

With On The Fly Document Clustering

Aparna PL (14IT132)

Neha B (14IT224)

Prerana K R (14IT231)

S S Karan (14IT252)



Introduction

• Common Techniques used by search engines:

S1 No.	Technique	Drawback
1	IR (Information Retrieval)	Keyword spamming
2	HVV (Hyperlink Vector Voting)	Does not distinguish between high and low quality pages.
3	Henzinger's PageRank algorithm	High computing power

Purpose

- To present a new web page recommendation system to help users reduce navigational time on the Internet, based on the *primacy effect* of browsing behaviour.
- Collect and rate relevant pages, by collecting search results from different search engines, *metasearch* .

Problem Statement

• To improvise the current result set produced by various search engines by presenting a novel clustering algorithm, called *On-The-Fly Document Clustering* (OTFDC), which produces a number of candidate clusters from other web search results. The candidate clusters so produced, not only generate a connective relation between the clusters, but also the relation is a semantic one.

Literature Review

- NECI metasearch engine-Lawrence and Giles(1998): analyses each document and displays local context around the query terms.
- Glover's search engine(1999): Metasearch engine architecture, which customises the searching and result ranking strategies based on user's information need.
- Svidzinska's *two-tier metasearch engine*(2001): 1-Collect information from topic specific search engines. 2, expand and route user queries to a subset of it.
- *HuddleSearch*-Osdin(2002): New clustering algorithm, which dynamically organises relevant documents into a traversable hierarchy from general to specific.
- Hai(2004): Extracting information from web search interfaces of e-commerce search engines, used to construct e-commerce metasearch engines.

Algorithms and Crux in the Papers

This section describes the proposed search methods in detail. First, the underlying <u>user behaviour function</u> is presented. Then, the formulation and <u>implementation of SVV</u> is described.

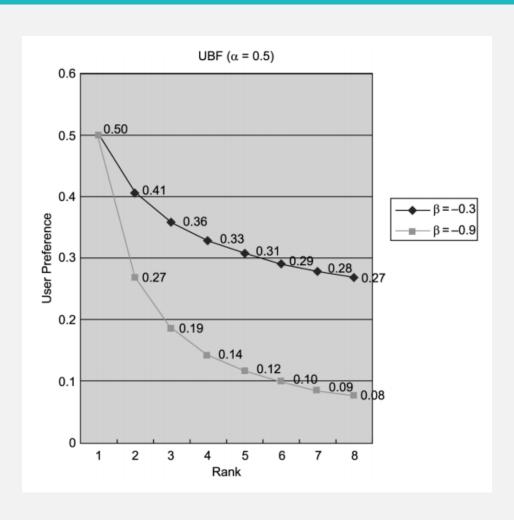
User Behaviour Function

- Primacy effect in psychology
- User behaviour function (UBF) for the i-th item li within an ordered item list l is defined as follows:

$$UBF(l, l_i) = \alpha i^{\beta} (\text{where } \beta < 0)$$

• where α denotes the user's preference of the first item, which represents the user's first impressions on the item list, and β denotes the user preference decay factor.

User Behaviour Function (Contd.)



Search engine vector voting(SVV)

- SVV is based on the voting concept that a web page's ranking is dependent on how several selected search engines rank it, rather than its own contents. A web page wins a vote from a particular search engine if it is listed in the search engine's results to a given query.
- SVV currently gathers top 10 items of search results from each of the following four well-known search engines, Yahoo, Bing, Ask and AOL.
- The SVV search method rearranges the returned web pages based on their weights. The weight of a particular web page considers both the voting tendencies of the 4 search engines mentioned above and the user behaviour function (UBF)

Weight and Vote Distribution of Webpage

• The weight of a web page p for a user query q is defined as follows:

$$w_{p,q} = \sum_{i=1}^{\mathbf{4}} lpha_{i,q} x_{i,p,q}^{\ eta}$$

- According to this definition, a web page clearly has larger weight if it either wins votes from more search engines or is ranked high in the results of at least one search engine.
- To understand the voting tendency of the four search engines on a particular web page, a web page's vote distribution is also provided with the following formula:

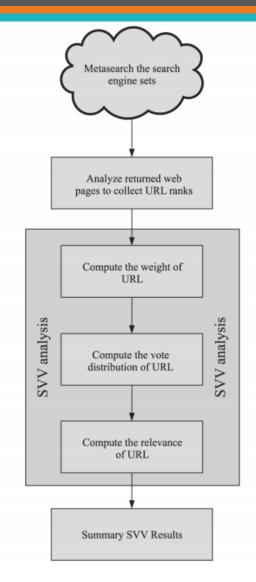
$$P_{p,q} = w_{p,q} / \sum_{i=1}^{4} \alpha_{i,q}$$

Relevance of Webpage

• The vote distribution alone is insufficient for measuring the relevance of returned web pages to a given query. For instance, occasionally the votes on a web page are sparsely distributed among the six search engines, and the resulting almost empty rectangle may lead the user to misinterpret its relevance. Thus the relevance of a web page to a given query is also defined as follows:

$$R_{p,q} = \begin{cases} High, & w_{p,q} > \bar{w} + n\sigma_w \\ Middle, & \bar{w} < w_{p,q} \le \bar{w} + n\sigma_w \\ Low, & otherwise. \end{cases}$$

• n denotes any value greater than 1, we have taken n=2.



