることができない.また,全文検索を改良した知的情報検索においても,検索時に検索キーワードを入力するが,ユーザの興味が具体的でなかったり,度忘れや知識不足などが原因で,適切なキーワードが入力できないこともある.>>続き

プリミティブ分解による多様な検索条件を扱いうるカラー画像検索
金原 史和　佐藤 真一　浜田 喬

Figure 5. Prototype browser showing list of recommended papers [19].

terms, which we called related papers, and technical terms that appeared in the set of related papers. We then ranked the related papers using the HITS algorithm [22] to analyze the bipartite graph. The prototype browser recommended top-ranked papers to users, as can be seen in Figure 5.

4) Prototype Browser: Figure 4 has the GUI of the prototype browser we implemented for reading research papers. The left window shows major bibliographies such as the title, authors, an abstract, and keywords. The title and authors are both in Japanese and English, but the abstract and keywords are only in Japanese because the prototype browser is only targeting Japanese papers at present. A paper on a “personalized Web search” has been displayed in the left window in Figure 4. The right window indicates a link to the Wikipedia article, three links to explanatory pages, and several links to software download pages. Figure 4 illustrates how these three kinds of links were displayed in the right window when the Japanese term meaning “collaborative filtering” was selected in the left window.

Figure 5 also shows the GUI of the prototype browser we implemented revealing a list of recommended papers. The left window has the same major bibliographies as those in Figure 4 while the right window has a list of recommended papers ranked with our proposed method of recommendation. The same paper as that in Figure 4 is displayed in the left window of Figure 5, and four recommended papers are visible in the right window.

B. Future Perspectives

We need to make the extraction of technical terms more sophisticated in the first place for entity extraction. Moreover, we plan to extract other useful entities such as figures and tables from whole papers instead of from only their title pages. We achieved reasonable accuracies for both explanatory page searches and software download page searches from the perspective of entity matching between extracted terms and their corresponding Web content. We also consider that matching extracted bibliographic entities against existing citation databases is worthwhile since such information on links is useful not only as it is but also for recommending papers [19]. Another focus for future work is to embed the proposed functions into existing document browsers on the basis of our findings with the prototype browser.

V. Conclusion

This paper overviewed our developing entity extraction and matching system that consists of a layout analyzer, entity extractors, entity matchers, and a database system. This paper also explained the prototypes of our entity extractors and matchers for research papers with some findings obtained through experiments. Our prototype entity extractor used both layout and textual information in the extraction of bibliographic entities to improve the quality of extraction. Moreover, it obtained extraction rules with less human cost in labeling by using active sampling techniques. Our prototype entity matcher also obtained reasonable links between technical terms in papers and their corresponding Web content. We plan to make all parts of the proposed system more sophisticated in the future to especially control errors propagated from module to module.

Acknowledgment

This work was supported by two JSPS Grants-in-Aid for Scientific Research (B) (23300040 and 24300097), a JSPS Grant-in-Aid for Young Scientists (B) (23700119), and the Collaborative Research Program of National Institute of Informatics.

REFERENCES

[1] X. Xin, J. Li, J. Tang, and Q. Luo, "Academic conference homepage understanding using constrained hierarchical conditional random fields," in *Proc. of ACM 17th Conf. on Information and Knowledge Management (CIKM 2008)*, 2008, pp. 1301–1310.

[2] G. Nagy, S. Seth, and M. Viswanathan, "A prototype document image analysis for technical journals," *IEEE Computer*, vol. 25, no. 7, pp. 10–22, 1992.

[3] G. A. Story, L. O'Gorman, D. Fox, L. L. Schaper, and H. V. Jagadish, "The rightpages image-based electronic library for alerting and browsing," *IEEE Computer.*, vol. 25, no. 9, pp. 17–26, 1992.

