Automatically Extracting Academic Papers from Web Pages Using Conditional Random Fields Model

Wei Liu, Jianxun Zeng
Institute of Scientific and Technical Information of China
China, 100038
Email: {liuw, zeng}@istic.ac.cn

***Abstract*—A huge amount of academic papers(including research reports) are being released in web pages. It is important to extract these papers in a structured way for many popular applications, such as science and technology information retrieval and digital library. However, few investigations have been done on the issue of academic paper extraction. This paper proposed a unified approach for automatically extracting academic papers from web pages based on CRF model. In the proposed approach, both academic paper extraction and semantic labeling are performed simultaneously by employing the theoretical Conditional Random Fields(CRF) model. Experimental results show that our approach can achieve significantly better extraction results.**

***Index Terms*—Web data extraction, Web intelligence, Machine learning, Conditional Random Fields**

I. Introduction

The rapid development of Web greatly facilitates the access and communication of science and technology information. According to our statistics, a huge amount of academic papers(including research reports) are presented on Web pages. As the important information source of many popular applications, such as science and technology information retrieval and digital library, the academic papers on web pages should be extracted in a structured way. However, at present, academic paper extraction is viewed as an engineering issue and is conducted manually. Therefore it is not affordable to web-scale applications. Our goal is to find an automatic way to address this problem. The work in this paper belongs to our NSTL(**N**ational **S**cience and **T**echnology **L**ibrary, http://www.nstl.gov.cn/NSTL/) project which aims to provide Chinese researchers the scientific literature information service.

Academic paper extraction belongs to the field of Web data extraction. A lot of efforts have been contributed to this field and shown the feasibility of automatic extraction of semantic data from the Web. And it is possible to use the techniques to extract academic papers. However, most of the existing methods (1) employed a specific machine learning model to identify each type of information independently or (2) built a number of extractors for different web page templates. It is ineffective for the two kinds of methods to perform academic paper extraction due to their natural disadvantages. In the first solution, a specific machine learning model has to be trained for each property of the academic paper. As a result, it is difficult to maintain many different models. Furthermore, the properties are (sometimes even strongly) dependent with each other. For example, *authors* can help recognition of *title* because *title* is always just on top of authors. Unfortunately, the separated models cannot take advantage of dependencies across the different properties. The second solution is template-dependent. However, accurately identifying Web pages for each template is not a trivial task because even Web pages from the same website may be generated by dozens of templates. Even if we can distinguish Web pages, template dependent methods are still impractical because the learning and maintenance of so many different extractors for different templates will require substantial efforts.

In this paper, we aim to conduct a thorough investigation on this issue. First, we formalize the problem. We view the problem as semantic labeling to the text blocks in Web pages, with each semantic representing one academic paper property. We employ the graphical model called Conditional Random Fields(CRF)[5] model to jointly optimize property detection and labeling. By using the vision-based page segmentation approach(VIPS)[15], which makes use of page layout features such as font, color, and size to construct a visual block tree for a Web page, we can get a better representation of a page compared with the traditional DOM tree. Given a visual block tree, property detection can be considered as the task of locating the minimum set of elements that contain some property. In this way, both property detection and labeling become the task of assigning labels to the nodes on a visual block tree, so they can be integrated in one probabilistic model. In contrast to existing solutions, our approach leverages the labeling results of attribute labeling for property detection, and at the same time benefits from the incorporation of some global features for attribute labeling. As a conditional model [17], CRF can efficiently incorporate any useful feature for Web data extraction.

Corresponding author: Wei Liu, liuw@istic.ac.cn

doi:10.4304/jsw.6.8.1409-1416

Neighborhood graphs for indexing and retrieving multi-dimensional data
USENIX
LISA '06 Paper
diabetes
Defective Suppressor Function in CD4⁺CD25⁺ T-Cells From Patients With Type 1 Diabetes
Research Article
Abstract

Figure 1. Examples of academic papers in web pages

The unified approach can achieve better performance in academic paper extraction than the separated methods, because the approach can take advantage of the interdependencies across the different properties. To the best of our knowledge, our approach is the first to formalize all the tasks of academic paper extraction in a unified approach and tackle all the problems simultaneous. Experimental results show that our approach outperforms the separated property extraction methods significantly. We utilized the profile information to help expert finding in social networks. Our contributions are: (1) formalization of the academic paper extraction problem, (2) proposal of a unified, template-independent extraction approach, and (3) empirical verification of the effectiveness of the proposed approach.

