Top K List Extraction from Web Pages

Priyanka Deshmane
M.E. student
DYPIET Pimpri, Pune
Maharashtra, india

Pramod Patil, PhD
HOD Dept. of computer
DYPIET Pimpri, Pune
Maharashtra, india

Abha Pathak
Professor Dept. of computer
DYPIET Pimpri, Pune
Maharashtra, india

ABSTRACT
In present days finding relevant and desired information in less time is very crucial however problem is that very small proportion data on internet is interpretable and meaningful and need lot of time to extract. The paper provides solution to problem by extracting information from top-k websites, which consist top k instances of a subject. For Examples"top 5 football teams in the world". In comparison with other structured information like web tables top-k lists contains high quality information . It can be use to enhance open-domain knowledge base [which can support search or fact answering applications]. Proposed system in paper extract the top k list by using title classifier, parser ,candidate picker , ranker, content processor .

General Terms
Web mining

Keywords
Data extraction, Structured information, top k list , top k web pages, web parser

1. INTRODUCTION
WWW is very important source for getting information and huge amount of information is available. Top k pages are rich source of valuable information available on internet and these pages exist in small percentage. It is important to extract top k list from such pages for getting correct information but it is difficult to extract knowledge from information explained in natural language and unstructured format. Some information over internet is present in organized or semi-organized form for example, as records coded with specific names e.g. html5 pages. As per a large measure of new technique has to be dedicated for understanding structured information on the web, (like web tables) especially from internet platforms .[1]

Quantity of web tables is large but slight proportion of them include helpful information. A very small amount of table include data interpretable without context. Many tables are not relational, since it is easy to interpret relational table with rows referring to entities and columns referring to characteristics of these entities. Based on Cafarella et al. [13],total of the 1.2 % of web tables which are relational, the most are worthless without context.

Consider a table which has 4 rows and 3 columns, where the three columns are marked as "bikes" , "model" and "prize" respectively. we don't understand why these 4 bikes are gathered together (e.g., are these most expensive, or fastest). We don't know the definite situations for which information is useful. The context is very important for extracting information, but in many of the cases, context is represented in such a manner that the machine could not understand it. In this paper instead of focusing on structured data (like tables, xml data) and ignoring context, concentration is on easily understanding context and use context to interpret less structured or free-text information and guide its extraction.

The title of top k page should consist minimum 3 section of information : i)k e.g. 12 , ten. Means number of items does page contain. ii)A topic or idea items is associated with. e.g. artists, players. iii) Ranking criterion e.g. fastest ,tallest, best seller. Sometime title contain two optional elements time and location [1]

2. LITERATURE SURVEY

2.1 Automatic Extraction Of Top K List From Web
Zhixian zhang , kenny Q.zhu,haixun wang , hong song li[1]

Author proposed a method for extracting information from top k web pages, which contains top k instances of a interested topic . This method gives improved performance by providing domain specific lists and focusing more on the content. It doesn't focus only on the visual area of the lists.

If list is divided into more than one pages it may not get included completely. Author demonstrated algorithm that automatically extracts such top k lists from the web snapshot and structure of each list was discovered . Algorithm achieves 92.0% precision and 72.3% recall in evaluation.[1] [16]

2.2 System for Extracting Top K List From Web
Z.zhang,K. Q. Zhu.H.wang[2] Author defined list extraction problem which concentrated on finding and extracting 'top-k' lists from web pages. The problem was different from other as top k lists a were easy to understand and contain high quality information. Probase can be enhanced with the help of knowledge stored in lists and use for developing a efficient search engine. 4 stage framework has demonstrated by author which has ability to extract top k list at very high precision.[2] [16]

2.3 Extracting general from web document
F. Fumarola,T. Weninger,R.Barber,D.Maleba and J.Han [6] Author proposed a new hybrid technique for extraction of general lists from the web . Method uses basic assumption on visual rendering of list and structural arrangement of items . The aim of system was to overcome the limitations of existing work which deals with the generality of extracted lists.Several visual and structural characteristics were combined for achieving goal.To find and extract the general list on web both information on visual list item structure and non visual information such as DOM tree structure of visually aligned items were used. [16]

