HindiWeb Page Collection tagged with Tourism Health and Miscellaneous

by

Nikhil Priyatam, V. Srikanth Reddy, Vasudeva Varma

in

The eighth international conference on Language Resources and Evaluation (LREC)

ICEC Istanbul Lütfi K■rdar Convention Exhibition Centre Harbiye 34267 Istanbul, Turkey

Report No: IIIT/TR/2012/-1

Centre for Search and Information Extraction Lab
International Institute of Information Technology
Hyderabad - 500 032, INDIA
May 2012

Hindi Web Page Collection tagged with Tourism Health and Miscellaneous

Pattisapu Nikhil Priyatam, Srikanth Reddy Vaddepally, Vasudeva Varma

International Institute of Information Technology
Hyderabad, Andhra Pradesh, India
nikhil.priyatam@research.iiit.ac.in , srikanthreddy.v@research.iiit.ac.in , vv@iiit.ac.in

Abstract

Web page classification has wide number of applications in the area of Information Retrieval. It is a crucial part in building domain specific search engines. Be it 'Google Scholar' to search for scholarly articles or 'Google news' to search for news articles, searching within a specific domain is a common practice. **Sandhan** is one such project which offers domain specific search for *Tourism* and *Health* domains across 10 different Indian Languages. Much of the accuracy of a web page classification algorithm depends on the data it gets trained on. The motivation behind this paper is to provide a proper set of guidelines to collect and store this data in an efficient and an error free way. The major contribution of this paper would be a *Hindi* web page collection manually classified into *Tourism,Health* and *Miscellaneous*.

1. Introduction

Web search in Indian languages is constantly gaining importance.With the fast growth of Indian language content on the web, many classic IR problems (Web page classification, focused crawling, Ranking etc) need to be addressed. Though most of these problems seem to be solved for high resource languages like English, the solutions cannot be applied to Indian languages because of dearth of resources. Indian languages have rich morphological features which can be used to solve the problem in a better way.Hence preparation of Indian language resources for IR tasks is very important.

Problems like WPC require huge amount of training data to ensure diversity and coverage. This requires huge amount of manual labor. Moreover for these algorithms to work accurately, the training data should be error free. A good training dataset should be an actual representation of the real data. In an ideal scenario the training and testing documents should come from the same distribution. But generally this does not happen since the distribution is unknown to us. One of the features offered by **Sandhan** is *domian* specific search. In this context we define *domain* as a category of web pages which satisfy a particular user need. Currently our search engine is expected to support only two domains namely *Tourism* and *Health*. The domain identification module is supposed to classify a web page from the crawl as *Tourism* or *Health* or *Misc* (neither *Tourism* nor *Health*).
The purpose of this work is to collect good quality training data for Domain Identification module.
Documents belonging to the tourism domain are supposed to contain the following information:

- Information about historical places
- Tourist attractions in India
- Travel Guide which includes cost of trip,best time to visit, transport facilities, nearby hotels, accommodation facilities, and nearby places of interest of a particular tourist spot.
- Food and other popular cuisines.
- On line services (if any) provided by the respective tourist spots.
- Latest news about a particular place which might include weather reports and any other news alerts.

Documents belonging to the health domain are supposed to contain the following information:

- Complete information about any disease: Symptoms, precautions, cure and other related information.
- Information about medicines, their ingredients, possible side effects, availability and the type of medicine (*ayurvedic,allopathy,homeopathy etc*) .
- Information about clinics or hospitals, doctors etc.
- Latest research news about some ailments.
- Information related to nutrition, diet, personal hygiene etc.

We used the proposed guidelines to collect, classify and store *Hindi* web pages. These web pages contain information about Indian tourist attractions, General health topics and *Miscellaneous* (neither *Tourism* nor *Health*).

2. Background

Several areas in Information Retrieval like Web page classification (WPC) and focused crawlers (FC) require Machine Learning algorithms for training their classifiers. Significant amount of research has gone into exploring different kinds of algorithms and right feature combinations which would work, but not enough work has gone into what qualifies to be a good training data and how to collect it. No algorithm can learn anything significant on junk.