The rest of the paper is organized as follows. In Section 2, we discuss the schema of academic paper. Section 3 introduces vision-based web page processing. In Section 4, we explain the CRF models employed in our approach and in Section 5 we present the features used for CRF. The experimental results are reported in Section 6. Before concluding the paper in Section 8, we introduce related work in Section 7.

II. Schema of Academic Paper

Since the task of academic paper extraction is to extract and label the property of the academic paper, we define the schema of an academic paper as 15 properties: *title*, *author*, *affiliation*, *address*, *country*, *post code*, *email*, *publication date*, *abstract*, *category*, *general term*, *section*, *acknowledgement*, *reference*, and *appendix*. In fact, there are still some properties that have not been considered in this paper because they are infrequent and not important to our NSTL project.

section is the body of an academic paper, and the body of an academic paper usually consists of multiple *sections*. Among these properties, some are key, and some are optional. From another angle, some are unique, while some may appear multiple times in one academic paper. Table 1 compares these properties on the two angles.

TABLE I. The comparison among the properties of academic paper

Property	Key	Optional	Unique	Multiple
title	√	×	√	×
author	√	×	×	√
affiliation	×	√	×	√
address	×	√	×	√
country	×	√	×	√
post code	×	√	×	√
email	√	×	×	√
abstract	√	×	√	×
publication date	×	√	√	×
category	×	√	×	√
general term	×	√	×	√
section	√	×	×	√
acknowledgement	×	√	√	×
reference	√	×	×	√
appendix	×	√	√	×

From this table, we can find that a majority of the properties are optional or multiple. Therefore, academic paper extraction is non-trivial due to the uncertainty caused by the "optional" properties and the multiple properties. Given one academic paper, it is hard to predict whether an "optional" property and how many times a "multiple" property appears.

In addition, two characteristics of web pages make the extraction task more challenging. First, the templates of web pages from different web sites lack structural consistency. For instance, similar content might be represented by different tags in pages from different web sites. Second, web pages often contain lots of noise information, such as navigation bar and advertisements. In our approach, the noise information in web pages will

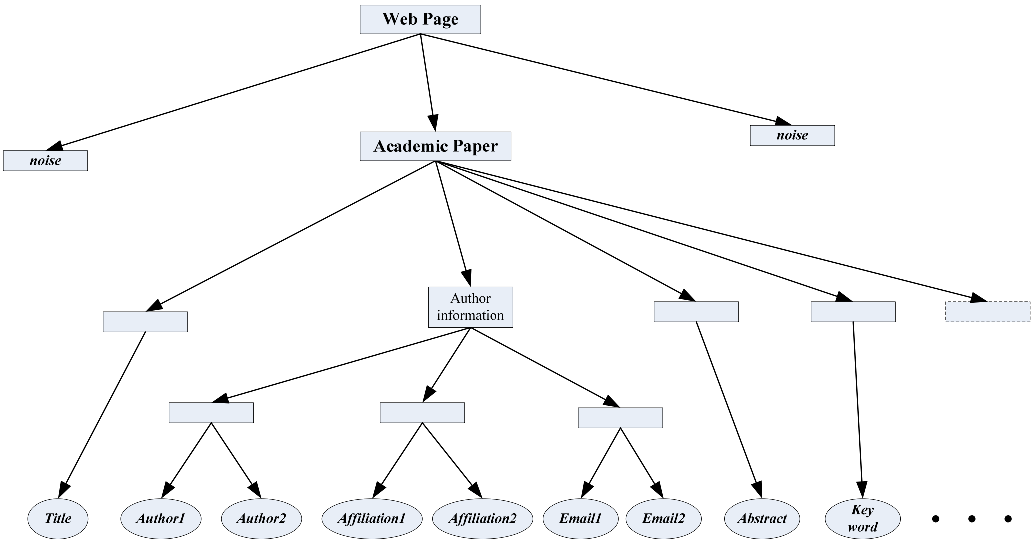


Figure 2. The vision block tree(partial) of the page in Fig. 1.

be treated as a special "optional" and "multiple" property. Compared with the papers in web pages, the papers in PDF format and WORD format are more clean and tidy. Especially, the file template provided by the journals and the conferences makes the extraction task simple and direct.

III. WEB PAGE PROCESSING: VISION-BASED SEGMENTATION

For Web data extraction, the first thing is to find a good representation format for Web pages. Good representation can make the extraction task easier and improve extraction performance. In most previous work, DOM tree, which is a natural representation of the tag structure, is commonly used to represent a Web page. However, as [15] pointed out, DOM trees tend to reveal presentation structure rather than content structure, and are often not accurate enough to discriminate different semantic parts in a web page. Moreover, since authors have different styles to compose web pages, DOM trees are often complex and diverse.

