A Machine Learning Approach to Build Web Search Engine Using Both Link Analysis and Textual Content

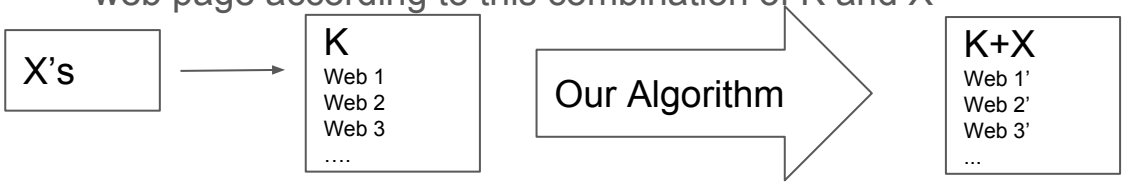
ECS 251 Spring 2016 Final Project
Guicheng Wu
Xingtai Li

Outline

- Introduction
  - Information Retrieval
  - Link Analysis of Web page
  - Textual Relevance between pages
- Our Approach
  - Dataset and UML diagram of Data Structure
  - Feature Engineering
  - Machine Learning Approach
  - Ranking Algorithm
  - Page Relevance Scoring Technique
- Evaluation of our implementation
  - Machine Learning Results
  - Sample Ranking Example
- Conclusion and Future Suggestion

Introduction (Information Retrieval)

- Retrieve documents in digital collections
  - Estimate the relevance of a page to a query
  - Web-page search engine combines features to estimate relevance
  - Specific features and mode of combination makes search engine different
- Problem to be solved
  - The size of web is growing, indexable pages exceeds 8 billion
  - Search Engine has difficulty to keep up-to-date and comprehensive search index
  - Users have trouble to get useful and high-quality information using SE
- Our Focus:
  - Given a specific keyword (K), if user add another keyword (X), we can provide high quality web page according to this combination of K and X



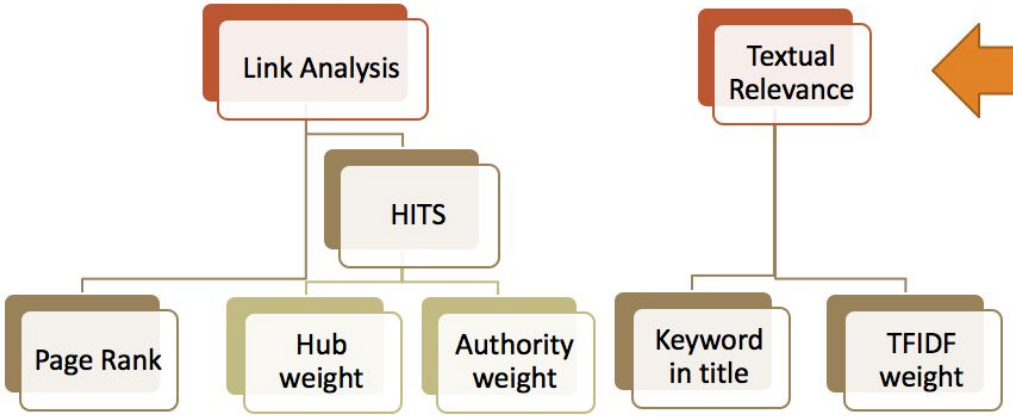
Introduction (Link Analysis)

- PageRank algorithm
  - Computed by weighting each in-link to a page proportionally to the quality of those pages
  - The higher the score, the better the quality the referring page
  - Problem: computation time
- HITS algorithm (Hyperlink-induced Topic Search)
  - Authority Score : A page to which many other pages
  - Hub Score: A page that points to many other pages
- Other Algorithm:
  - SALSA: Combine page rank and hits algorithm

Introduction(Textual Relevance)

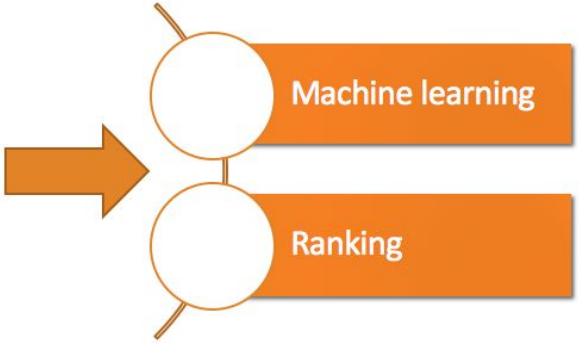
- TFIDF - Term Frequency- Inverse Document Frequency
  - Statistical way to reflect how important a word to a document in a collection of corpus
  - A web page containing words that are found in the list can be considered more revelant
- Title of Page
  - Extract keywords from the title of the webpage
- URL Address
  - Contains useful information about the page (domain address, authoritative ‘.com’ < ‘.gov’)
  - Some believe that less slashes is more useful than those with more slashes

