A Survey of Automatic Deep Web Classification Techniques

Umara Noor
Center of Information Technology, Institute of Management Sciences, Pakistan

Zahid Rashid
School of Electrical Engineering and Computer Science, NUST, Pakistan

Azhar Rauf
Department of Computer Science, University of Peshawar, Pakistan

ABSTRACT
To devise vision of the next generation of the web, deep web technologies have gained larger attention in a last few years. An eminent feature of next generation of web is the automation of tasks. A large part of Deep web comprises of online structured domain specific databases that are accessed using web query interfaces. The information contained in these databases is related to a particular domain. This highly relevant information is more suitable for satisfying the information needs of the users and large scale deep web integration. In order to make this extraction and integration process easier, it is necessary to classify the deep web databases into standard\ non-standard category domains. There are mainly two types of classification techniques i.e. manual and automatic. As the size of deep web is increasing at an exponential rate with the passage of time, it has become nearly impossible to classify these deep web search sources manually into their respective domains. For this purpose, several automatic deep web classification techniques have been proposed in the literature. In this paper apart from the literature survey, we propose a framework for analysis of automatic classification techniques of deep web. The framework provides a baseline for the analysis of rudiments of automatic classification techniques based on the parameters such as structured, unstructured, simple/advance query forms, content representative extraction methodology, level of classification, performance evaluation criteria and its results. Furthermore, we studied a number of automatic deep web classification techniques in the light of proposed framework.

General Terms
Deep web; Classification; Survey

Keywords
Deep web; web databases; data integration; domain concepts; Survey

1. INTRODUCTION
World Wide Web comprises of surface web and deep web. The surface web (also known as the visible web or indexable web) consist of that portion of World Wide Web which is indexed by conventional search engines. Deep web (also called deepnet, the invisible web, dark web or the hidden web) refers to contents hidden behind HTML forms; normally made up of domain specific databases, dynamic content , unlinked content, private web, contextual web, limited access content, scripted content, non-HTML/text content[5]. The data lies in deep web cannot be crawled and indexed by conventional web search engines. Information underlying deep web sites can only be accessed through their own query interfaces and results are produced dynamically in response to a direct request. Deep web contains more information as compared to surface web. It was estimated in a survey in 2001 that, there are 43,000- 96,000 "deep web sites" and an informal estimate of 7,500 terabytes of data exist in deep web compared to 19 terabytes of data in the surface web [6]. In another study an increase of 3-7 times in the volume of deep web was observed during 2000 – 2004 [10].

What makes the deep web so significant is the quality of the contents found within. The total quality content of the deep web is at least 1,000 to 2,000 times greater than that of surface web. Deep web contents may be relatively highly relevant to information needs of the users. More than half of the deep web content resides in topic specific databases and these search engines yield most relevant results as the data contained by them is naturally clustered. Comparison of query results retrieved from general purpose search engines and domain specific search engines (which are part of deep web) indicate a three-fold improved likelihood for obtaining quality results from the deep web as for the surface web [6]. Domain specific search sources focus on documents in confined domains such as documents about an organization or in a specific subject area. Most of the domain specific search sources consist of organizations, libraries, businesses, universities and government agencies.

In our daily life we are provided with several kinds of database directories to store crucial records. For example a telephone directory stores an organized record of landline telephone numbers. Similarly to locate a particular site in the ocean of WWW there have been efforts to organize static web content in the form of web directories i.e. Yahoo, dmoz. The procedure adopted is both manual and automatic. Similarly to organize myriad deep web databases, we need a grand database to store information about all the online deep web databases. Efforts have been proposed to organize deep web source into category domains. Due to numerous deep web sources, automatic classification is getting popularity in the recent few years. A few aspects which make the task of automatic organization of deep web sources indispensable are: first deep web content is highly relevant to user's information needs, second for business intelligence tasks, information integration of the deep web data sources can lead to economic peak, and third the realization of the semantic web can be made possible.

This paper surveys the automatic deep web organization techniques proposed so far in the light of their effectiveness and coverage. The key characteristics significant to both exploring and integrating deep Web sources are thoroughly discussed. The contributions made in this paper are:

1) A frame work of automatic deep web classification is devised. The essentials of the surveyed approaches are discussed under the umbrella of the proposed framework.