On The Fly Document Clustering (OTFDC)

- This clustering algorithm, called On-The-Fly Document Clustering (OTFDC), produces a number of candidate clusters from other web search results. The candidate clusters not only generate a connective relation between the clusters, but also the relation is a semantic one.
- In OTFDC analysis, we also assumed the candidate clusters derived from an important web document were more likely to be related to the source cluster. Thus, OTFDC only considers the candidate cluster t when the weight of a web document $w_{\rm pt,m}$ is greater than a Threshold. To realize the similarity degree of any two clusters, OTFDC uses the following equation to calculate the correlation coefficient between any two clusters:

 $R(Cluster_i, Cluster_t) = \min(Norm_i, Norm_t)/Norm_i$

Reverse Links and Building Ontologies

- Three features:
 - SR
 - a) Equivalence
 - b) Hierarchy, and
 - c) Association.
 - HSR
 - CHSR

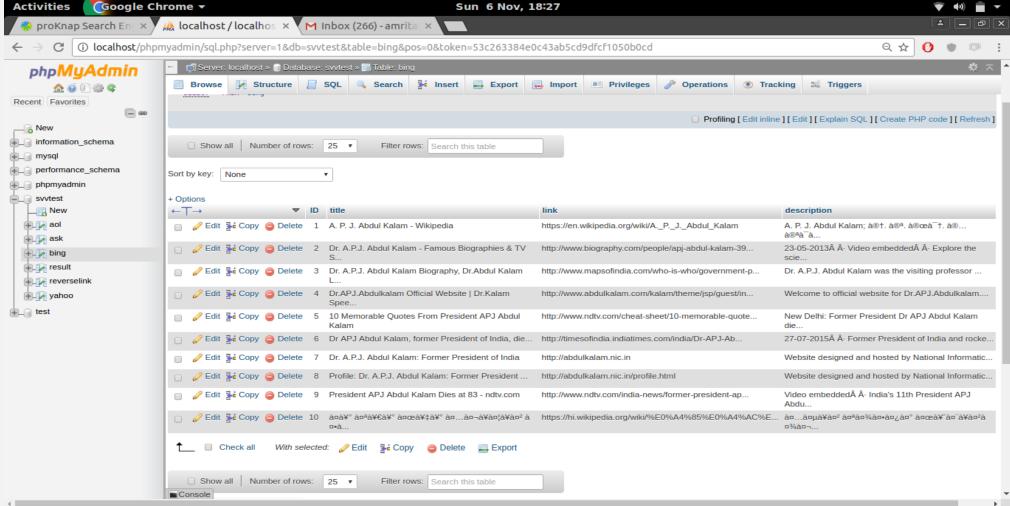


OTFDC Algorithm

```
The pseudo code of OTFDC_Core is listed as follows:
     Algorithm OTFDC_Core (Cluster, as String)
     (wp_{i,m}, w_{wp_{i,m},i}) = Call SVV (Cluster_i);
     (wp_{i,m}, w_{wp_{i,m},i}) = Sort wp_{i,m} according to its weight w_{wp_{i,m},i};
     Normalize the weight w_{wp_{i,m},i};
     (CandidateCluster<sub>t</sub>) = Use the reverse links of an important web document wp_{i,m} to find out all the candidate
     clusters.
     Foreach (CandidateCluster,)
     (wp_{t,m}, w_{wp_{t,m},t}) = Call SVV (CandidateCluster_t);
     (wp_{t,m}, w_{wp_{t,m},t}) = Sort wp_{t,m} according to its weight w_{wp_{t,m},t};
     If (w_{wp_{t,m},t} < T)
12
       Continue:
     Normalize the weight w_{wp_{t,m,t}};
     Compute the correlation coefficient;
    If (R(Cluster_i, CandidateCluster_t) \ge T and CandidateCluster_t \notin FinalClusterSet)
16
       Append CandidateCluster_t into FinalClusterSet;
17
       Append R(Cluster_i, CandidateCluster_t) into FinalCorCoefSet;
    } End of If;
     } End of Foreach;
     (FinalClusterSet, FinalCorCoefSet) = Sort FinalClusterSet according to its correlation coefficient FinalCorCoefSet;
     Return (FinalClusterSet, FinalCorCoefSet);
23 } End of Algorithm;
                                                                                                         7 November, 2016
```