Our Approach



Input: Keyword

	Page rank	Hub weight	Authority weight	Keyword in title	TFIDF weight	(Label 1 or 0)
Web page 1						0
Web page 2						1
Web page 3						1



Dataset we used

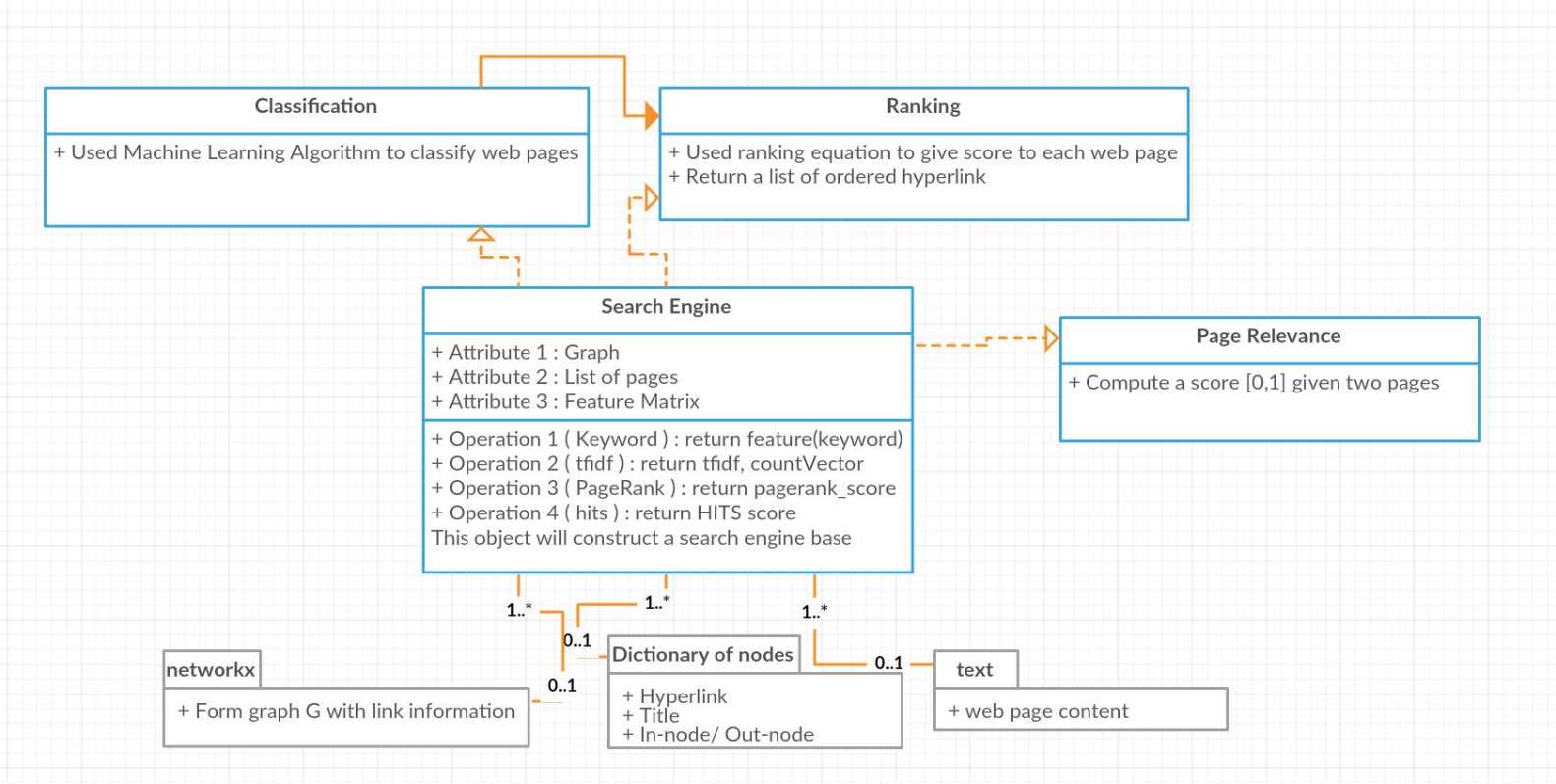
- **Data Sets for Link Analysis Ranking Experiments**
  - http://www.cs.toronto.edu/~tsap/experiments/download/download.html
- Node
  - We used the data set for “Basketball”
  - There are 6049 nodes

```
0 (0) [R]
http://www.nba.com
NBA.com
697 5
```