2) Our study contributes a set of parameters on the basis of which a detailed comparative analysis between the underlying approaches is performed and the strength and effectiveness of the classification approach is judged. Our key findings are summarized in table 2 of the appendix.

3) Finally we critically analyze the overlap among the key parameter.

The rest of the paper is organized as follows: Section 2 describes the Deep Web Classification framework. In Section 3 the anatomy of deep web is discussed. In Section 4 classification techniques along with their rudiments are thoroughly analyzed in the light of the proposed framework. Section 5 concludes the survey along with some thoughts of evolution in learning method of deep web classification technique.

2. DEEP WEB CLASSIFICATION FRAMEWORK

Broadly there are two approaches of deep web classification: manual and automatic. In manual approach, deep web sources are manually classified and their links are archived in commercial deep web directories such as CompletePlanet [3], InvisibleWeb [4] and SearchEngineGuide [16]. The obvious advantage of the manual classification is the development of a high quality directory service but the approach can't be adopted due to the following limitations.

- It involves human intervention which make it time consuming and expensive.
- It is non-scalable as the population of deep web sites is increasing at an exponential rate. CompletePlanet that claims to be the largest deep web directory (comprising over 70,000 deep web sources) covers only 16% of the whole deep web [1].

Due to the vast and dynamic nature of deep web in which new databases are constantly being added and old ones removed or modified, the classification process for finding online deep web sources must be automated in order to discover the searchable forms that serve as entry points to those deep web data sources. For the purpose we devise a framework for automatic deep web classification as shown in figure 1.

Here is the formal definition of the proposed framework.

Definition 1: *Given a set of deep web data sources Di where i=1,…,n, and a set of Category domains CDi where i= 1,…,n,. A set of procedures Pi where i = 1,…,n, are adopted to extract the content representative (CR) from each deep web source. CR is submitted to a Classifier (C), which results in the classification of Di into one or more category domains CDi.*

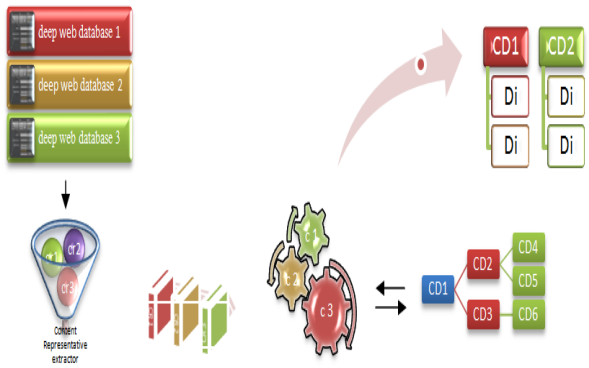

Figure 1: General Deep Web Classification Framework

3. ANATOMY OF DEEP WEB SOURCES

Conceptually deep web is viewed in the form of online distributed databases which may be interconnected. Deep web is discovered through the query interface provided on the surface of the web. Three main ingredients to describe the deep web are a site, database and query interface. A Web site is a deep web site if it has at least one query interface further a deep web site can have more than one database connections [10]. A closer look at the anatomy of deep web sources tells about the coverage and scalability of a deep web source.

3.1 Deep Web Databases

Deep web databases are either structured or unstructured. Structured databases provide structured query interfaces and results in which data objects are structured records with multiple fields e.g. 'a Book database like Amazon.com queries and returns books with author, title and ISBN etc. Most of the work done regarding deep web classification focuses on structured data sources.

Unstructured databases comprises of unstructured data objects e.g. text, images audios and videos in response to user queries. The term 'text database' is often used to refer unstructured databases.

The distinction between structured and unstructured sources is important as to classify both different paradigms and approaches are employed. Structured deep web sources are more significant both quantitatively and qualitatively than unstructured deep web sources as they comprise 80% of the whole deep web and their integration produces meaningful outcome for business intelligence tasks. A recent study also indicates a ratio 3.4:1 of structured and unstructured sources respectively [1].