Meanwhile, people can still easily identify one paper from a web page at a glance despite the structural dissimilarities among pages of different templates. For example, when people browse the paper in Figure 1, they can quickly locate the properties without any difficulty, even in the case in which the paper is written in a foreign language. Motivated by this, [15] proposed a vision-based page segmentation (VIPS) approach. VIPS makes use of page layout features such as font, color, and size to construct a vision-tree for a page. It first extracts all suitable nodes from the DOM tree, and then finds the separators between these nodes. Here, separators denote the horizontal or vertical lines in a web page that visually do not cross any node. Based on these separators, the vision block tree of the Web page is constructed. Each node on this tree represents a rectangular region in the web page, which is called a block. The root block represents the whole page. Each inner block is the minimized rectangle that covers all its child blocks in the web page. All leaf blocks are atomic units and form a flat segmentation of the web page. Since vision block tree can effectively keep related content together while separating semantically different blocks from one another, we use it as our data representation format. Figure 2 is a vision block tree for the page in Figure 1, where we use rectangles to denote inner blocks and use ellipses to denote leaf blocks (or elements). Due to space limitations, the blocks denoted by dotted rectangles are not fully expanded.

Since the leaf blocks are atomic units, the extraction task is transformed into assigning possible semantic tags to each leaf block. The semantic tags correspond to the 15 paper properties and the "noise" tag(if a leaf block does not belong to any property). In our approach, we employ the theoretical Conditional Random Fields as the tagging model to perform the extraction task. In the left part of this paper, we will first introduce the CRF model, and then discuss the features of academic papers for CRF.

IV. CONDITIONAL RANDOM FIELDS(CRF)

To the best of our knowledge, several models have been applied for information extraction, such as Hidden Markov Models(HMMs)[20], Maximum Entropy Markov Models(MEMMs)[21], and CRF model. Recent studies[5] have shown that CRF outperforms the other two models because CRF can overcome their inherent defects (i.e. the output independence assumption for HMMs and the label bias problem for MEMMs). Therefore, we choose CRF as the tagging model perform the extraction task. In this section, we first introduce the linear CRF model, and then introduce the Hierarchical CRF which can incorporate hierarchical interactions. In our experiments, we will employ the two models separately and make the comparison between them.

A. Linear CRF

Conditional Random Fields are undirected graphical models. Its definition is given as follows.

Definition. *Let G* = (*V*, *E*) *be a graph such that* $Y=(Y_v)v\in V$, *so that Y is indexed by the vertices of G. Then* (*X*, *Y*) *is a conditional random field in case*, *when conditioned on X*, *the random variable* Y_v *obey the Markov property with respect to the graph*: $p(Y_v|X, Y_w, w\neq v) = p(Y_v|X, Y_w, w\sim v)$, *where* $w\sim v$ *means that w and v are neighbors in G.*

According to its definition, *X* is a random variable over data sequences to be labeled, and *Y* is a random variable over corresponding label sequences. All components *Yi* of *Y* are assumed to range over a finite label alphabet *Y*. CRF constructs a conditional model *p*(*Y*|*X*) with a given set of features from paired observation and label sequences. The conditional distribution of the labels *y* given the observations data *x* is represented as,

$$p(y\,|\,x)=\frac{1}{Z(x)}\exp\left(\sum_{e,k}\lambda_k t_k(y|_e,x)+\sum_{v,k}\mu_k f_k(y|_v,x)\right) \qquad (1)$$

where *x* is a data sequence, *y* is a label sequence, and $y|_e$ and $y|_v$ are the set of components of *y* associated with edge *e* and vertex *v* in the linear chain respectively; t_j and s_k are feature functions; parameters λ_k and μ_k correspond to the feature functions t_k and f_k respectively, and are to be estimated from the training data; *Z*(*x*) is the normalization factor, also known as partition function.

Since linear CRF is a sequence modeling framework, we use the leaf blocks in the visual block tree to construct the sequence. The construction process is very simple and direct: all leaf blocks are collected with the left-to-right order. In training, the CRF model is built with labeled data and by means of an iterative algorithm based on Maximum Likelihood Estimation. In training, the CRF model is built with labeled data and by means of an iterative algorithm based on Maximum Likelihood Estimation. In tagging, the model is used to find the sequence of tags *Y** with the highest likelihood $Y^* = \max_Y P(Y|X)$ using the Viterbi algorithm[22].