- Adjacency List
  - Number of edges: 24409

```
1: 256 258 262 270 273 274 275 -1
2: 316 317 318 319 320 321 322 323 324 325 326 -1
```

Data structure of our code



Feature Engineering

- Page Rank Score(p) = || (1-d) + d* ∑ [(Pagerank(q))/c(q)] ||2
- Authority Score (p) = ∑ (Hubscore(q))
- Hubscore (p) = ∑ (Authorityscore (r))
- Title Score (p,query) = ∑ (query relevant words)
- Content Score (p,query) = tf * (idf + 1)
    - tf = term frequence; idf = inverse document-frequency

```
7.608704431755380134e-04 1.28941667708331437e-50 7.210245637026558836e-05 2.000000000000000000e+00 7.729578888250805691e-02
1.112765831980647071e-03 0.000000000000000000e+00 2.063411444187389799e-04 1.000000000000000000e+00 8.122090484421674861e-02
1.166884507953014967e-04 2.031920320556255002e-06 6.849374380577110820e-05 1.000000000000000000e+00 0.000000000000000000e+00
```

Machine Learning Approaches

- Supervised learning:
  - Ridge regression
  - K nearest neighbor
  - Support vector machine
  - Bayes gaussian
  - Logistic regression
  - Linear discriminant analysis
  - Lasso
  - Decision tree

Statistical metrics

For classification tasks, the terms *true positives(TP)*, *true negatives(TN)*, *false positives(FP)*, and *false negatives(FN)* compare the results of the classifier under test with trusted external judgments. FP is type I error; FN is type II error.

Precision Rate

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

Recall Rate

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

F1-Score

$$\text{F1-score} = 2*(\text{Precision} * \text{Recall}) / (\text{Precision} + \text{Recall})$$

Ranking Equation

Ranking function returns a list of hyperlinks of web page. The order is based on the rank score assigned to each page:

Rank Score (query,page) = α * pagerank(page)

+ β * authority(page)*(1 + E[pagerank(incoming neighbors)])

+ γ * hus(page)*(1 + E[pagerank(out neighbors)])

+ ϕ * Title(query,page)

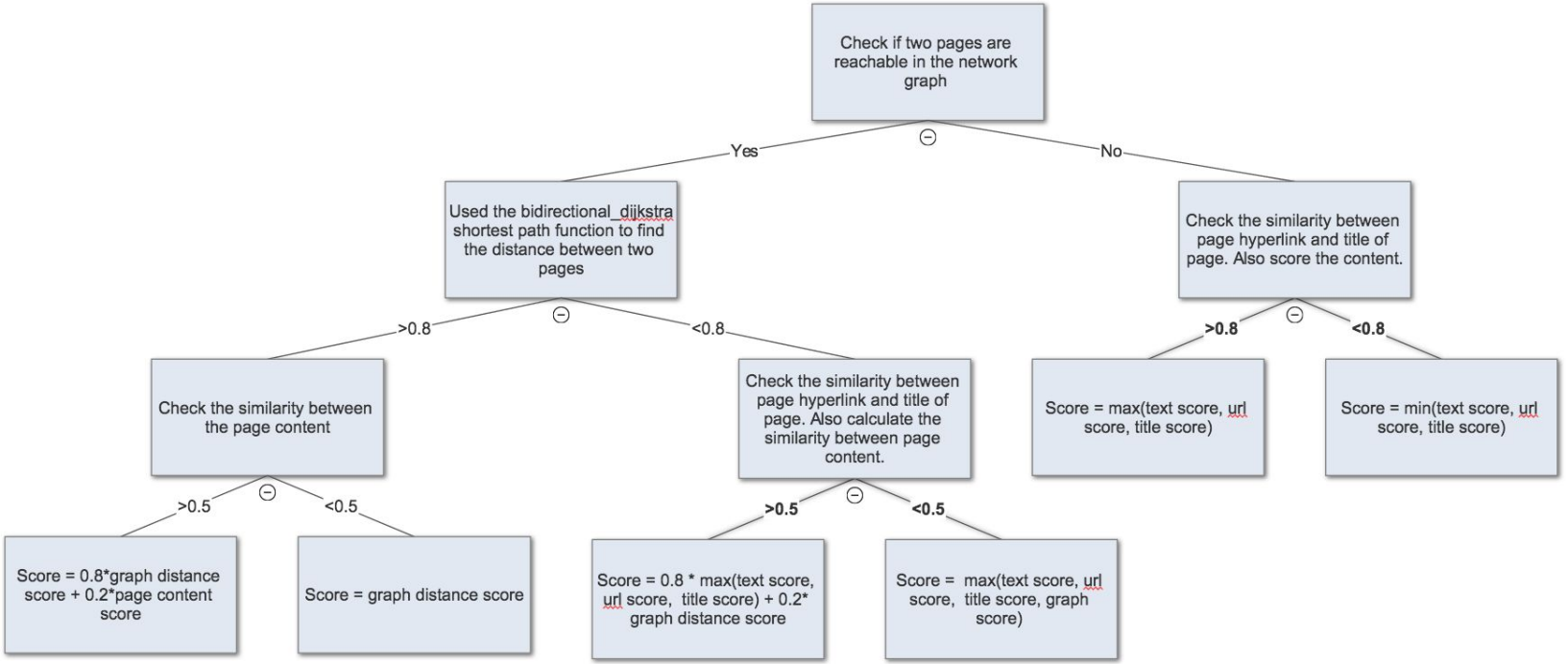
+ φ * □ (TFIDF(query relevant,page)) / (number of relevant query)