3.2 Query Interface Type

In order to determine whether a deep web source is structured or unstructured the query interface is examined. This study examines two kinds of deep web query interfaces: *Simple* and *Advanced*. A simple query interface is the one which comprise of less number of attributes without enough understanding of attribute labels and usually accept key word based queries. In other words such interfaces do not usually contain enough visible features on the surface of query form to guide the automated classifier in the classification process. Such interfaces

are commonly found on the web and their classification is more challenging as compared to classification of advanced query form [12]. An example of such a query interface is shown in the figure 2. Moreover the approach proposed by Sahami et al. in [9] discusses classification of simple query interface forms.

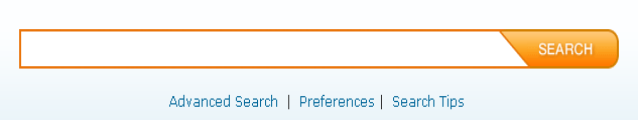

Figure 2. A Simple Query Interface

On the other hand, an advanced query interface comprises of a sufficient number of attributes, well defined attribute labels along with default input values and enough visible features on the query form as a guide to the searcher and to an automated classifier. An example is a query interface for book search with text inputs as title, author, ISBN etc. the technique proposed by H. Xu et al. in [11] deals with advance query interfaces for classification.

Definition 2: (Query Interface)

Q denotes the query interface with a set of attributes A=(ai,), i=1,…….,n and attribute values defined by their data types V=(vi), i=type 1,…….,type n.

Definition 3: (Result Page)

R denotes the result page retrieved in response to user's query.

A common observation is that behind the simple query interface there exist unstructured databases and behind the advanced query interface there exist structured databases. Although there are some exceptional overlapping zones, but generally we can classify the techniques encountering structured web sources as having advanced query forms unless specified explicitly as in [12]. Thus the technique proposed in [2],[13],[14], [17],[18],[19] works with advanced query forms.

The technique proposed by Barbosa et al. in [7] works with both simple and advance search query interfaces. Visible features of both page and form attributes are considered as content representative of the deep web data source.

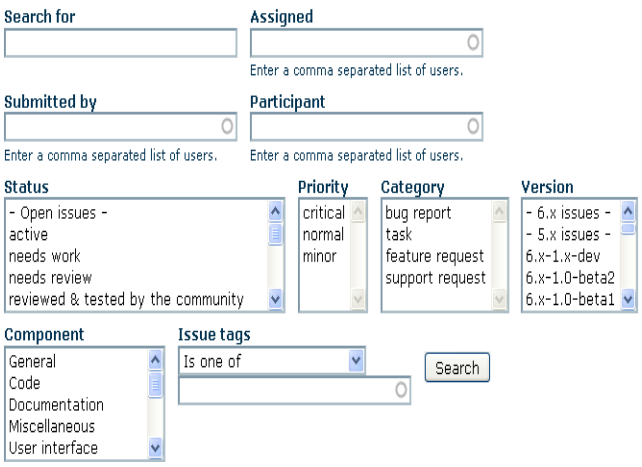

Figure 3. An Advance Query Interface

4. DEEP WEB CLASSIFICATION APPROACHES

Several deep web classification approaches are proposed in the literature [2],[7],[9],[11],[12],[13],[14],[17],[18],[19]. Here we provide a comprehensive comparative analysis of the above approaches regarding their organization approach, content representative extraction methodology, granularity of category domains, classification methodology and performance evaluation metrics. Our findings are summarized in table 2.

4.1 Organization Approach

There are two basic kinds of classification: supervised and unsupervised. All of the supervised classification methods share the same property – they use machine learning techniques to train the classifier using a provided pre-classified training set to later perform the classification of previously unseen data sources. During the training process, a classifier gathers knowledge that is essential for distinguishing categories based on data source features. The rest of classification is performed on the bases of acquired knowledge. Here in this study the techniques proposed in [9],[12],[13],[14],[18] classify deep web sources using supervised classification approach. In the recent years innovation to traditional supervised classification approach is introduced by the enhancement of data source features using some external knowledge sources. In the deep web classification technique [11], features are enhanced through building ontology model for each category domain.