2.4 Short text conceptualization using probabilistic knowledge base
Y.song,H.Wang,Z.Wang,H.Li and W.Chen[7] Author proposed a technique to improve text understanding by

making use of a probabilistic knowledge.Conceptualization of short words is done by Bayesian interference mechanism. Comprehensive experiments were performed on conceptualizing textual terms and clustering short segments of text such as Twitter messages. Compared with purely statistical technique like latent semantic topic modelling or methods that use existing knowledge base (e.g. WordNet, Freebase and Wikipedia) approach brings notable improvements in short text conceptualization as shown by the clustering accuracy.[7] [16]

2.5 Extracting data records from web using tag path clustering

G.Miao , J.Tatemura,W.P.Hsiung,A.Sawires,L.E.Moser[10]

Author proposed a technique for extraction of record that recognize the list in powerful fashion on basis of detail analysis of web page.The focus is on frequent appearance of distinct tag path in DOM tree. It correlate tag path pattern pair (visual signal) to calculate similarity between two tag path.Data record clustering of tag path is done on basis of similarity measure . Results were compared with state of art algorithm . Algorithm shows high accuracy in extracting atomic-level as well as nested-level data records. The algorithm has linear execution time in the document length for practical data sets.[10] [16]

2.6 Towards domain independent information extraction from web tables

W .Gattterbaur, P. Bohunsk , Herzog, B.krupal B.Pollak[14] Author mentioned the difficult task of extraction of domain independent information from web tables by moving focus from representation in tree format of web page to variety of visual box model which are multi-dimensional and used by web browsers to show the information on screen. The gap formed by missing domain specific knowledge about content and table templates can be fill by topological information obtained.[14][16]

3. PROBLEM DEFINITION

To extract a top k lists from structured as well as unstructured information on web by using efficient web mining algorithms

4. PROPOSED SYSTEM

A Proposed system consists of 4 components .Title classifier, Candidate Picker, Ranker , content processor.

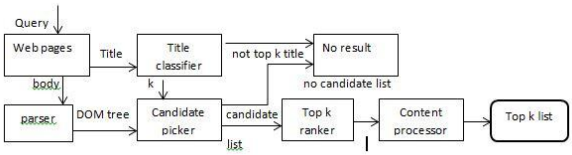


Fig 1: System Architecture

4.1 Title classifier

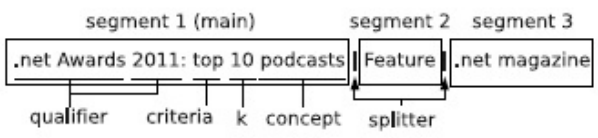


Fig 2: Example of top k title

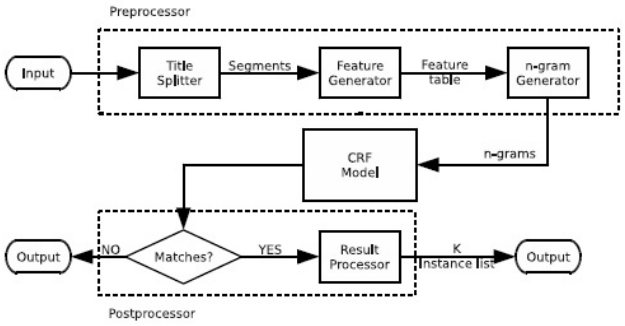


Fig 3: Working of title classifier

Online page title help us to identify top k page.First reason to use title is, for many situation page title gives introduction about the subject . Second the page body could have different and complex formats but top k page titles have similar structure. Analysis of title is light weight and economical. If the analysis result shows that a page isn't a top k page then such pages are skipped . Example of top k title is shown in fig 2. Title may contain additional segments like time and location which are optional in addition to k ,concept and ranking criterion. Segments may be separated with "-" or "— ". Main segment contains the topic and other segments contain additional information.Title is split and the part which contain number is obtained. number k is important for representing topic concept. Feature extraction of title is done in fixed size window which is centred around number k.