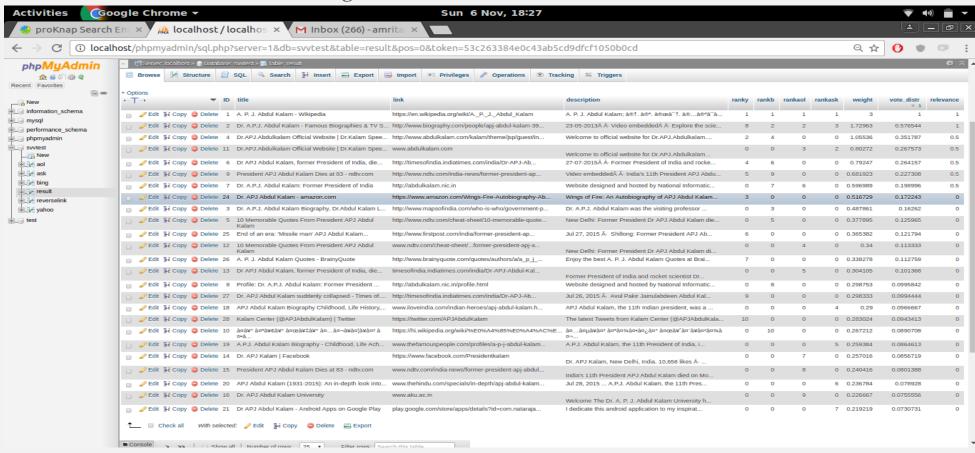
Methodology

- 4 popular search engines were chosen -Bing, Yahoo, Ask, AOL
- Assigned weights to each of them based on popularity
- HTML pages of web results were collected using BeautifulSoup and Urllib libraries in Python
- Title, link, description were collected using information of their enclosing HTML tags
- Results from each search engine were stored into separate tables in a database using MySQL

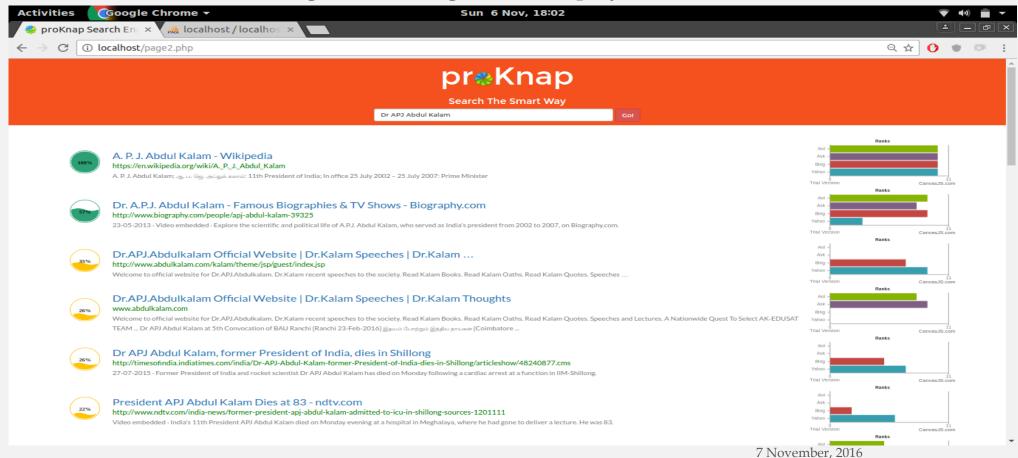


- Serial Scraping of all 4 search engines : 20 seconds
- Parallel Scraping: 10 seconds
- Parallelization achieved through Python's Multithreading library

Results were combined, and weights and vote distribution was calculated and stored



- GUI designed with HTML, CSS, JavaScript and PHP
- All results sorted according to their weights and displayed



Approaches / Challenges faced in SVV

- Scraping Google
- Tried to use search engine API
- Removing video and image results
- Making URLs uniform
- Time consuming to scrape 4 search engines

• Implemented OTFDC in Python

Challenges:

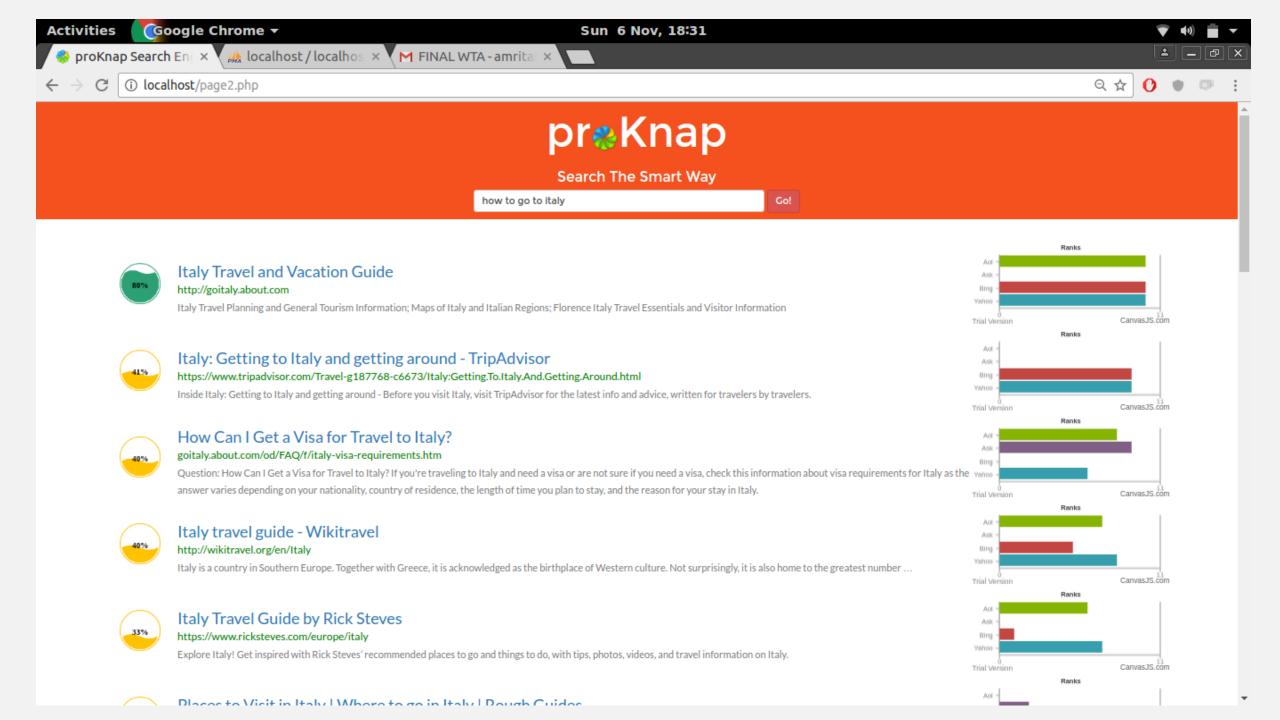
- Get reverse links
- Time Consuming
- Most important link not present in successive iterations, or relevance does not exceed threshold

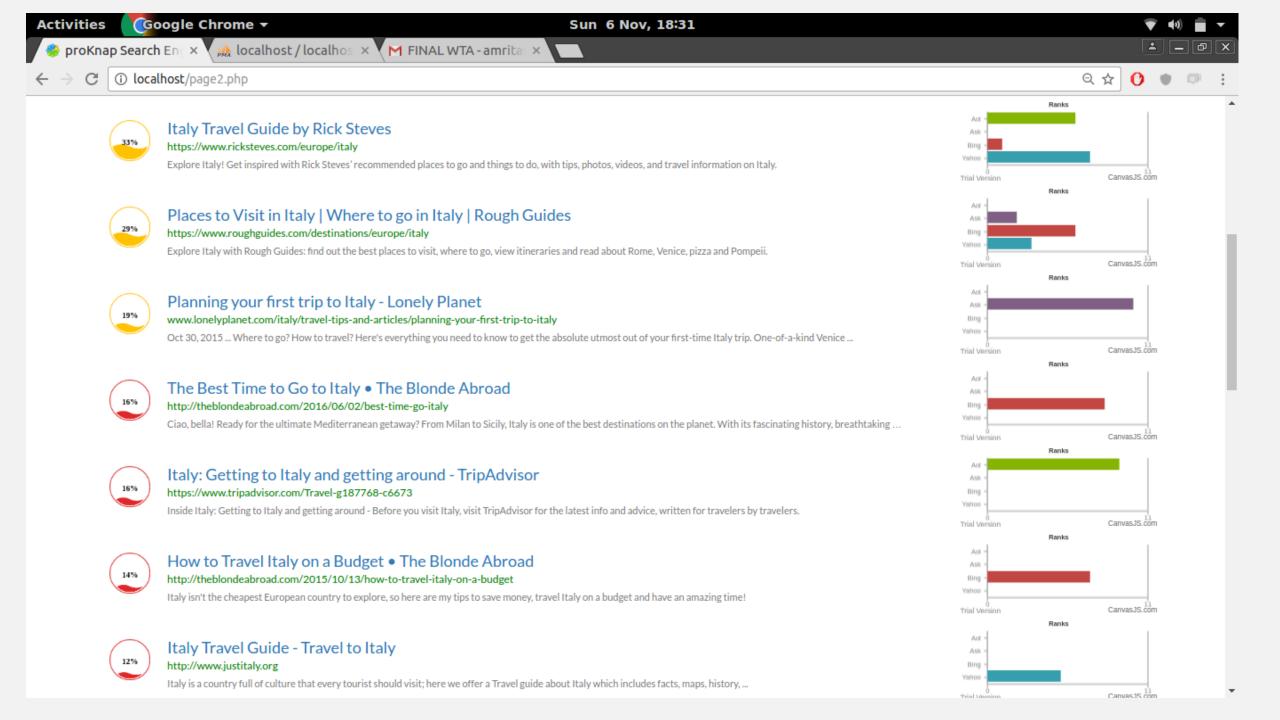
Solution:

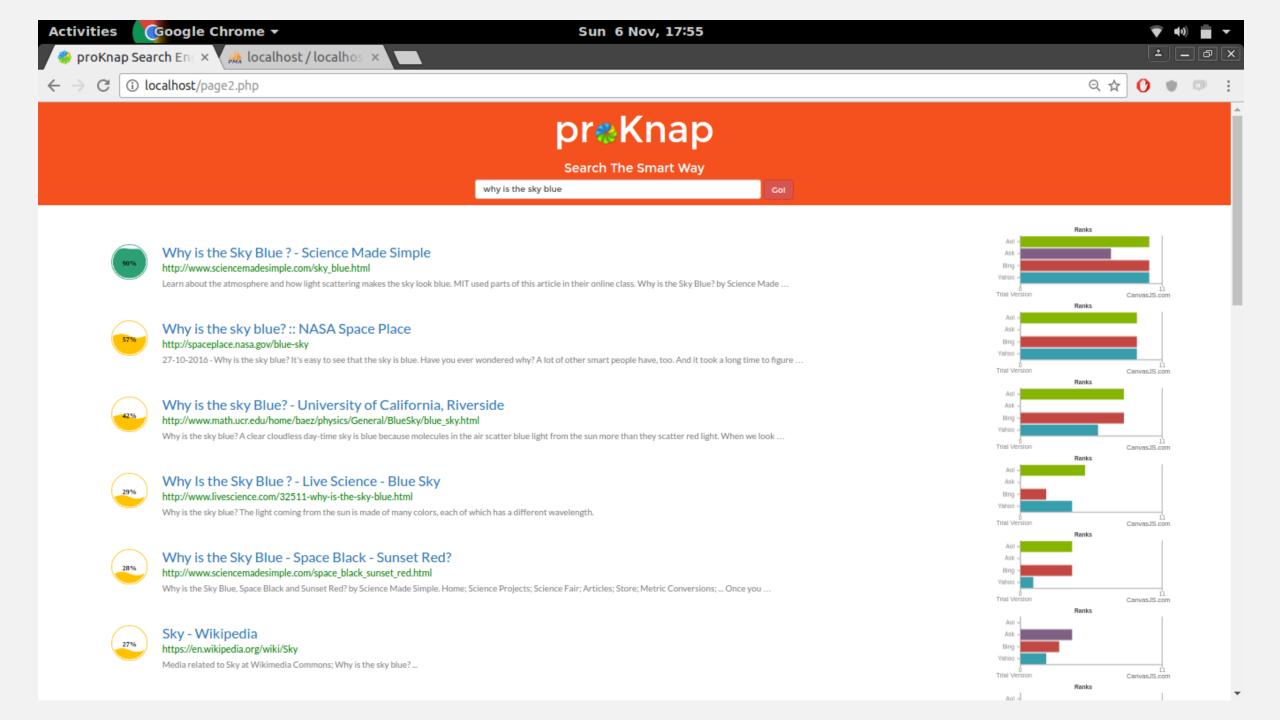
- Used semantically related queries based on HSR values
- Use OTFDC on closed domain