On the other hand unsupervised methods do not use any kind of external information for classification. They focus on discovering those features of the data source that enable them to assemble similar data sources into coherent groups, called clusters. The [2],[7],[17],[19] techniques of deep web classification incorporate unsupervised classification approach i.e. clustering.

4.2 Content Representative Extraction

To classify a deep web database in a domain we need to extract the content representative from each data source. A strong representative which briefly encounters all the majors of a data source helps in efficient classification. Formal definition for content representative of a deep web source is given below:

Definition 4: (Content Representative)

Given a deep web source Di, a procedure Pi is employed to extract a sample CR from Di. CR represents the whole population of Di .

There are two basic approaches for extracting content representative from a deep web data source: through Query Probing and Visible form features. The former is also known as post-Query and later as pre-Query [12].

4.2.1 Query Probing

Query Probing is a very common method adopted by most of the supervised deep web classification approaches [9],[12],[13],[14]. In query probing, queries are designed through special techniques and are probed against the deep web databases. The results retrieved based on query probing determine the content representative to that web database. Query probing approach is effective regarding simple, keyword based query interfaces as it can't be easily adopted for multi-attribute structured query interfaces [7].

The technique proposed by Sahami et al. in [9] is based on training a rule-based document classifier in order to generate probing queries. The queries are submitted to the hidden web sources to extract their content representative. The count (no. of matches found) serves as the content representative of the database, used to determine the category of the data source based on the parameter of coverage and specificity.

In the technique proposed by X. Xian et al., queries are probed but instead of extracting the 'count' the result pages are successfully retrieved. Schema extracted from the result page ultimately becomes the content representative of the deep web source [12].

In the technique proposed by W. Su et al., the queries are manually formulated from the titles of the nodes in a topic hierarchy. The wrapper extracts the count from the result page which is submitted to the classifier. In the scenario where query interface has more than one edit box, the one with the largest number of reported count is selected. To measure the database's content distribution over the topics the counts are normalized rather than having absolute values. This helps resolves the coverage and specificity issue related to data sources [13].

In the technique proposed by T. Nie et al., queries are generated by training sample instances for the subject specific categories. The sample instances are gathered from the static web pages and query result pages. The data scale of the query result is the content representative of the data source and is employed to determine its category [14].

4.2.2 Visible Form Features

Sometimes content representative is obtained from the surface of the deep web form instead of probing beneath the surface of the deep web. Such representative is in the form of visible features found on the form page, query schema or on the deep web site. Visible form features are easy to crawl and possesses discriminative properties for clusters to be classified in. Query schemas are believed to be the right representative for the structured deep web sources due to their availability on the "surface" of online databases [2].

The deep web classification approaches that choose query schema of the deep web source as a content representative are [2],[11],[14],[17],[18],[19]. In these approaches, the attribute labels along with their data types are extracted for classifying a data source in a domain. All these approaches intrinsically cover only structured data sources as they deal with features of advance query interfaces. However, Barbosa et al in [7] employs a broader set of metadata associated with the deep web data source i.e. the text on the form page, the attribute labels of the form i.e. query schema and the hyperlink structures around the form page. Such extracted content works for both simple/advance search query interfaces and structured/unstructured deep web sources. Thus the content representative extraction approach proposed in [7] provides highest deep web coverage as compared to the rest of the approaches.

4.3 Granularity of Category Domains

This parameter describes the granularity of the categories utilized in the classification process. Here in this study we refer to the granularity of category domains by the terms macro and micro.

Macro category domains are more general and coarse. They represent general domain concepts e.g. Books, movies, music etc. There are several drawbacks of macro level classification: The number of categories is limited. Domain is a general concept that just implies the most basic functions of a source [14]. It is insufficient to answer user's query and compromises important content during information integration. Thus to encounter more granular concepts we need micro category domains. Micro categories also termed subject oriented categories or simply subcategories are more detailed and subject related e.g. a book category can be diversely distributed as cooking books, art books, science books etc. Further for science books there can be several types i.e. computer science, biological sciences etc. This hierarchy goes down to encounter more minor concepts. Formal definition for macro and micro category domains is given below:

Definition 5: (Category Domains)

A macro category domain CDi can be further partitioned into 'n' micro subcategories denoted as cdi where i= 1…n.