4.1.1 CRF MODEL

X is defined as a word sequence and lable sequence is define as Y Yi{TRUE,FALSE}

Conditional probability for linear chain CRF is calculated as $P(Y|X) = \frac{1}{Z(x)} \exp(\sum_{i=1}^{n} \sum_{j=1}^{m} \lambda_j f_j (y_{i-1}, y_i, x, i))$

Normalization factor Z(x) one of the m function is fj feature weight to be trained is λj

4.1.2 Creating training data set

"top CD" : top word with number for e.g. top 10 singers

"top CD" without word „top"

"*CD JJS": "JJS"* means superlative adjective for e.g. tallest building

"CD RBS JJ": "RBS" and "JJ" stands for superlative adverbs and adjective resp. .For e.g. most expensive

4.1.3 Feature extraction

Feature extraction of title is done in fixed size window which is centred around number k. four features are selected *Word, Lemma , POS tag ,Concept*

4.1.4 Working of title classifier :

Fig 3 shows how title classifier works. 1) Feature generated by pre-processor. 2) n gram pattern are labelled as TRUE or FALSE by classifier 3)if value is true ,then post processor extracts k, concepts, ranking criterion from title.

4.2 Candidate picker

The structures that looks like top k lists are extracted at this stage.a top k candidate must be initial and should have listing of k things. Visually it must recognize as k horizontal and k vertical aligned in regular pattern.structurally it is list of nodes with equivalent tag path. Tag path is a path from root node to definite node. It can be given as list in sequence of tag names. following basic rules are applied for extracting candidate list

- K items: exact k item must be present in candidate list
- Identical Tag path : each node in candidate list must have same tag path

Algorithm 1 shows tag path clustering algorithm.it process the input according to two basic rules .At line 2 tag path of each node is calculated and then text nodes having the similar tag paths are grouped into one node list.After completion the set of list of nodes is obtained which are precisely k nodes. Although above method computes a top k list with high recall, it may produce false positive results.So for further filtering three pattern based rules are applied .

1. index: index is an integer number present in front of list item.e.g. 1,2,3. numbers should be in sequence and within range of [1,k]
2. Highlighting tag: for highlighting purpose candidate list tag path should contain at least one tag among < b >< strong >< h1 □ h6 >
3. Table: candidate list present in table format List satisfying at least one rule gets to stay in candidate set.

Algorithm 1 Tag Path Clustering Method

1: procedure TAGPATHCLUSTERING (n; table)

2: n:T agPath ← n:P arent:T agPath +

Splitter + n:T agName;

3: if n is a text node then

4: if table contains the key n:T agPath then

5: list ← table[n:T agPath];

6: else

7: list ← new empty lists;

8: table[n:T agPath] ← list

9: end if

10: Insert n into list;

11: return;

12: end if

13: for each node i ∈ n:Children do

14: TagPathClustering (i; table);

15: end for

16: return;

17: end procedure

4.3 Top k ranker

Top k ranker select the top ranked list by ranking the candidate set and using scoring function which is weighted sum of two feature scores.

4.3.1 P -Score:

P-score is used to measure the interaction of the title and list. one of the fundamental concept of the top-k list from a set of concepts is extracted from title.Main idea is that one or more items in the list should be instance of fundamental concept from title. For example if concept is ”museum” then items in the main list should be instances of ”museum” concept . The probase taxonaomy gives the large number of concept and instances.

Each candidate p score in list L is calculated by:

$$pscore = \frac{1}{k}\sum_{n\in L}\frac{LM1(n)}{Len(n)}$$

Where LM1(n) is word count of longest matched instance in the text node n while len(n) means the word count of entire text in node n. LM1(n) is divided by Len(n) to normalize the p score to [0,1] and the contribution of each node will be no more than 1/k, which make sure that one single node's effect doesn't dominate the whole score. In addition p score prefers lists with fewer words because node with many words(e.g. a description paragraph) are less likely to be a part of top k list.

4.3.2 V -Score:

visual area inhabited by list is calculated V score. The main and important list of the page is preferably larger and outstanding than minor lists. The sum of visual area of each node is v score and v score is calculated by:

$$Area(L) = \sum_{n\in L}(TextLength(n) \times Fontsize(n)).^2$$

4.4 Content processor

Attribute/value pairs for each item are extracted from its description. Goal is to get structured information for each item in list.By analysing the title valuable information can be obtained. Three important steps in the content processor.