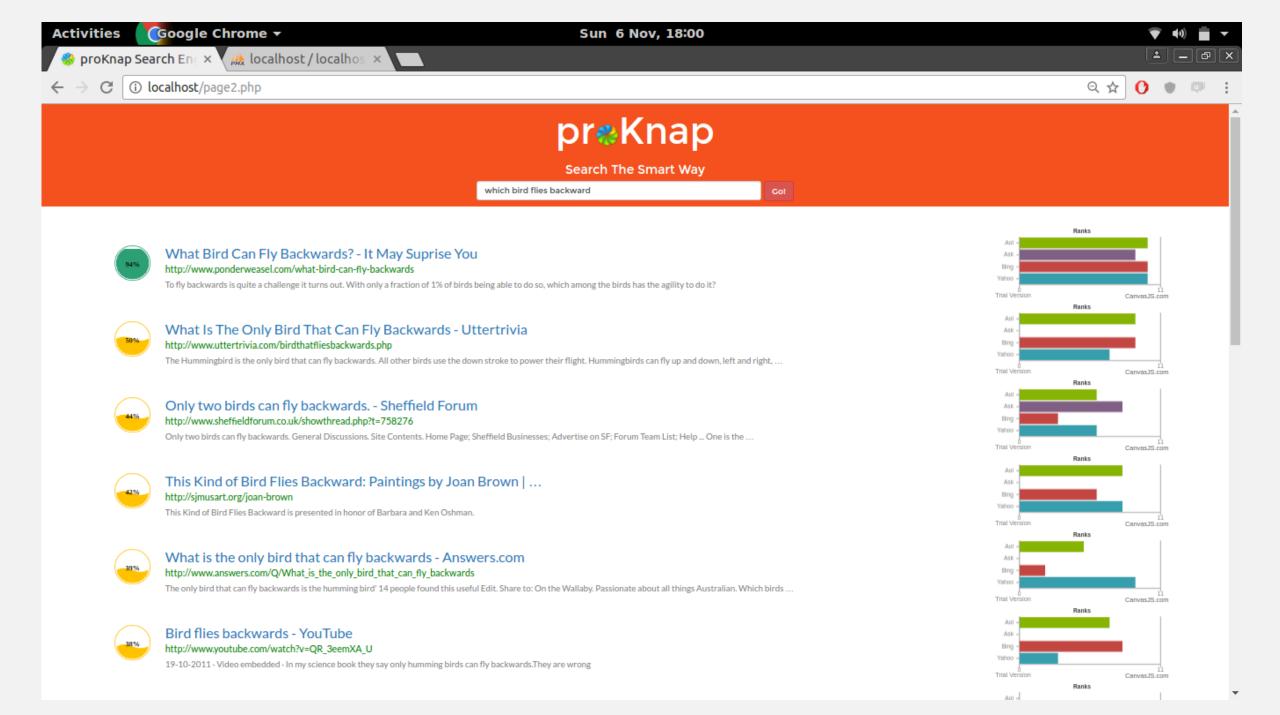
Results

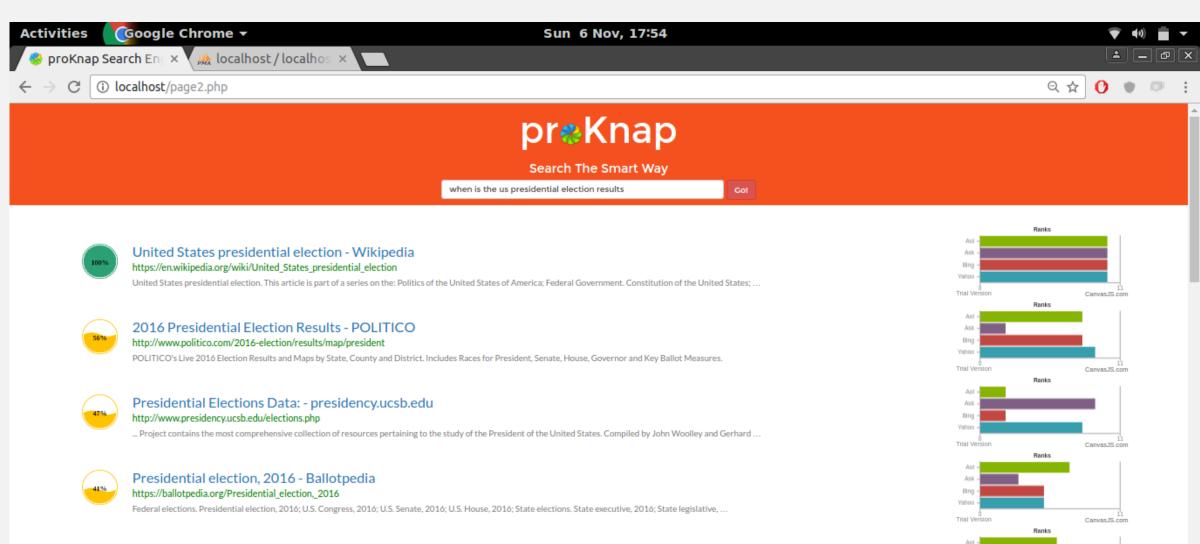
- The SVV algorithm was successfully implemented
- Some of the queries that received successful output are
 - Who
 - When
 - What
 - Why
 - Which
 - How
 - Named Entities