Page Relevance

- Decision Tree based on graph and textual content:
  - Use Graph
    - Check if there is path between two nodes
    - If there is path, use bi-directional dijkstra method to calculate the distance
  - Use textual content
    - Check the useful keywords in both hyperlink and title of pages
    - Use CountVector to calculate the frequency of words in web page

- Multivariate Classification
  - Use the classifier for each keyword to label each page
  - Calculate the two similarity between two pages
    - Time-consuming & Complexity

Page Relevance (Decision Tree)



Evaluation and Results

Query Selection:

```
query = ['news','season','sports','team','NBA','women','schedule','college','NCAA','player']
```

Classification and Ranking:

```
classifier = ['SVM','kNN','Ridge Regression','Bayes Gaussian','Logistic Regression','LDA','Decision Tree','Lasso']
```

Result Representation:

```
def print_score(self):
    np.savetxt('result/'+self.q+ '_score.txt',self.score,fmt="%s")
```

```
def save_graph(graph,file_name):
```

Result Judgement:

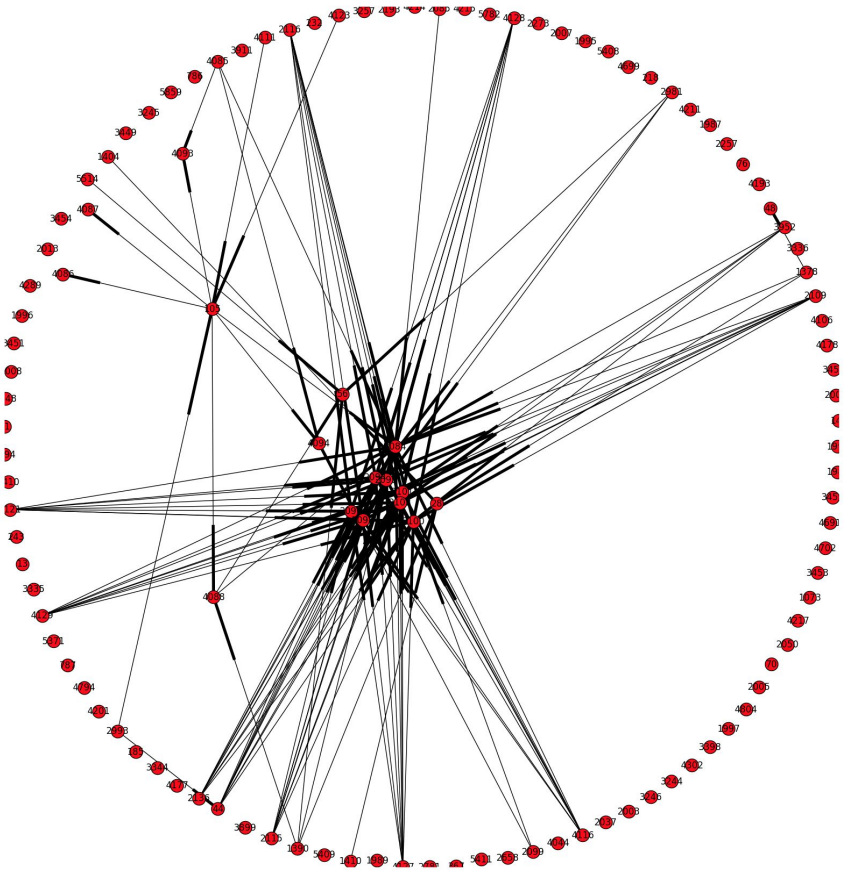
http://gostanford.fansonly.com/sports/w-baskbl/stats/teamstat.html