4.4.1 Infer the structure of text node

In most of the situations the text node describing each item may contain inner structure or semi structure.(E.g XXX(YYY)). inner structure of text is inferred by computing frequency distribution for each possible separated tokens ”by”,”:”,”:” from all items of top k list. Sharp spike in the distribution for a specific token means the number of occurrences for that token is similar among all the list nodes, then a separator is found which can be used to separate the text into multiple fields. Each field is often an attribute or property of the entity described by the list item.

4.4.2 Conceptualize the list attribute:

It is beneficial to infer the schema for the attribute after distributing the list item into attribute value. In system three methods are utilized to conceptualize list attributes:

Table head : if the list is shown in table format then it can be use to conceptualize the table directly. the table head is included in the< th > tags.

Attribute/value pair: The list might contain explicit attribute/value pairs. For e.g. in fig.1. ” Hosted by” is an attribute of the list item ”The Big Web Show” and its value is ”Jeffrey Zeldman and Dan Benjamin”. In general if every element of column contains the same text and ends with colon. Column is considered as the attribute column and

the column to the right as the value column . After that the attribute name is used to conceptualize the corresponding values .

Column Content: If both table head and attribute/value pairs are not present ,the basic technique is to conceptualize the extracted column content by technique proposed by song et al[10] using probase and Bayesian model.For each text column the longest known probase instance in the text is use to represent.

4.4.3 Detect when and where:

time and location information is important semantic information for extracted top k lists. extracting this information is investigated from the page title. A named-entity recognition (NER) problem can be solved by applying state-of-art NER tools.

The experimental results indicate that both ”when” and ”where” can be detected with high recall but precision for locations is low as many location entities are not related to the main topic of title.

For example, some locations included in the title of the website, such as ”New York Times”. Thus two additional rules given below are effectively applied for filtering irrelevant location entities without causing too much harm to coverage.

- The main segment: The location entity must be in the main segment of the title.
- Proper preceding word:

The word that precedes the location entity must be a proper preposition such as ”in”, ”at”, ”of” etc. Further for date attribute, temporal relations are discovered such as ”before”,”during and ” after” . This can be done by looking for certain key words before the entity, which is similar to the second rule above. for example, a proper preposition for the relation ”after” can be ”after”,”since”or ”from”.