However, a visual block is actually the hierarchical representation of the data. Transforming the visual block tree into a flat sequence would lose the hierarchical interactions which are very important for web data extraction, which is just like that shown in Fig. 2. Based on such consideration, we adopt the hierarchical CRF as the substitution.

B Hierarchical CRF

The hierarchical CRF is proposed by [7]. In the hierarchical CRF, each node on the graph is associated with a random variable Y_i. We currently model the interactions of sibling variables via a linear-chain, although more complex structure such as two-dimensional grid can be used. The observations which are globally conditioned on are omitted from this graph for simplicity. HCRF assumes that every inner node contains at least two children. Otherwise, the parent is replaced with its single child. Notice that this assumption has no affect on the performance because the parent is identical to its child in this case. This assumption is made just for easy explanation and implementation.

The cliques of the graph are its vertices, edges and triangles. So, the conditional probability in formula (1) can be concretely expressed as,

$$p(y\,|\,x)=\frac{1}{Z(x)}\exp\begin{pmatrix}\sum_{e,k}\lambda_k t_k(y|_e,x)+\sum_{v,k}\mu_k f_k(y|_v,x)\\ +\sum_{t,k}\gamma_k h_k(y|_t,x)\end{pmatrix} \qquad (2)$$

where t_k , f_k and h_k are feature functions defined on three types of cliques (i.e. vertex, edge and triangle) respectively; λ_k, μ_k and γ_k are the corresponding weights; $v\in V$, $e\in E$ and *t* is a triangle. Although the feature functions can take real values, here we assume they are Boolean, that is, *true* if the feature matches and otherwise *false*.

In training, the spherical Gaussian prior with mean μ=0 and variance matrix $\Sigma = \sigma^2 I$ is used to penalize the log likelihood to avoid over-fitting. Tagging assignment can be done using the same junction tree algorithm[11] just with a simple modification of the two-phase schedule algorithm[12] by merely replacing the summation.

We refer the readers to [5] and [7] for more details about the two CRF models.

V. Features for CRF

In this section, we present the types of features that are essential for our extraction task. As we explain below, some of the features were first introduced as heuristic rules in some existing methods [31][32] on web data extraction. Our goal is to learn their importance weights. In practice, hundreds of features have been used in our experiments. Due to space limitations, we only list some features for each type.

A. Leaf Block Features

Text Features

a. Font features: font size, font color, font style, etc.

b. Text length features: the total number of words, the total number of sentences, the total number of lines, etc.

c. Frequent word features: the words that appear frequently in a property, such as “university” in *affiliation*.

d. Pattern features: some properties are usually presented with regular patterns, such as *email*.

e. Hyperlink features: Hyperlink text, URL, etc.

Spatial Features

a. Coordinates features: the horizontal distance from the block’s left border to the web page’s left border; the vertical distance from the block’s top border to the web page’s top border.

b. Size features: the width of the block; the height of the block.

c. Area feature: the area of the block.

Tree Features

a. Leaf distance feature: the distance of two properties. For example, the distance between *title* and *abstract* in Fig.2 is 7.

b. Level features: the level of a property; the level difference of two properties.

B. Inner Block Features

Inner block features can also be classified into the three types above. In each type, some inner block features are similar to leaf block's. We only present some peculiar features of inner blocks here.

Text Features

a. Font feature: the total number different fonts used in the block.

b. Information type features: plain text, list, table, equation, etc.

Spatial Features

a. Closeness feature: the closeness degree of the block's child blocks. In a vision tree, the area of a block usually does not equal to the area sum of the block's child blocks.

b. Gravity center feature: the information center of the block because the texts in the block are usually not even-distributed. The gravity center of the block is determined by the mass and the gravity center of the block's child blocks. For a leaf block, its gravity center of is the center of the area, and its mass is the sum of the texts.

Tree Features

a. Tree structure feature: the tree distance of two blocks as a measure of their structure similarity. The tree structure of two blocks is defined as the edit distance of their corresponding sub-trees. For the tree distance algorithm, the readers can refer to [18] for more details.

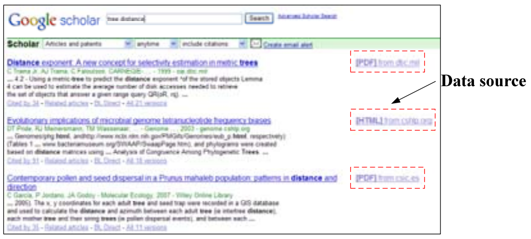
b. Relation feature: the relation of two blocks in the vision tree, which is father, sibling, ancestor, etc.