Machine learning Result table

Basketball dataset: Prediction results					
Algorithm	class	precision	recall	f1-score	test size
Ridge regression	Irrelevance(label 0)	0.85	1.00	0.92	1981
	Relevance(label 1)	1.00	0.17	0.3	439
	avg / total	0.87	0.85	0.80	2420
K nearest neighbor	Irrelevance(label 0)	0.99	0.99	0.99	1981
	Relevance(label 1)	0.97	0.95	0.96	439
	avg / total	0.98	0.98	0.98	2420
Supportvectormachine	Irrelevance(label 0)	0.88	1.00	0.93	1981
	Relevance(label 1)	0.98	0.38	0.55	439
	avg / total	0.90	0.89	0.86	2420
Bayes gaussian	Irrelevance(label 0)	0.99	1.00	0.99	1981
	Relevance(label 1)	0.99	0.94	0.96	439
	avg / total	0.99	0.99	0.99	2420
Logistic regression	Irrelevance(label 0)	0.85	1.00	0.92	1981
	Relevance(label 1)	1.00	0.23	0.37	439
	avg / total	0.88	0.86	0.82	2420
LDA	Irrelevance(label 0)	0.88	1.00	0.93	1981
	Relevance(label 1)	0.98	0.36	0.53	439
	avg / total	0.90	0.88	0.86	2420
Lasso	Irrelevance(label 0)	0.82	1.00	0.90	1981
	Relevance(label 1)	0.00	0.00	0.00	439
	avg / total	0.67	0.82	0.74	2420
Decision tree	Irrelevance(label 0)	0.99	0.99	0.99	1981
	Relevance(label 1)	0.96	0.94	0.95	439
	avg / total	0.98	0.98	0.98	2420

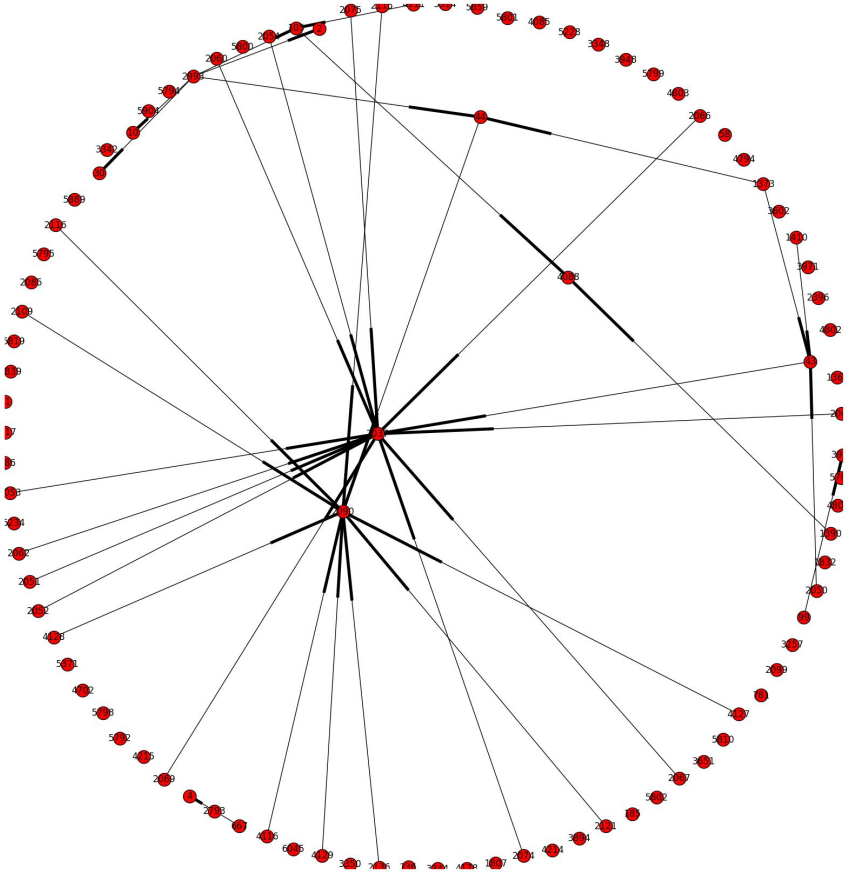
Sample Ranking Result1

Keyword: “NCAA”