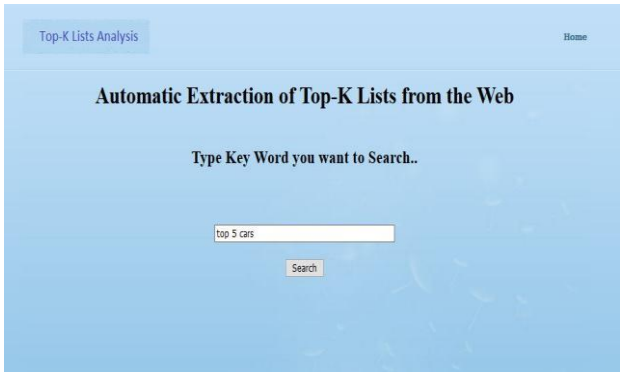
5. COMPARISON OF SYSTEMS

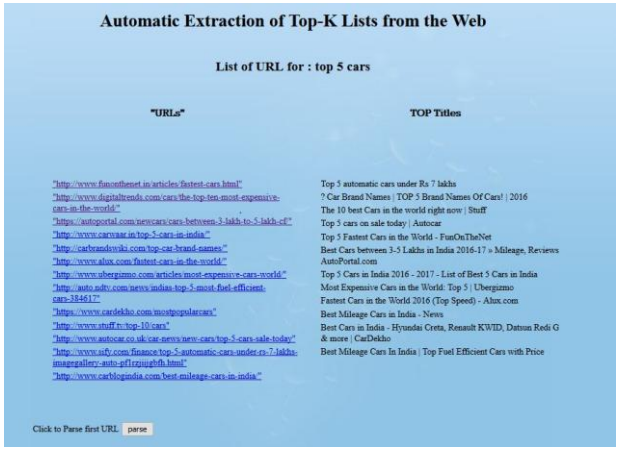
Table 1. Comparison of similar systems

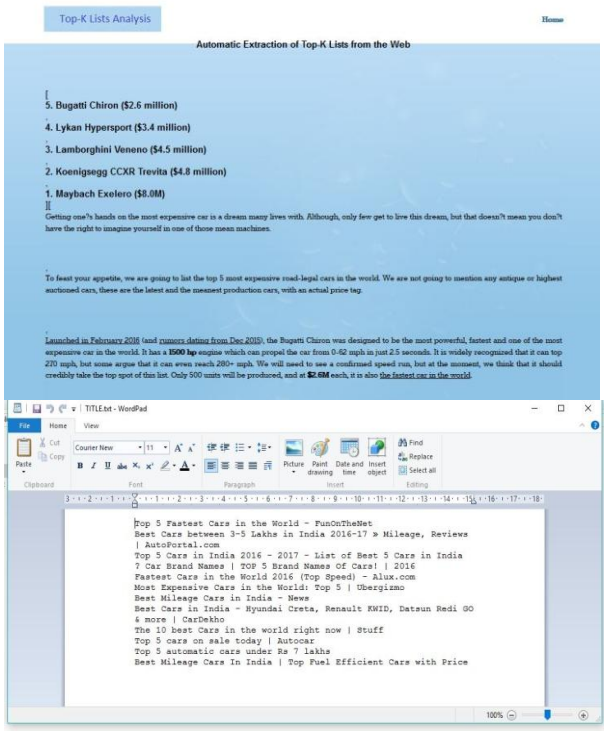
	Extracting General list Hybrid approach	Extracting data records using tag path clustering	Proposed approach
Working/Algorithm	HyLiEn (**Hy**brid approach for automatic **Li**st discovery and **E**xtractio**n** on the Web),	Tag path clustering spectral clustering algorithm	Tag path clustering
Advantage	Employs both general assumptions on the visual rendering of lists, and the structural representation of items	can also detect nested data records. Template tags and decorative tags are distinguished naturally.	Extract top k list with high performance
Limitation	The computation time for HyliEn is 4.2 seconds on average.	extracts data records from single Web pages.	Cannot extract web pages that are interlink
complexity	bounded on the structural complexity	$O(M\times L)+O(M^3)$, computation time 0.3 sec	Computation time is much less

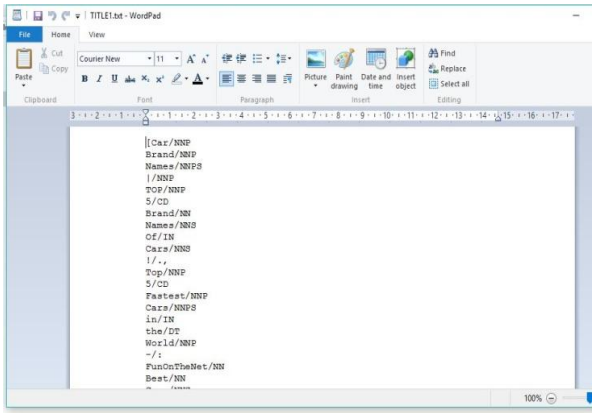
6. IMPLEMENTATION DETAILS AND RESULTS

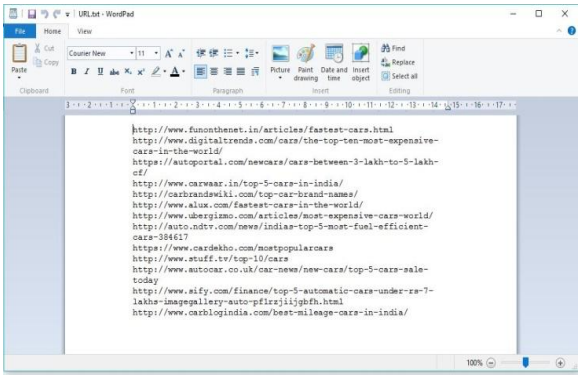
First user type a query to search. After inserting query the url from Google API and classified titles are displayed and titles are stored in text file title 1.Part of speech tagging is done on titles to classify it and store it in title2 file .urls are stored in url text file. After parsing list is shown to user along with details











6.1 Performance measures

System uses Google API for searching which gives best result. Accuracy of system is high as compared with similar system. Performance is measured in terms of precision and recall for how many titles it recognises correctly.