Linear CRF model only uses leaf block features, while hierarchical can use both leaf block features and inner block features. These features can be easily incorporated into CRF model by defining Boolean-valued feature functions. Finally, two sets of features are defined in the CRF model: transition features and state features. For example, a transition feature $y_{i-1}=y'$, $y_i=y$ implies that if the current tag is y and the previous tag is y', then the value is true; otherwise false. The state feature $w_i=w$, $y_i=y$ implies that if the token is w and the current tag is y, then the feature value is true; otherwise false.

VI. Experimental Results

In this section, we report our experimental results by applying the two CRF models to extract academic papers from web pages. A prototype system APE(**A**cademic **P**aper **E**xtractor) has been implemented for our experiments.

A. Data sets



Google scholar tree distance Search Advanced Scholar Search

Scholar Articles and patents anytime include citations Create email alert

Distance exponent: A new concept for selectivity estimation in metric trees [PDF] from dtic.mil
C Traina Jr, AJ Traina, C Faloutsos, CARNEGIE-... - 1999 - oai.dtic.mil
... 4.2 - Using a metric tree to predict the distance exponent *of the stored objects Lemma 4 can be used to estimate the average number of disk accesses needed to retrieve the set of objects that answer a given range query QR(oR, rq). ...
Cited by 34 - Related articles - BL Direct - All 21 versions

Evolutionary implications of microbial genome tetranucleotide frequency biases [HTML] from cshlp.org
DT Pride, RJ Meinersmann, TM Wassenaar, ... - Genome ..., 2003 - genome.cshlp.org
... Genomes/ping.html, andhttp://www.ncbi.nlm.nih.gov/PMGifs/Genomes/eub_g.html, respectively) (Tables 1 ... www.bacteriamuseum.org/SWAAP/SwaapPage.htm), and phylograms were created based on distance matrices using ... Analysis of Congruence Among Phylogenetic Trees. ...
Cited by 51 - Related articles - BL Direct - All 15 versions

Contemporary pollen and seed dispersal in a Prunus mahaleb population: patterns in distance and direction [PDF] from csic.es
C Garcia, P Jordano, JA Godoy - Molecular Ecology, 2007 - Wiley Online Library
... 2005). The x, y coordinates for each adult tree and seed trap were recorded in a GIS database and used to calculate the distance and azimuth between each adult tree (ie intertree distance), each mother tree and their siring trees (ie pollen dispersal events), and between each ...
Cited by 35 - Related articles - BL Direct - All 11 versions

Figure 3. Example of data source discovery

We collected 1000 web pages that contain academic papers from 10 web sites with a semi-automatic way. The process is three-stage. The first stage is data source discovery. Here a data source means one web site that has the web pages that contain academic papers. We adopt a smart method for this stage. GoogleScholar is a freely accessible Web search engine that indexes the full text of scholarly literature across an array of publishing formats and disciplines. Fig. 3 shows a snapshot of its search result page. In the search result pages, some hyperlinks(the texts in red-dashed rectangles in Fig. 3) are provided on the right part to guide users to download the full text of the paper. If the prefix of a text is "[HTML]", we select this web site as one data source. In this way, 10 data sources are collected easily. In the second stage, we implemented a focused crawler to collect web pages containing academic papers from each data source. A regular expression is defined for each data source. If the URL of one page satisfies the regular expression, this page will be downloaded. About 100 pages were collected from each data source. In the third stage, all the pages were checked manually to get rid of noise pages. As last, the left pages are stored as our data set. To make the comprehensive evaluation for our approach, the whole data set is divided into four parts: training set 1(***TrS1*** for short), training set 2(***TrS2*** for short), test set 1(***TeS1*** for short) and test set 2(***TeS2*** for short). Table 2 gives the description for them.

TABLE II. THE DESCRIPTION OF TRAINING SETS AND TEST SETS

Data set	Description
TrS1	Total 100 pages: 20 pages randomly selected from No. 1-5 data sources separately.
TrS2	Total 100 pages: 20 pages randomly selected from No. 6-10 data sources separately.
TeS1	Total 400 pages: the left pages in No. 1-5 data sources
TeS2	Total 400 pages: the left pages in No. 6-10 data sources

B. Evaluation metrics

For academic paper extraction, we use the same measures defined in [7]. The performance on each attribute is evaluated by precision (i.e. the percentage of returned elements that are correct), recall (i.e. the percentage of correct elements that are returned), and their harmonic mean F1. Their definitions are given below:

precision: the percentage of returned elements that are correct.

recall: the percentage of correct elements that are returned.