2.1. Web Page classification

Web Page classification is the task of categorizing web pages into different classes. This is an extended problem of document classification. Document classification is a fundamental learning problem that is at the heart of many information management and retrieval tasks (Power et al., 2010).

A document classification works on plain text as compared to the rich set of features that can be explored in web pages. Most of the approaches use machine learning algorithms to solve this problem. Irrespective of which algorithm one uses, accuracy of a classification algorithm depends on the feature sets. Web page classification uses wide number of features from the simplest like URLs (Baykan et al., 2009) to the complex ones like topic models (Sriurai et al., 2010). In addition to these features there are also many common features that these algorithms use. Some of them are:

- In links
- Out links
- Page size
- Content
- Title

and any valid combination of these features. Mainly 3 types of algorithms are used to solve this problem.

2.1.1. Supervised Algorithms

Supervised algorithms are one specific class of learning algorithms which infer a function from supervised (labeled) training data (Duda et al., 2001). These are often called as classifiers. Once the classifier learns on the labeled data, it predicts the label of the unlabeled data. The most common supervised algorithm is the Naive Bayes algorithm (Duda et al., 2001).

2.1.2. Semi-supervised Algorithms

Semi-supervised learning is a learning paradigm concerned with the study of how computers and natural systems such as humans learn in the presence of both labeled and unlabeled data (Zhu and Goldberg, 2009). There also exist methods which use large unlabeled samples to boost performance of a learning algorithm when only a small set of labeled examples is available (Blum and Mitchell, 1998).

2.1.3. Unsupervised Algorithms

Unsupervised learning algorithms do not require any training data. These algorithms work by calculating the similarity between different samples and clustering them under one group. Multilingual document clustering is one such area. (Steinbach et al., 2000) compares different document clustering algorithms.

Be it a single algorithm or a mix of different algorithms, the labeled web page data is crucial to solve this problem. Even for unsupervised methods labeled data will always help in determining quality of a cluster (Steinbach et al., 2000).

2.2. Focused Crawlers

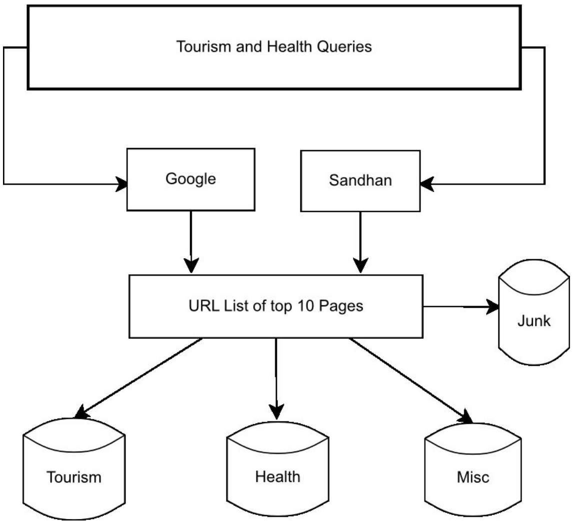
Focused crawlers are the ones which selectively seek out pages that are relevant to a pre-defined set of topics. Rather than collecting and indexing all accessible Web documents to be able to answer all possible ad-hoc queries, a focused crawler analyzes its crawl boundary to find the links that are likely to be most relevant for the crawl, and avoids irrelevant regions of the Web (Chakrabarti et al.,). To achieve this task focused crawlers need some training either on a predefined ontology as in (Chakrabarti et al.,) or tagged URLs.

3. Data Collection

For web page classification we require real web pages to be classified into respective domains. In our dataset we identify each web page with its URL classified into *Tourism*, *Health* and *Misc*. Since we cannot pick random pages from the web and tag them, a systematic approach needs to be followed. The Following section describes our approach.

3.1. Approach

Our data collection process can be logically broken down into 3 steps: Query collection, Firing them on a Search Engine and Classifying retrieved results. The data collection process can be represented in the form of a flowchart.