Election Results - Dave Leip's Atlas of U.S. Presidential Elections

http://uselectionatlas.org/RESULTS

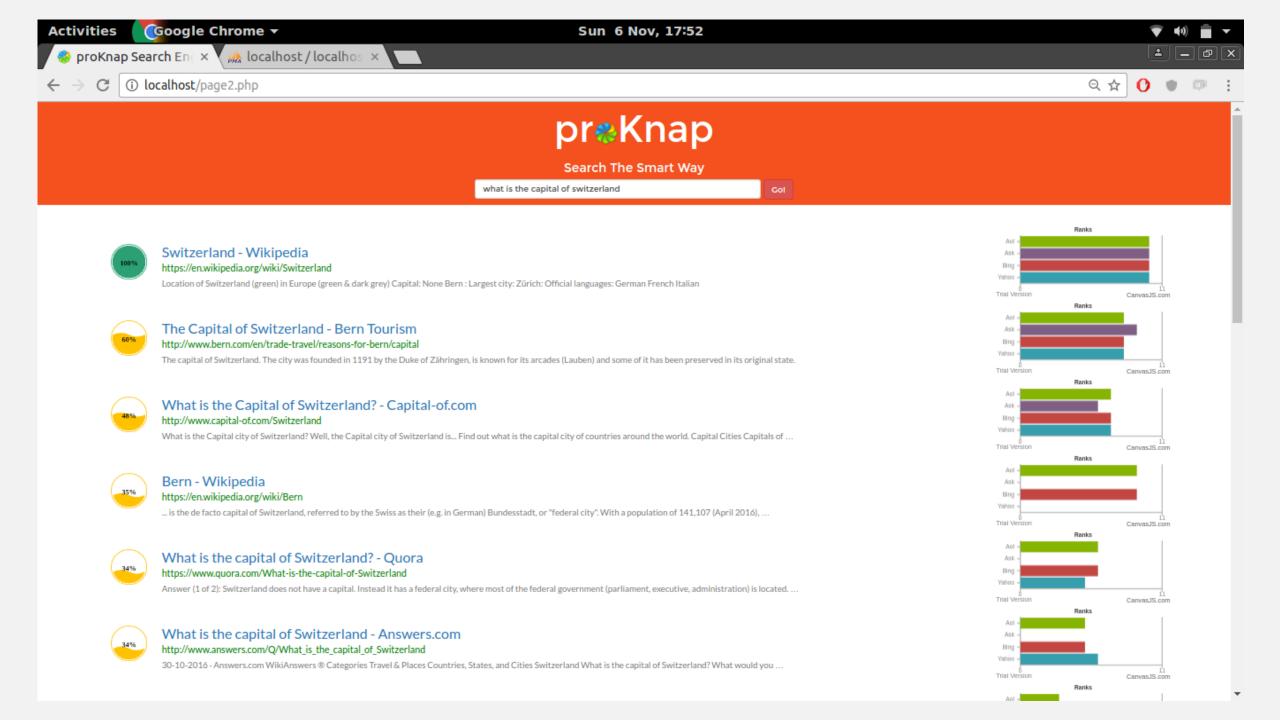
Detailed national results of US Presidential Elections from 1789 through 2012, US Senate and Gubernatorial Elections since 1990. Site includes election data, county ...

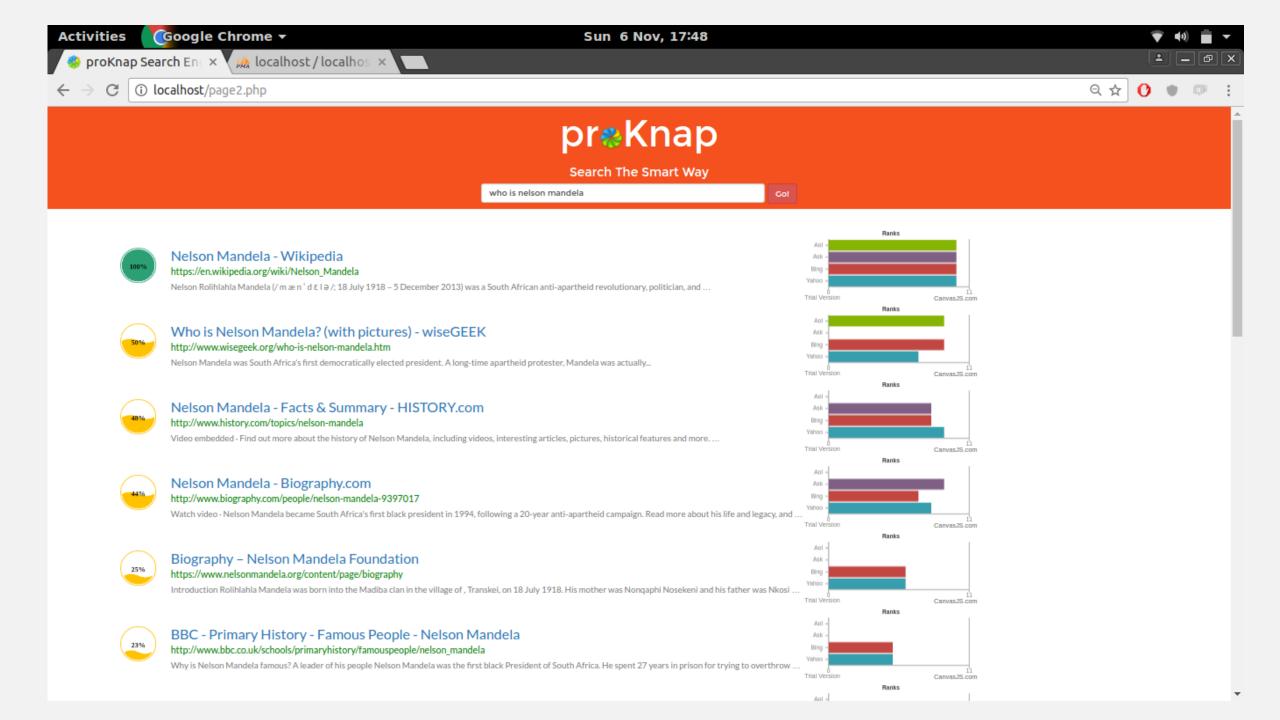
United States presidential election, 2012 - Wikipedia