F1: the harmonic mean of precision and recall.

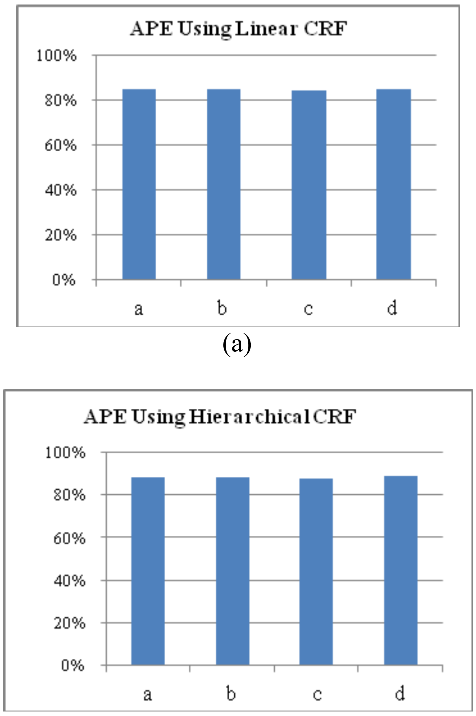
We also introduce a comprehensive evaluation metric **AVG_F1**: the average of **F1** values of different attributes.

C. Experiments on accuracy

We evaluated the performance of our approach with two CRF models separately. We also employ SVM(Support Vector Machine) as the baseline approach for performance comparison and conduct the experiment.

TABLE III. THE EXPERIMENTAL RESULTS ON ACCURACY, WHERE THE TRAINING SET IS *TrS1*+*TrS2* AND THE TEST SET IS *TeS1*+*TeS2*.

Property	SVM			Linear CRF			Hierarchical CRF		
	precision	**recall**	**F1**	**precision**	**recall**	**F1**	**precision**	**recall**	**F1**
title	0.916	0.932	0.924	0.935	0.941	0.938	0.937	0.943	0.940
author	0.781	0.663	0.717	0.847	0.792	0.819	0.896	0.852	0.873
affiliation	0.622	0.595	0.608	0.773	0.725	0.748	0.853	0.839	0.846
address	0.547	0.634	0.587	0.816	0.858	0.836	0.874	0.907	0.890
country	0.761	0.727	0.744	0.894	0.836	0.864	0.933	0.889	0.910
post code	0.846	0.808	0.827	0.848	0.817	0.832	0.907	0.898	0.902
email	0.918	0.873	0.895	0.930	0.908	0.919	0.972	0.950	0.961
abstract	0.783	0.807	0.795	0.838	0.865	0.851	0.869	0.893	0.881
publication date	0.775	0.747	0.761	0.852	0.875	0.863	0.866	0.875	0.870
category	0.639	0.603	0.620	0.806	0.780	0.793	0.839	0.802	0.820
general term	0.685	0.628	0.655	0.854	0.871	0.862	0.863	0.897	0.880
section	0.750	0.726	0.738	0.866	0.904	0.885	0.925	0.944	0.934
acknowledgement	0.857	0.813	0.834	0.862	0.838	0.850	0.968	0.951	0.959
reference	0.793	0.866	0.828	0.904	0.929	0.916	0.922	0.938	0.930
appendix	0.733	0.785	0.758	0.776	0.792	0.784	0.805	0.819	0.812
AVG_F1	0.753			0.853			0.894		



(a)

(b)

Figure 4. Template independency evaluation for APE, where a, b, c and d refer to the four parts of this experiment: (a) *TrS1* for training and *TeS2* for testing; (b) *TrS1* for training and *TeS1* for testing; (c) *TrS1* for training and *TeS2* for testing; (d) *TrS2* for training and *TeS2* for testing.

The features used in SVM are same to linear CRF and hierarchical CRF. Table 3 shows the experimental results. As we can see from Table 3, two important conclusions can be made. First, our CRF-based approach outperforms the baseline approach. The baseline approach only considers the state features and loses the insight of the transition features, while our CRF-based approach can combine the two types of features effectively. For example, the extraction accuracy of *affiliation* depends on those of its neighboring properties, such as *author* and *abstract*. As a result, our CRF-based approach has an evident improvement compared to the baseline approach. Second, hierarchical CRF is better than linear CRF on some properties. For example, hierarchical CRF can incorporate hierarchical interaction between *author information* and *author*(see Fig. 2). Unfortunately, linear CRF suffers from ignorance of such hierarchical interactions.