All the deep web classification approaches proposed in [2],[7],[9],[11],[12],[13],[17],[18],[19] classify deep web sources under macro category domains except [14] which classifies deep web sources under micro category domains.

4.4 Classifier (Classification Method)

Here in this section we briefly discuss the methodology employed to classify deep web sources in the surveyed approaches and its effectiveness regarding classification.

The Classification method proposed by Sahami et al. in [9] is based on training a rule-based document classifier in order to generate probing queries. RIPPER (a tool developed at AT&T Research Laboratories) is used to develop a classifier. A training set of categories along with their pre-classified documents is provided and the tool returns a classifier comprising of rules for each category. To extract the content representative of the hidden web sources, each rule is turned into a query. The number of matches for each query will be the number of documents in the web source that satisfy the corresponding rule. The parameters defined to classify deep web sources in one or more categories are coverage and specificity. Coverage defines the "accurate" amount of information that a data source contains about a specific category. Specificity defines how "focused" a data source is on a given category. After the probing phase and approximation of the coverage of a database for the pre-defined categories is calculated. To calculate the specificity of the database, the size of the database is required, which is approximated by $|D| \simeq \sum_{i=1}^{k} n_{i\cdot}$. This is just a rough approximation and do not determine the real size of the database. As it is just the sum of all the counts returned from the query probes. Due to the overlap of matched results the size of the database approximated may be much greater than the original size. This classification technique provides the lowest level coverage as it encounters only unstructured sources. The number of matches found i.e. "count" is the weakest source of obtaining content representative.

B. He et al. in [2] proposed model-differentiation as a new objective function for clustering, which allows principled statistical measure for determining cluster homogeneity. The problem of deep web organization is abstracted as the clustering

of categorical data. A new similarity measure is derived for the HAC algorithm. Statistical hypothesis testing is performed for which pre-clustering and post-classification techniques are designed. The approach proposed yields effective classification of structured data sources on macro category domains and considers attribute aggregate instead of semantic relations between the attributes.

W. Su in [13] proposed a hierarchical classification method that classifies structured deep web data sources into a predefined topic hierarchy automatically using a combination of machine learning and query probing techniques. Human classified sources are used as training set at each node of the topic hierarchy. Queries are constructed from the titles of the topics in the hierarchy. The query result "count" of the training data sources is used to train and construct a support vector machine classifier for each internal node of the topic hierarchy. Whenever a new deep web data source is to be classified, the same set of queries is posed, and the SVM classifier is utilized to classify the database into the hierarchy. The major limitation observed in the above approach is the construction of queries from the titles of the topic hierarchy. As such queries are simply keyword based and have several compatibility issues in coping with structured data sources. Thus the above scheme is overall not much effective in dealing with structured data sources.

The classification method proposed by H. Xu et al. in [11] is based on constructing a category ontology model and VSM for the deep web sources. The classifier is trained based on predefined domains of deep web interface schemas. Each new deep web source is classified in that pre-defined domain. Ontology model is defined for each category domain. The rationale behind this act is the argument that for structured deep web sources in each domain there are limited number of attributes of the interface schema [15]. The ontology model defines: a set of interface schemas of deep web sources; a set of attributes along with their label and type of interface schema; a special characteristic of attributes whether it is exclusive in a domain, shared in multi-domains or is a noise (contributes nothing to the classification process); conceptual partition of attribute in a specific domain, computed by encountering the semantic relation among the entries such as synonym, hypernym/hyponym, meronym/ holonym and homonyms etc; a reference function for mapping attribute to a concept; a set of pre-defined domains; a reference function to map a concept to a pre-defined domain and a root to capture all those entries that cannot be classified in any pre-defined domains. A deep web vector space model is built to compute similarity among the deep web interfaces. Each individual vector defines a group of features and their related weights. The concepts of the ontology model mapped by attributes are selected as features and their weights are calculated by using a novel technique i.e. DWTF, based on feature frequency of the deep web sources.