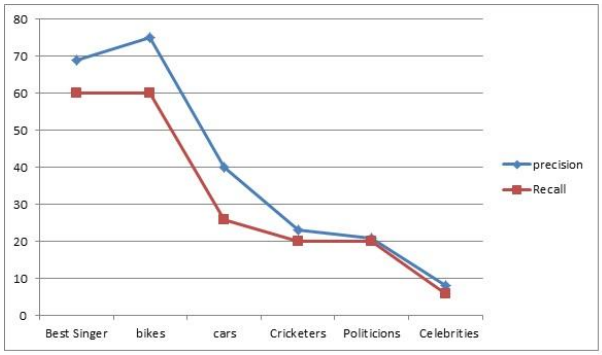


Fig 4: precision and recall of proposed system

7. CONCLUSION

The problem of top k list extraction from web is very important because top k list are easy to understand and contain high quality information. The system is interesting search system in which user enter the top query and get the top k list as output. More work can be done in future as data on internet is increasing and more use of internet gives rise to new demands.

8. ACKNOWLEDGMENTS

I express my gratitude towards my project guide HOD Dr. Pramod Patil and Prof Abha Pathak for their valuable guidance and inspiration.

9. REFERENCES

[1] Zhixian Zhang, Kenny Q. Zhu, Haixun Wang Hong song Li , “Automatic Extraction of Top-k Lists from the Web” IEEE ,ICDE Conference, 2013, 978-1-4673-4910-9.

[2] Z. Zhang, K. Q. Zhu, and H. Wang, “A system for extracting top-k lists from the web” in KDD, 2012.

[3] W. Wu, H. Li, H. Wang, and K. Q. Zhu, ”Probase: A probabilistic taxonomy for text understanding” in SIGMOD, 2012.

[4] X. Cao, G. Cong, B. Cui, C. Jensen, and Q. Yuan, ” Approaches to exploring category information for question retrieval in community question-answer archives,” TOIS, vol. 30, no. 2, p. 7,2012.

[5] J. Wang, H. Wang, Z. Wang, and K. Q. Zhu, ”Understanding tables on the web,” in ER, 2012, pp. 141155.

[6] F. Fumarola, T. Weninger, R. Barber, D. Malerba, and J. Han, ” Extracting general lists from web documents: A hybrid approach,” in IEA/AIE (1), 2011, pp. 285294.

[7] Y. Song, H. Wang, Z. Wang, H. Li, and W. Chen, ”Short text conceptualization using a probabilistic knowledge base,” in IJCAI, 2011.

[8] A. Angel, S. Chaudhuri, G. Das, and N. Koudas, ”Ranking objects based on relationships and fixed associations,” in EDBT, 2009, pp. 910921.

[9] G. Miao, J. Tatemura, W.-P. Hsiung, A. Sawires, and L. E. Moser,” Extracting data records from the web using tag path clustering,” in WWW, 2009, pp. 981990.

[10] EK. Fisher, D. Walker, K. Q. Zhu, and P. White,”From dirt to shovels: Fully automatic tools generation from ad hoc data,” in ACM POPL,2008.

[11] N. Bansal, S. Guha, and N. Koudas, ”Ad-hoc aggregations of ranked lists in the presence of hierarchies,” in SIGMOD, 2008, pp. 6778.

[12] M. J. Cafarella, E. Wu, A. Halevy, Y. Zhang, and D. Z. Wang,”Web tables: Exploring the power of tables on the web,” in VLDB, 2008.

[13] W. Gatterbauer, P. Bohunsky, M. Herzog, B. Krupl, and B. Pollak, ”Towards domain-independent information extraction from web tables,” in WWW. ACM Press, 2007, pp. 7180.

[14] K. Chakrabarti, V. Ganti, J. Han, and D. Xin, ”Ranking objects based on relationships,” in SIGMOD, 2006, pp. 371382.

[15] B. Liu, R. L. Grossman, and Y. Zhai, ”Mining data records in web pages,” in KDD, 2003, pp. 601606.

[16] P Deshmane , P.Patil, Abha Pathak “Survey on web mining techniques for Extraction of top k list”IJMTER 2015