3.1.1. Query Collection

This is a very crucial part of our system. If queries are not proper and diverse enough we cannot expect our training set of URLs to be an actual representation of entire data. For query collection we have 30 people from different places in India speaking 10 different languages. Everyone of them is explained what qualifies to be a *Tourism* query and *Health* query and then each person is asked to prepare 3 set of queries:

- Regional Queries: These are tourism queries about specific places or locations where the person belongs.
- Non regional Queries: These are also tourism queries but are quite generic in nature (not related to the place where the person belongs).
- Health Queries: Any health related query.

Along with the queries the person is asked to provide English translation of the queries and the intent of the query.

First let us consider only *Tourism* queries, By collecting queries in this way we are ensure that the coverage is diverse enough. Note that one person's regional query can be others' non regional query, so we have both local as well as global coverage of queries. Even for *Health* queries, by collecting queries from different people we ensure sufficient diversity in the health queries too. Once we have the queries we translate all of them into ***Hindi*** for our experimental purpose. The translation is done manually so that no error creeps in this part. After translation we eliminate all duplicates (if any). We also make sure that all the queries are collected in isolation i.e. no one knows queries given by other people. The following table gives the split up of the queries.

Type of queries	No of queries
Regional	60
Non-Regional	60
Health	60

3.1.2. Firing Queries on Search Engines

Once we have the above set of queries we use a search engine to get the relevant web pages. Search engines like Google usually provide diverse results i.e. results from different hosts. For our work we actually fire the queries on Google as well as **Sandhan** to ensure wider coverage and pages for diverse websites.

3.1.3. Classifying Retrieved Results

The results retrieved for each query are manually looked at and classified into one of the 3 defined domains as per the set guidelines. The top 10 results for each of Google and **Sandhan** are considered for manual classification. More than 10 results were also considered for some queries whenever the top 10 results had many miscellaneous pages. The pages going into *Misc* category are the ones returned by the search engine in response to a *Tourism* or a *Health* query but belong to neither of the domains. Therefore we are not making any specific efforts to fetch Miscellaneous pages, because our main interest is to get *Tourism* and *Health* pages only. For our current work we do not do a further analysis on sub-domains of Tourism and Health. It has to be noted that a page is classified as *Tourism* or *Health* if it strictly falls within our definition. It need not be completely relevant to the fired query. Pages which have no or meagre text are discarded. The following table shows the statistics of the data collected.

Domain	No of Web Pages
Tourism	885
Health	978
Misc	1354

4. Data Storage

Storage of Indian language datasets needs to overcome various challenges. This problem is much more pronounced when it comes to storing web page collections. This is because Indian language content authors use very proprietary (Non standard) character based encoding formats alongside with the standard Unicode format. For proper distribution and machine readability Unicode is most preferable. We store the URLs (not the content) and its corresponding tag. This is to ensure that when we have the data ready to be used for Web Page Classification, we can crawl the collected set of URLs and extract any features that we wish (which may include features other than raw text like images,titles etc). Since all of these pages are not in the standard Unicode format, we provide Unicode converted pages along with the dataset.

We have chosen *SQLite3* [1] as the storage media. *SQLite3* is a portable, self-contained database. According to their website, it is the "most widely deployed database engine in the world". Also there are no licensing issues when using *SQLite3*, since it's in the public domain [2]. *SQLite3* has the following advantages over traditional storage mechanisms like flat files, Xml etc.

- *SQLite3* is a zero configuration fully featured RDBMS system. It fully supports the SQL 92 standard.
- This allows easy alteration like updates, inserts, deletes etc.
- All data is stored in a single file which makes it distributable. *SQLite3* also allows encryption of the database file if the data has to be distributed in a secure manner.
- The open storage standard of *SQLite3* allows it to be used independent of any operating system.
- *SQLite3* allows storage of Unicode data. It supports UTF-8, UTF-16BE and UTF-16LE [3].
- APIs are available in various programming languages which makes it easy to consume the data or convert it into any required format.

4.1. Table Structure

Our dataset is stored in a single *SQLite3* database with 2 tables namely URL_Tags and URL_Content. Table 1 shows the structure of URL_Tags. The URL_Content table provides the content of all the URLs after converting to Unicode. Table 2 shows the structure of this table.