In their work, Barbosa et al. in [7] casts the problem of deep web data source classification as document clustering. The task of clustering is performed on a large set of metadata objects present on the "surface" of the deep web source. As not all the information is relevant, to identify the relevant data objects a form-page model is defined. Which represent the textual information associated with the deep web data source and models the importance of individual terms. Form-page model contains the record of page contents and form contents. To generate homogenous clusters, the identification of relevant terms is important, thus TF-IDF measure is used to model the importance of terms and to eliminate noise. To compute the similarity between the form pages, cosine similarity measure is used. The proposed algorithm CAFC-C uses k-means as the basic clustering strategy. Further to improve the clustering process hyperlink structures are utilized. The quality of the clusters is measured through entropy and F-measure. The F-measure provides a combined measure of precision and recall. This technique provides the highest coverage as compared to the rest of the techniques as it covers both structured and unstructured deep web sources and works with both simple and advance query interfaces.

The method proposed by X. Xian et al. in [12] is based on a combination of query probing and SVM learning techniques. The integral part of the framework is a domain specific classifier (DSC) which is constructed by using the features extracted from advanced query interface (forms) in domain. On the basis of a large number of observations, it is analyzed that the result schema is a good indicator of the database domain and contains the metadata and data. Also there is a great big similarity between result schema and interface schema. So this becomes the rationale of constructing a DSC by using the features of advanced query forms. The first step is to retrieve simple query interface forms through focused crawler. The proposed framework consists of three components: a probe query, the result schema extraction (RSE), and the domain specific classifier (DSC). The probe query model submit a series of random queries to simple query interfaces, result pages are successfully retrieved. RSE is an instance-based result schema extraction method to extract schemas from result pages. DSC makes use of the result schemas of web databases to identify among searchable forms, the ones that belong to the target database domain. For each domain, positive and negative advanced forms are manually selected as the training set, and positive and negative web databases with simple query interface are selected as the testing set. Three kinds of evaluation metrics are used: Recall, Precision and F-measure.

In their paper, T. Nie et al. [14] address micro category domain deep web classification approach. Both query probing and visible form features are used to extract content representative from the deep web data source. The classification task is performed in two stages. In the initial stage, the existing sources are categorized. The second stage is the increasing stage, in which new sources are categorized after the initial stage. Visible form features are utilized to extract the most basic function of each data source. Thus the technique only works with advanced query interfaces. For query probing, a set of sample instances are trained for existing subject categories. Instances are gathered from both static web pages and query result pages by using automatic wrapper techniques to extract structured data from web pages. Sample instances are obtained using two methods: one is manual training samples for each category. Another uses existing works to extract instances from web pages.

In their paper, P. Lin et al. [17] proposes a new similarity computing algorithm i.e. literal and semantic based similarity computing (LSSC) to compute similarity among deep web query interfaces. Further LSSC is combined with NQ clustering technique to cluster deep web query interfaces. To perform the

task, the schema characteristics of query interfaces and common attributes in a same domain are analyzed thoroughly. Based on the analyzed observations a new representation of query interface is formulated i.e. form term and function term. The form term is the literal information in the form that is used to describe corresponding controls, and this information can be searched by search engines; the function term is the control information that can't be searched by search engines but the information in that can be used to cluster web forms. The common attributes of a domain form features of that domain. After the integration of query interfaces, every cluster is matched with these features to appoint clusters to their corresponding domains.

For convergence in results of deep web classification domains, H. Le et al. [18] investigates the problem of identifying suitable feature set among all the features extracted from the deep web search interfaces. Such features remove divergence of domain in the retrieved results. The classification approach employs a filtering FS method of text classification with a Gaussian process classifier. Each search interface is treated as a simple bag-of-words. At first a suitable subset of words is chosen by conducting experiments with various FS techniques, such as X2 (CHI), Information Gain (IG), Bi-normal separation (BNS) to verify that feature selection improves classification performance. Then a new feature selection method is devised with new metrics Top-two-category separation (T2CS) and Top-two-category separation (T2CS-CHI) and a simple ranking scheme.