Sample Ranking Result2

Keyword: “women”



Sample Ranking Result

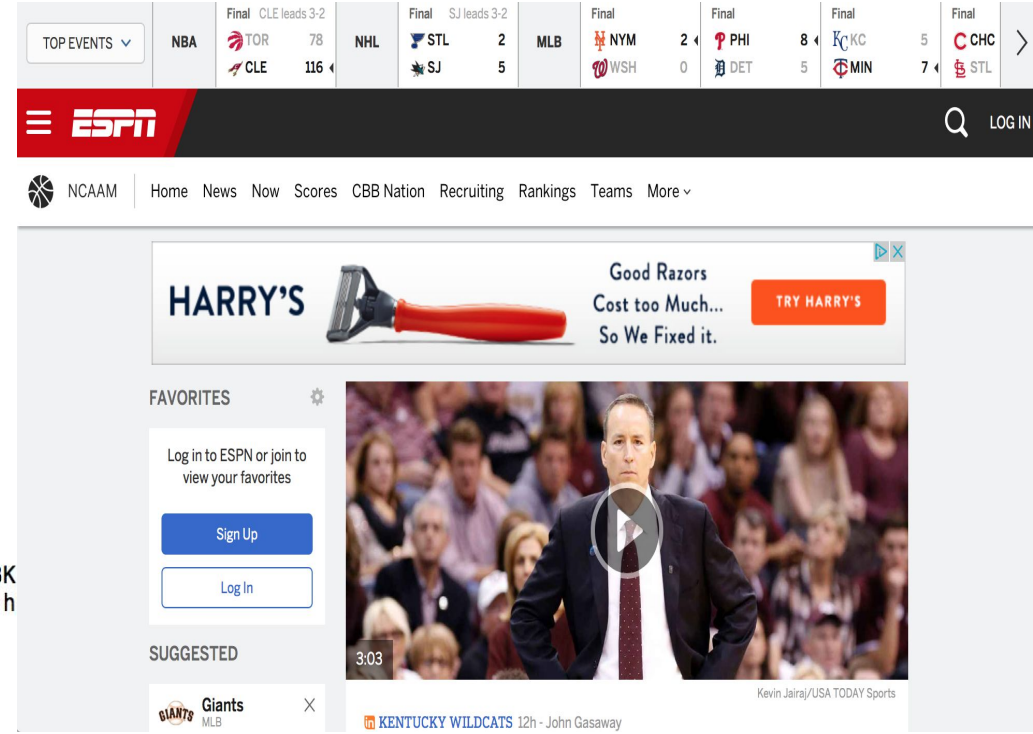
Keyword: “NCAA”

http://espn.go.com/ncb
http://www.ncaasports.com
http://sports.espn.go.com/ncb/index
http://www.ncaafootball.net
http://www.usbasket.com
http://www.finalfour.net
http://sports.espn.go.com/ncb/rankings
http://sports.espn.go.com/ncaa/index
http://espn.go.com/ncb/bracketology
http://sports.espn.go.com/ncb/statistics
http://www.sportingnews.com/cbasketball
http://sports.espn.go.com/ncb/schedules
http://sports.espn.go.com/ncb/players
http://sports.espn.go.com/ncb/standings
http://sports.espn.go.com/ncb/scoreboard
http://www.wbca.org
http://www.ncaa.org/champadmin/basketball/officiating_bulletins
http://www.socalhoops.com
http://www.ukfans.net/jps/uk/Statistics/statistics.html
http://sports.espn.go.com/sports/tvlistings/scheduleWeek?sport=BK
http://fantasygames.sportingnews.com/hoops/playoffs/basic/index.html?aff_origin=nav_top_ult_cbk
http://sports.espn.go.com/ncb/powerranking?pollId=2
http://ncaasports.com/basketball/womens

Sample Ranking Result

Keyword: “NCAA”