D. Experiments on template independency

It is widely known that one of challenges on Web data extraction is the web pages to be extracted are often generated with different templates. Hence template independence is one of the most important requirements to Web data extraction tools. In this section, we evaluate the template independence of our approach. The experiment is conducted under a reasonable assumption that the web pages from different web sites are also generated with different templates. Based on this assumption, this experiment consists of four parts: (a) ***TrS1*** for training and ***TeS2*** for testing; (b) ***TrS1*** for training and ***TeS1*** for testing; (c) ***TrS1*** for training and ***TeS2*** for testing; (d) ***TrS2*** for training and ***TeS2*** for testing. Obviously, ***TrSi*** and ***TeSi*** are from same web page templates, while ***TrSi*** and ***TeSj*** are not. If the extraction accuracies of the former and the latter are close to each other, we can conclude that our approach is template independent, otherwise our approach is template dependent. As it can be seen from Fig. 4, our approach shows the good characteristic of template independency, whether employ linear CRF or hierarchical CRF.

E. Experiments on efficiency

The extraction speed is also another important factor to Web data extraction tools in practice due to web scale. In this part, we evaluate the efficiency of APE, i.e. the number of web pages processed per unit time. Averagely, building vision block tree(web page processing) requires 0.27 second, and extracting academic paper properties from vision block tree requires 0.07 second. We can conclude that (1) the extraction speed is about 3 pages per second; (2) most time is spent on web page processing. We admit the current extraction still cannot meet the requirement of real applications. But the efficiency can be significantly improved if web page processing speeds up.

VII. Related Work

The problem studied in this paper belongs to the field of web data extraction. It has received a lot of attentions in recent years. Survey[8] has given a good summary for these efforts. The research efforts in this field are either template-dependent[3,4,9,10,18] or template-independent[6,7,13,14,17]. In this section, we give a brief introduction for them first. Then, the works on news article extraction will be introduced and compared.

Template-dependent works mainly focus on extracting structured data records and data items in the web pages through inducing the common template. Most of them utilize the structure information on the DOM tree of a web page to represent the templates of similar web data records. In recent works, some visual features are also combined with the DOM tree to improve the performance, such as the method introduced by ViNTs[4]. [23] only utilizes pure visual features to implement the structured data extraction from deep web pages. However, the generated wrappers are sensitive by those works can only be applied for the web pages that share similar templates, and are not practical for the task of academic paper extraction from general web sites. In addition, an annotation task is needed to assign right semantics for the extracted data. Template-independent works aim to extract structured data from different-template web pages. Most of these methods are based on probabilistic models, which integrate semantic information and human knowledge in inference. For example, Conditional Random Fields (CRF)[5] and its variations(such as 2D-CRF[6], HCRF[7] and Semi-Markov CRF[14]) infer the semantics of the text blocks in web pages by learning the order dependencies of web data distribution. Their distinct advantage is insensitive to the templates of web pages. They are focusing on assigning the semantic label to the extracted data and can be seen as complementary to template-dependent works[6]. In addition, template-independent works are not suitable for the rich-noise web pages because the too many noises will significantly weaken the dependency of web data distribution.

Academic paper extraction is a special topic in the field of web data extraction. Until now, several works have been proposed for special genre articles extraction(e.g. web news articles), but most of them only focused on article body extraction. [2] proposes a top-down approach to generate a tree-structured wrapper. This approach is template-dependent, so it is not practical when web pages come from different web sites. [1] is a template-independent approach for news content extraction based on the state features. But for most academic paper properties, such as *affiliation*, the performance will be poor if only their state features are considered. Our approach is different to them on both application and technique. For application, our approach targets at extracting multiple academic paper properties not only its body. For technique, our approach utilizes both the transition features and the state features, which can improve the extraction performances of properties in a unified way.

VIII. Conclusion

In this paper we propose a unified approach to extract academic papers from web pages based on CRF model. The extensive experiments show the effectiveness of our approach. We believe it is a promising way to improve the extraction performance by exploiting the neighboring relations among the academic paper properties.

ACKNOWLEDGMENT

The research is under the support of Advance Research Foundation of Institute of Scientific and Technical Information of China under grant No. YY-201005, the China Postdoctoral Science Foundation funded project under grant 20080440256 and 200902014.