In their paper, P Zhao et al. [19] express deep web in the form of graphs with heterogeneous multiple relationships. The nodes in the graph denote query interface form, the edges denote the relation between the relevant query interfaces, and the relative weight denotes the similarity between them. Thus the whole form-set is represented in the form of weighted undirected graph. The weight of the edge in the graph is measured by matching degree between schemas of two attribute sets. For schema matching instead of using binary value logic, fuzzy set theory is used. Finally the extracted feature set is clustered using fuzzy clustering method.

4.5 Performance Evaluation Metrics

To evaluate the performance of the deep web classification approaches discussed above five kinds of evaluation metrics have been used: Accuracy, Precision, Recall, F-measure, and Entropy.

Accuracy is the overall correctness of the model and is calculated as the sum of correct classifications divided by the total number of classifications.

Precision is the proportion of returned documents that are targets while recall is the proportion of target documents returned. In other words precision measures the exactness of a classifier while recall measures the completeness or sensitivity of a classifier. Mathematical formulation of precision and recall comes from a confusion matrix shown in table 1.

Table 1. Confusion Matrix to find Precision and Recall

	Predictive Positive	**Predictive Negative**
Actual Positive	**True Positive (TP)**	**False Negative(FN)**
Actual Negative	**False Positive(FP)**	**True Negative (TN)**

$$\text{Precision} = TP / (TP + FP)$$

$$\text{Recall} = TP / (TP + FN)$$

Here TP shows the number of deep web sources classified correctly. FP shows the number of deep web sources which belong to some other category but falsely classified in a domain. FN shows the number of deep web sources falsely excluded from the domain.

A higher precision means less false positives, while a lower precision means more false positives. This is often at odds with recall, as an easy way to improve precision is to decrease recall. Similarly higher recall means less false negatives, while lower recall means more false negatives. Improving recall can often decrease precision because it is very hard to be precise as the data set increases. We can understand the complex difference by considering the scenario. A precision value of 1.0 for a class X means that every item determined as belonging to class X does indeed belong to class X, but gives no information about the number of items from class X but were not determined correctly. Similarly a recall valued 1.0 means that every item from class X was determined as belonging to class X but gives no information about how many other items were incorrectly also determined as belonging to class X.

To overcome the tradeoff among precision and recall they are not used in isolation to determine the performance of the classification process. Instead, either scores for one measure are compared for a fixed level at the other measure e.g. precision at a recall level of 0.8 or both are combined into a single measure, such as the F-measure, which is the weighted harmonic mean of precision and recall.

$$\text{F-measure} = (2*\text{Precision}*\text{Recall}) / (\text{Precision} + \text{Recall})$$

A high F-measure means that both recall and precision have high values. A perfect classification would result in an F-measure with a value equal to 1.

Entropy is used to measure the results of clustering. For a given number of clusters N, the score of conditional entropy lies in the range of 0—logN. In which '0' denote that the clustering process performed is 100% correct while logN denotes that the clustering results are purely random. Due to which the items are evenly distributed into all clusters.

5. CONCLUSIONS AND FUTURE WORK

Online databases lie deep in the ocean of WWW and their structure restricts crawlers from indexing their contents. These databases contain highly relevant content to satisfy user's information needs. In order to generate knowledge for making accurate and timely decisions we need to integrate data from these heterogeneous deep web sources. In this paper a detailed survey of automatic deep web classification techniques is presented, which is key to the realization of the data integration from heterogeneous sources. A general framework for automatic deep web classification is devised. On the basis of the proposed framework all the classification techniques are thoroughly examined. Our study concludes by formulating a brief summary of the key parameters analyzed in the surveyed approaches shown in table 2 of the appendix. Currently all the solutions for automatic deep web classification are training based. Our future work relates to proposing a training less ontology based solution for the deep web classification.

6. REFERENCES

[1] K. C.-C. Chang, B. He, C. Li, M. Patel, and Z. Zhang. Structured databases on the web: Observations and implications. SIGMOD Record, 33(3):61–70, Sept. 2004.

[2] B. He, T. Tao, and K. C.-C. Chang. "Organizing structured web sources by query schemas: a clustering approach," Proc. Of Conference on Information and Knowledge Management (CIKM 04), ACM Press, 2004, pp.22--31.

[3] Deep web search directory service: http://www.completeplanet.com.

[4] Deep web search directory service: http://www.invisibleweb.com.