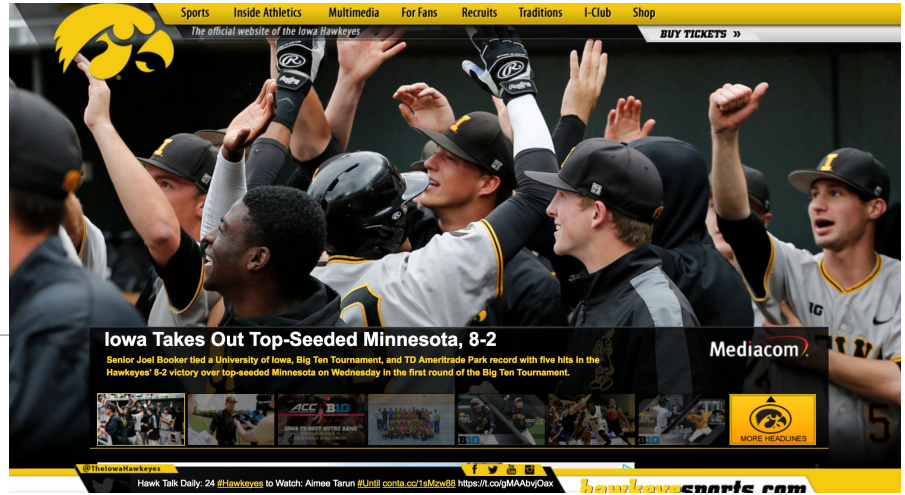
```
http://espn.go.com/ncb
http://www.ncaasports.com
http://sports.espn.go.com/ncb/index
http://www.ncaafootball.net
http://www.usbasket.com
http://www.finalfour.net
http://sports.espn.go.com/ncb/rankings
http://sports.espn.go.com/ncaa/index
http://espn.go.com/ncb/bracketology
http://sports.espn.go.com/ncb/statistics
http://www.sportingnews.com/cbasketball
http://sports.espn.go.com/ncb/schedules
http://sports.espn.go.com/ncb/players
http://sports.espn.go.com/ncb/standings
http://sports.espn.go.com/ncb/scoreboard
http://www.wbca.org
http://www.ncaa.org/champadmin/basketball/officiating_bulletins
http://www.socalhoops.com
http://www.ukfans.net/jps/uk/Statistics/statistics.html
http://sports.espn.go.com/sports/tvlistings/scheduleWeek?sport=BK
http://fantasygames.sportingnews.com/hoops/playoffs/basic/index.h
aff_origin=nav_top_ult_cbk
http://sports.espn.go.com/ncb/powerranking?pollId=2
http://ncaasports.com/basketball/womens
```