[4] A. Doucet, G. Kazai, and J.-L. Meunier, "ICDAR 2011 Book Structure Extraction Competition," in *Proc. of 11th Intl. Conf. on Document Analysis and Recognition (ICDAR 2011)*, 2011, pp. 1501–1505.

[5] A. K. Elmagarmid, P. G. Ipeirotis, and V. S. Verykios, "Duplicate record detection: A survey," *IEEE Trans. on Knowledge and Data Engineering*, vol. 19, no. 1, pp. 1–16, 2007.

[6] M. Bilenko and R. J. Mooney, "Adaptive duplicate detection using learnable string similarity measures," in *Proc. of ninth ACM SIGKDD Intl. Conf. on Knowledge discovery and data mining (KDD 2003)*, 2003, pp. 39–48.

[7] L. Shu, B. Long, and W. Meng, "A latent topic model for complete entity resolution," in *Proc. of 25th Intl. Conf. on Data Engineering (ICDE 2009)*, 2009, pp. 880–891.

[8] A. Takasu, "Information extraction by two dimensional parser," in *Proc. of 20th IEEE Intl. Conf. on Tools with Artificial Intelligence (ICTAI 2008)*, 2008, pp. 333–340.

[9] S. Sarawagi, "Information extraction," *Foundations and Trends in Databases*, vol. 1, no. 3, pp. 261–377, 2008.

[10] H. Bunke and P. Wang, Eds., *Handbook of Character Recognition and Document Image Analysis*. World Scientific, 1997.

[11] L. Getoor and C. P. Diehl, "Link mining: A survey," *ACM SIGKDD Explorations Newsletter*, vol. 7, no. 2, pp. 3–12, 2005.

[12] A. Takasu and K. Aihara, "Dvhmm: Variable length text recognition error model," in *Proc. of 16th Intl. Conf. on Pattern Recognition (ICPR 2002)*, 2002, pp. 110–114.

[13] A. McCallum, K. Bellare, and F. Pereira, "A conditional random field for discriminatively-trained finite-state string edit distance," in *Proc. of 21st Conf. on Uncertainty in AI (UAI)*, 2005, pp. 388–395.

[14] J. Lafferty, A. McCallum, and F. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proc. of 18th Intl. Conf. on Machine Learning (ICML 2001)*, 2001, pp. 282–289.

[15] M. Ohta, T. Yakushi, and A. Takasu, "Bibliographic element extraction from scanned documents using conditional random fields," in *Proc. of third Intl. Conf. on Digital Information Management (ICDIM 2008)*, 2008, pp. 99–104.

[16] M. Ohta, R. Inoue, and A. Takasu, "Empirical evaluation of active sampling for CRF-based analysis of pages," in *Proc. of 11th IEEE Intl. Conf. on Information Reuse and Integration (IRI 2010)*, 2010, pp. 13–18.

[17] M. Ohta and A. Takasu, "CRF-based authors' name tagging for scanned documents," in *Proc. of Joint Conf. on Digital Libraries (JCDL 2008)*, 2008, pp. 272–275.

[18] M. Ohta, T. Hachiki, and A. Takasu, "Using Web resources for support of online-browsing of research papers," in *Proc. of 10th IEEE Intl. Conf. on Information Reuse and Integration (IRI 2009)*, 2009, pp. 348–353.

[19] ——, "Related paper recommendation to support online-browsing of research papers," in *Proc. of fourth Intl. Conf. on the Applications of Digital Information and Web Technologies (ICADIWT 2011)*, 2011, pp. 130–136.

[20] Yahoo!Japan, "Yahoo!Search." http://search.yahoo.co.jp/ [Accessed: 08.05.2012]

[21] Wikipedia, "The Free Encyclopedia." http://ja.wikipedia.org/ [Accessed: 08.05.2012]

[22] J. M. Kleinberg, "Authoritative sources in a hyperlinked environment," *Journal of the ACM*, vol. 46, no. 5, pp. 604–632, 1999.

ISBN: 978-1-61208-220-2