[5] Wikipedia: http://en.wikipedia.org/wiki/Deep_Web

[6] BrightPlanet.com. The deep web: Surfacing hidden value. Accessible at http://brightplanet.com, July 2000.

[7] Barbosa, L., Freire, J., Silva, A. "Organizing hidden-web databases by clustering visible web documents," Proc. of IEEE 23rd International Conference on on Data Engineering (ICDE 07), IEEE Press, 2007, pp.326--335.

[8] L. Gravano, P. G. Ipeirotis, and M. Sahami. QProber: A system for automatic classification of hidden-Web databases. ACM TOIS, 21(1):1–41, 2003.

[9] Panagiotis G. Ipeirotis , Luis Gravano , Mehran Sahami, Automatic Classification of Text Databases Through Query Probing, Selected papers from the Third International Workshop WebDB 2000 on The World Wide Web and Databases, p.245-255, May 18-19, 2000

[10] B. He, M. Patel, Z. Zhang, and K. C.-C. Chang. Accessing the Deep Web: A survey. Communications of the ACM, 50(5):95–101, 2007.

[11] H. Xu, X. Hau, S. Wang, Y. Hu: A method of Deep Web Classification. Proceedings of sixth international Conference on Machine Learning and Cybernetics, Hong Kong, 19-22 August 2007.

[12] X. Xian, P. Zhao, W. Fang, J. Xin, Z. Cui: Automatic Classification of Deep Web Databases with Simple Query Interfaces. International Conference on Industrial Machatronics and Automation (ICIMA). 2009.

[13] W. Su, J. Wang, F. Lochovsky: Automatic Hierarchical Classification of Structured Deep Web Databases. WISE 2006, LNCS 4255, pp 210-221.

[14] Tiezheng Nie, Derong Shen, Ge Yu, Yue Kou: Subject-Oriented Classification Based on Scale Probing in the Deep Web. WAIM 2008: 224-229

[15] B. He and K. C. -C. Chang. Statistical schema matching across web query interfaces. SIGMOD Conference, 2003.

[16] A helpful guide to search engines: http://www.searchengineguide.com/

[17] Peiguang Lin, Yibing Du, Xiaohua Tan, Chao Lv: "Research on Automatic Classification for Deep Web Query Interfaces", Intl. Symp. on Information Processing (ISIP), Moscow, pp. 313-317, May 2008.

[18] Hieu Quang Le, Stefan Conrad: Classifying Structured Web Sources Using Support Vector Machine and Aggressive Feature Selection. Lecture Notes in Business Information Processing, 2010, Volume 45, IV, 270-282.

[19] Pengpeng Zhao, Li Huang, Wei Fang and Zhiming Cui: Organizing Structured Deep Web by Clustering Query Interfaces Link Graph. Lecture Notes in Computer Science, 2008, Volume 5139/2008, 683-690.

Appendix: Comparative Analysis of Deep Web Classification Techniques

Evaluation metrics:

A % = Accuracy

P % = Precision

R % = Recall

F % = F-measure

E % = Entropy

Table 2. A brief summary of key deep web classification parameters

Technique	Structured	Un-structured	Simple search interface	Advance search interface	Classificat-ion	Clustering	Query Probing	Visible form features	Macro	Micro	Use of Ontology	Evaluation Metric				
												A%	P%	R%	F%	E%
B. He et al. [2]	✓			✓		✓		✓	✓							0.32
Barbosa et al. [7]	✓	✓	✓	✓		✓		✓	✓						86	0.46
Sahami et al. [9]		✓	✓		✓		✓		✓			90				
H. Xu et al. [11]	✓			✓	✓			✓	✓		✓		92	92	92	
X. Xian et al. [12]	✓		✓		✓		✓		✓				88	83	86	
W. Su et al. [13]	✓			✓	✓		✓		✓						78	
T. Nie et al. [14]	✓			✓	✓		✓	✓	✓	✓		85				
P. Lin et al. [17]	✓			✓		✓		✓	✓				95	94		
H. Le et al. [18]	✓			✓	✓			✓	✓				96	95	95	
P. Zhao et al [19]	✓			✓		✓		✓	✓						87	0.28