One observation about our data is that even though our queries were purely *Tourism Queries* or *Health queries*, the number of general or miscellaneous documents outnumber the tourism and health documents. This behavior is consistent in both Google as well as **Sandhan**. This shows that most of the pages do contain tourism or health related keywords but do not belong to *Tourism* or *Health* domain, in specific. This might also be a consequence of our strict categorization policy. Miscellaneous pages are as important as tourism and health pages because in the task of web page classification they play the role of negative samples.

[1] http://www.sqlite.org/different.html

[2] http://www.switchonthecode.com/tutorials/php-tutorial-creating-and-modifying-sqlite-databases

[3] http://www.sqlite.org/datatype3.html

Table 1: URL Tag table structure

Column Name	Data type	Notes
Doc_id	INT	A numeric id for each web page
URL	TEXT	The URL of the web page.
Lang	TEXT	A 2 character ISO language code
Domain	TEXT	Domain tag of the web page

Table 2: URL Content Table structure

Column Name	Data Type	Notes
Doc_id	INT	A numeric id of a web page as given in table 1 (foreign key constraint)
Unicode Content	TEXT	UTF-8 HTML content obtained by converting proprietary encodings if any. The Text datatype can comfortably store large web pages.

4.2. SQLite3 API

SQLite3 is an open source database with API support in many languages. PHP 5 has in built support for *SQLite3*. This is the version that we have used for building our data collection tool. The following PHP code fragment shows how to issue a SELECT query to an *SQLite3* database.

```
<?php
$db_file = 'database_file_path';
$table_name = 'Some_table_Name';
$db = new SQLite3($db_file);
$query='SELECT * FROM $table_name';
$results = $db->query($query);
while($row = $results->fetchArray
(SQLITE3_ASSOC))
{
  \\ Process a single row.
}
$db_close();
?>
```

The following are some sample SQL statements that can be used to retrieve data from our dataset.

- Aggregate information: Selects all the information of a particular document by combining both the tables.

  *SELECT * FROM url_List,url_Content*

- Selecting Hindi tourism URLs.

  SELECT url FROM url_List WHERE lang='hi' AND domain='tourism'

Since *SQLite3* fully supports the SQL92 standard, any valid SQL statement can be used to retrieve data from the database.

5. Conclusion and Future Work

We conclude that this mechanism of data collection is simple and efficient. It minimizes manual effort and reduces errors. We also discuss an efficient storage mechanism which is easy to use and distributed. We also believe that the tagged *Hindi* web page collection prepared by us will find great use in further research on Web page Classification and Focused crawling. We plan to collect data in a similar fashion for all other Indian languages and evaluate it. We also plan to further minimize manual effort by using these URLs as input to a semi supervised focused crawling algorithm. We plan to provide data quality evaluation statistics by using measures like cluster quality, classification accuracy etc for all our datasets. In future, we wish to extend this approach to do a much more fine grained analysis on the sub domains of Tourism, Health on the same data.

6. References

Eda Baykan, Monika Henzinger, Ludmila Marian, and Ingmar Weber. 2009. Purely url based topic classification. In *World wide web poster*, April.

Avrim Blum and Tom Mitchell. 1998. Combining labeled and unlabeled data with co training. In *COLT' 98 Proceedings of the eleventh annual conference on Computational learning theory*.

Soumen Chakrabarti, Martin van den Berg, and Byron Dom. Focused crawling: a new approach to topic-specific web resource discovery. *Computer Networks*, 31.

Richard O. Duda, Peter E. Hart, and David G. Stork, editors. 2001. *Pattern Classification*. John Wiley and Sons. 2nd edition.

Russell Power, Jay Chen, Trishank Karthik, and Lakshminarayanan Subramanian. 2010. Document classification for focused topics. In *Association for the Advancement of Artificial Intelligence*.

Wongkot Sriurai, Phayung Meesad, and Choochart Haruechaiyasak. 2010. Hierarchical web page classification based on a topic model and neighboring pages integration. In *(IJCSIS) International Journal of Computer Science and Information Security*.

Michael Steinbach, George Karypis, and Vipin Kumar. 2000. A comparison of document clustering techniques. Technical Report 00-034.

Xiaojin Zhu and Andrew B. Goldberg. 2009. Introduction to semi-supervised learning. June.