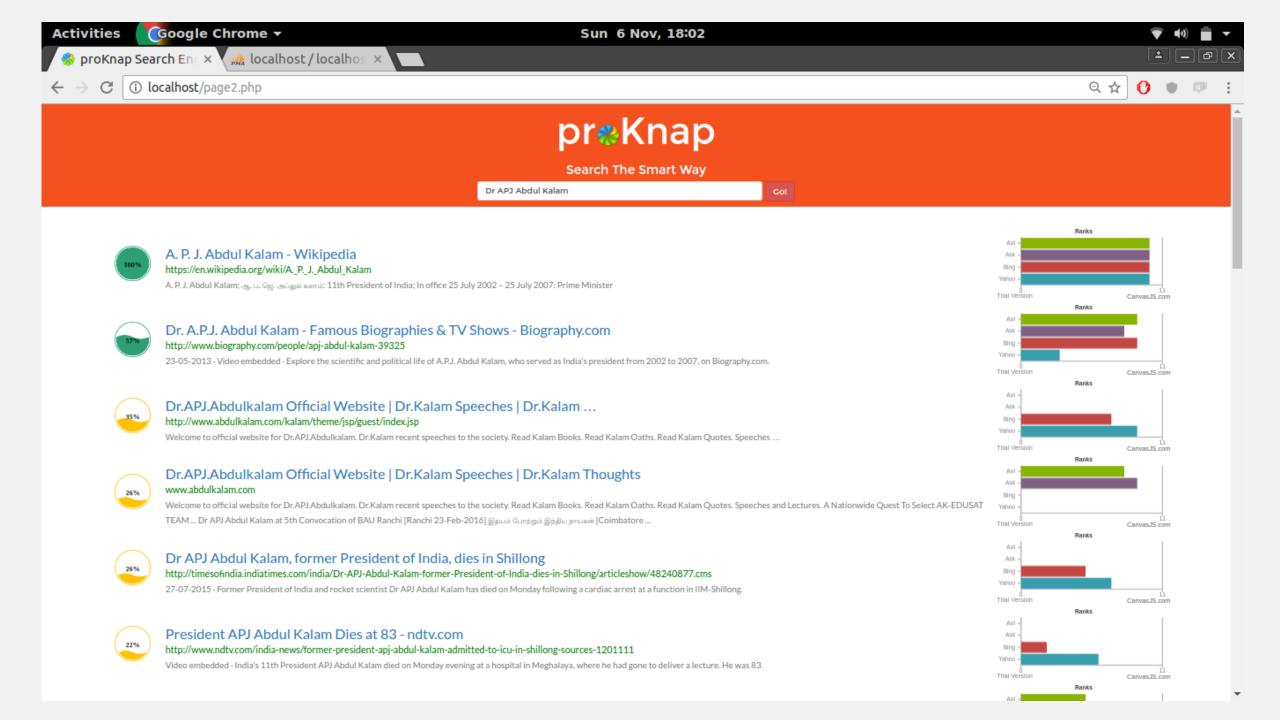
https://en.wikipedia.org/wiki/United_States_presidential_election,_2012

Presidential election results map. Red denotes states won by Romney/Ryan (24), Blue denotes those won by Obama/Biden (26+D.C.). Numbers indicate electoral votes ...

Yahoo Trial Version CanvasJS.com Banks Anl Ask Yahoo Trial Version Canvas 15 com







Conclusion

- A survey conducted by authors of the paper, showed that users were more satisfied with the proposed search method than general search engines
- Ultimately the 'goodness' of the search results depends on the user behavior input given

Future Work

• Results can be improved by using ontologies that would be created when the net becomes fully semantic

• Collect user feedback to get accurate values of α and β

Reduce execution time

Thank you