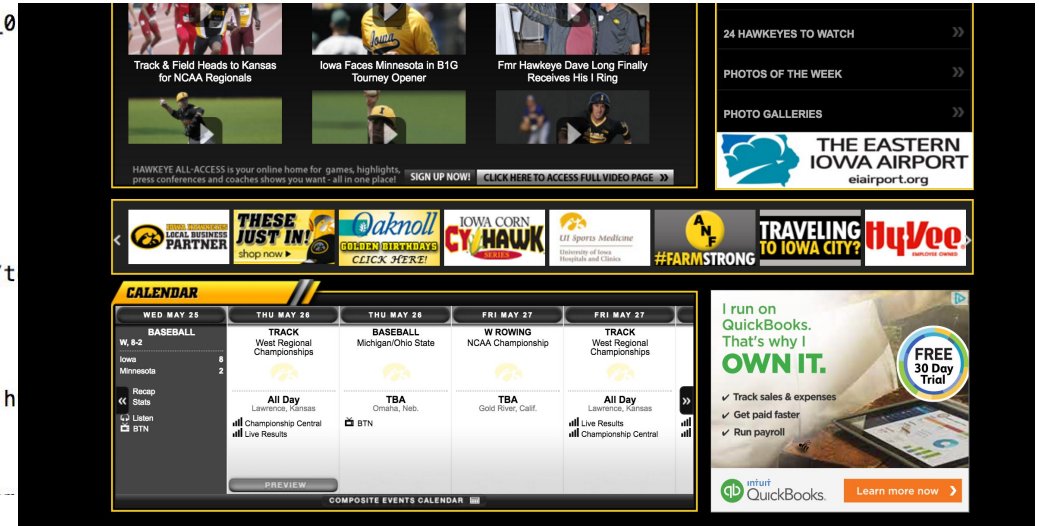


Sample Ranking Result

Keyword: “Schedule”

```
http://www.hawkeyesports.com
http://www.hawkeyesports.com/bball/2003/news/041303.asp
http://espn.go.com/nba/s/2002/0829/1424660.html
http://www.hawkeyesports.com/bball/2003/news/032403.asp
http://www.hawkeyesports.com/news/2002/nitstory2.asp
http://www.athletics.uwaterloo.ca/camprec/comp_leagues/refs.htm
http://www.heartlandconf.org/links.htm
http://www.ocaa.com/aboutus/theocaa.html
http://www.texassports.com/mainpages/wbb_pages/2001_02/006/wbb_0
http://www.hawkeyesports.com/bball/2003/news/031303.asp
http://www.hawkeyesports.com/bball/2003/news/031203.asp
http://www.hawkeyesports.com/bball/2003/news/031103.asp
http://www.hawkeyesports.com/bball/2003/news/030803.asp
http://www.und.com/sports/w-baskbl
http://www.hawkeyesports.com/bball
http://sports.espn.go.com/nba/roster?team=sea
http://www.hawkeyefootball.com
http://www.hawkeyesports.com/news/2002/012903a.asp
http://www.fansonly.com/schools/tennw/sports/w-baskbl/spec-rel/t
http://interact.fansonly.com/nl_sign_up/index.cfm?nl_code=md
http://www.ivyleaguesports.com/news.asp?id=90
http://www.hawkeyesports.com/news/2002/084.htm
http://www.hawkeyesports.com/wrestle
http://tarheelblue.ocsn.com/sports/w-lacros/spec-rel/020503aaa.h
http://www.hawkeyesports.com/news
http://www.hawkeyesports.com/news/2002/082.htm
http://www.hawkeyesports.com/basket/index.asp
```





Page Relevance Example

```
Web page 1: http://www.usatoday.com/sports/basketba/skm/acc/skma08.htm
Web page 2: http://www.usatoday.com/sports/basketba/skm/pac10/skmg09.htm
Page Relevance Score: 0.932203389831
```

```
Web page 1: http://gostanford.fansonly.com/sports/w-baskbl/stats/teamstat.html
Web page 2: http://gostanford.fansonly.com/sports/w-baskbl/archive/stan-w-baskbl-archive.html
Page Relevance Score: 0.775510204082
```

```
Web page 1: http://www.nba.com
Web page 2: http://www.cnn.com
Page Relevance Score: 0.911111111111
```

Page Relevance Example

```
In [194]: relevance_pbject = page_relevance(engine)
          test = relevance_pbject.print_scores()
          test

Out[194]: ['Web page 1: http://uclabruins.ocsn.com/sports/m-baskbl/spec-rel/ucla-wooden-page.html',
           'Web page 2: http://www.woodenclassic.com',
           'Page Relevance Score: 0.576923076923']
```

```
In [196]: relevance_pbject = page_relevance(engine)
          test = relevance_pbject.print_scores()
          test

Out[196]: ['Web page 1: http://und.ocsn.com/sports/m-baskbl/spec-rel/080202aaa.html',
           'Web page 2: http://www.fansonly.com/schools/unc/sports/c-track/spec-rel/021102aaa.html',
           'Page Relevance Score: 0.642201834862']
```

```
In [208]: relevance_pbject = page_relevance(engine)
          test = relevance_pbject.print_scores()
          test

Out[208]: ['Web page 1: http://mathforum.org/library/topics/estimation',
           'Web page 2: http://www.finalfour.net/local/wfinalfour_central.html',
           'Page Relevance Score: 0.263157894737']
```

Conclusion

- Naive Bayes Learning Method has better prediction than other methods in our experiment
- Given a keyword that is relevant to “basketball”, we can return a list of web pages based on their relevance to that keyword
- Page relevance gives score of >0.5 if two pages have similarity of textual content or short distance in graph, <0.5 otherwise
- Problems:
  - 10% of the web page has expired
  - Learning process is computational complex
  - Links from the same domain weights to much

```
http://sports.espn.go.com/ncb/schedules
http://sports.espn.go.com/ncb/players
http://sports.espn.go.com/ncb/standings
http://sports.espn.go.com/ncb/scoreboard
```

Future Work Suggestion

- Try our method on other dataset: “weather”, “movie”... Try dataset with more links with each other.

- Explore other features of the web page that may be helpful to rank: user behavior on the web page

- Try different size of train dataset in classification step. Validate that smaller train set will give promising classifier with similar performance