REFERENCES

[1] S. Zheng, R. Song, J.-R. Wen: Template-Independent News Extraction Based on Visual Consistency. *AAAI* 2007: 1507-1511

[2] D. Reis, P. Golgher, A. Silva: Automatic web news extraction using tree edit distance. *WWW* 2004: 502-511

[3] Y. Zhai, B. Liu: Web data extraction based on partial tree alignment. *WWW* 2005: 76-85

[4] H. Zhao, W. Meng, Z. Wu: Fully automatic wrapper generation for search engines. *WWW* 2005: 66-75

[5] J. D. Lafferty, A. McCallum, F. Pereira: Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. *ICML* 2001: 282-289

[6] J. Zhu, Z. Nie, J.-R. Wen.: 2D Conditional Random Fields for Web information extraction. *ICML* 2005: 1044-1051

[7] J. Zhu, Z. Nie, J.-R. Wen: Simultaneous record detection and attribute labeling in web data extraction. *KDD* 2006: 494-503

[8] C.-H. Chang, M. Kayed, M. R. Girgis, K. F. Shaalan: A Survey of Web Information Extraction Systems. *IEEE Trans. Knowl. Data Eng*. 18(10): 1411-1428 (2006)

[9] V. Crescenzi, G. Mecca, P. Merialdo: RoadRunner: Towards Automatic Data Extraction from Large Web Sites. *VLDB* 2001: 109-118

[10] B. Liu, R. L. Grossman, Y. Zhai: Mining data records in Web pages. *KDD* 2003: 601-606

[11] R. G. Cowell, A. P. Dawid, S. L. Lauritzen, D. J. Spiegelhalter: Probabilistic Networks and Expert Systems. Springer, 1999.

[12] F. V. Jensen, S. L. Lauritzen, K. G. Olesen: Bayesian updating in causal probabilistic networks by local computation. *Computational Statistics* Quarterly, 4:269-82, 1990.

[13] J. Wang, X. He, C. Wang, J. Pei, J. Bu, C. Chen, Z. Guan, G. Lu: News article extraction with template-independent wrapper. *WWW* 2009: 1085-1086

[14] S. Sarawagi, W. W. Cohen: Semi-Markov Conditional Random Fields for Information Extraction. *NIPS* 2004

[15] Cai, D., Yu, S., Wen, J.-R. and Ma, W.-Y., VIPS: a vision based page segmentation algorithm, *Microsoft Technical Report*, MSR-TR-2003-79, 2003

[16] L. Yao, J. Tang, J.-Z. Li: A Unified Approach to Researcher Profiling. *Web Intelligence* 2007: 359-366

[17] D. Pinto, A. McCallum, X. Wei, and W. B. Croft. Table extraction using conditional random fields. *SIGIR* 2003, 235-242

[18] H. Touzet: A Linear Tree Edit Distance Algorithm for Similar Ordered Trees. *CPM* 2005: 334-345

[19] J. Zhu, Z. Nie, B. Zhang: Dynamic hierarchical Markov random fields and their application to web data extraction. *ICML* 2007: 1175-1182

[20] Z. Ghahramani and M. I. Jordan. Factorial hidden Markov models. *Machine Learning*, Vol.29, 1997, pp. 245-273

[21] A. McCallum, D. Freitag, and F. Pereira. Maximum entropy Markov models for information extraction and segmentation. *ICML* 2000: 591-598

[22] M.S. Ryan and G.R. Nudd. The Viterbi Algorithm. Technical Report University of Warwick RR-238. 1993

[23] W. Liu, X. Meng, W. Meng: ViDE: A Vision-Based Approach for Deep Web Data Extraction. *IEEE Trans. Knowl. Data Eng.* 22(3): 447-460 (2010)

Wei Liu was born in Shandong Province of China. He received the M.S. degree in Computer Science from School of Computer Science and Technology at ShanDong University in 2004, and the Ph.D. degree in Computer Science from School of Information at Renmin University of China in 2008. He did his Postdoc research with Prof. Jianguo Xiao in Institute of Computer Science & Technology at Peking University. His current research interests include Web data extraction, Deep Web data integration, and Science and technology information retrieval.

He is currently an assistant researcher at the information resource centre, Institute of Scientific and Technical Information of China. He has published over 20 papers in some reputational international conferences and journals, such as TKDE, PRICAI2010, etc.

Jianxun Zeng was born in Jiangxi Province of China. He received the M.S. degree in Library and Information Science from School of Information Management at Wuhan University. His current research interests include Information analysis and research and Information resource construction. He is currently a research librarian at the information resource centre, Institute of Scientific and Technical Information of China.

He has published over 70 papers and has presided over many national research projects.