

## **Information Retrieval Project 3**

Related News for Wikipedia

12/2/2014

Team BuffaloBoys

Team Members :

Sagar Bansilal Shinde

Ankit Goyal

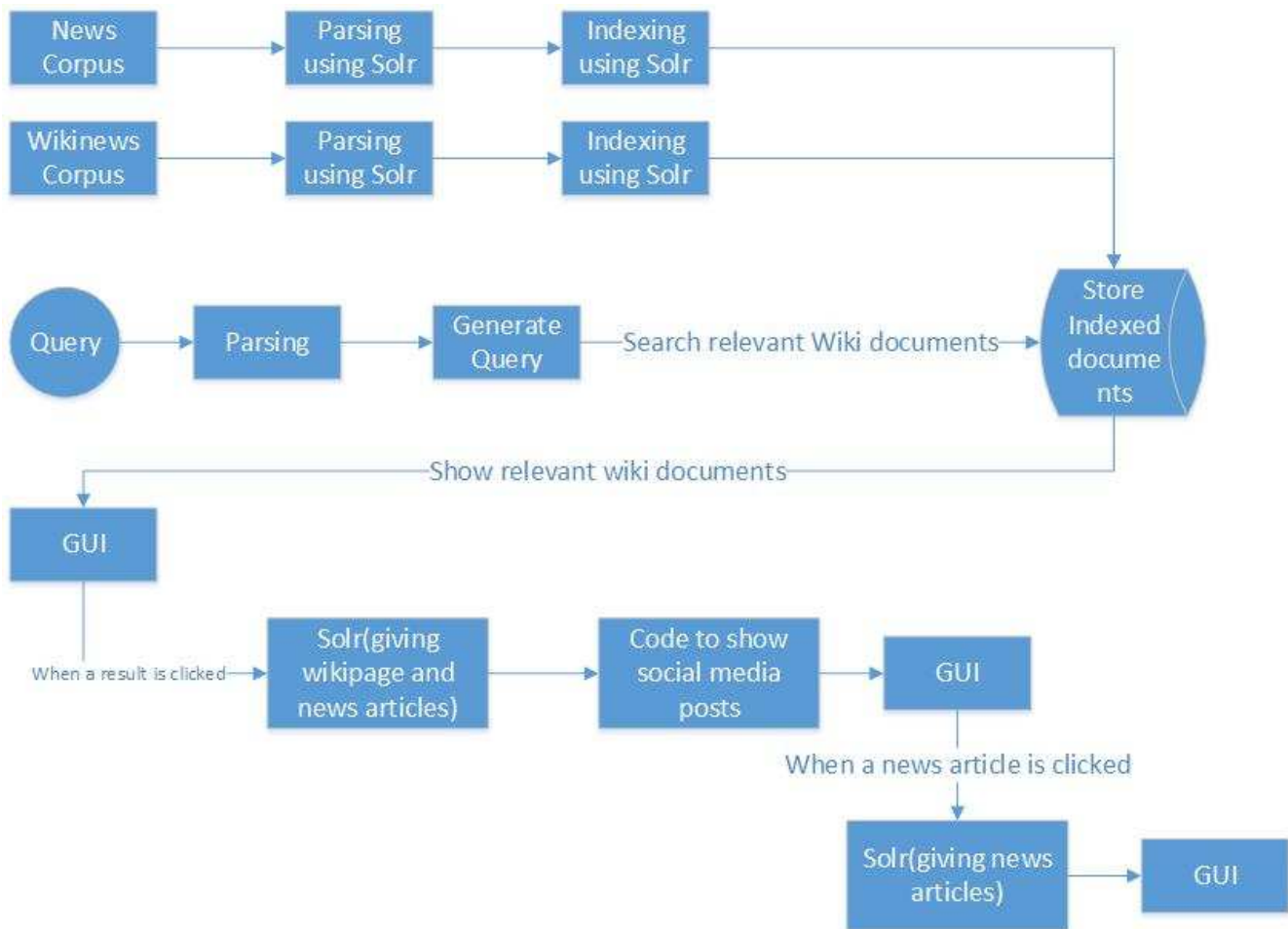
Manikanta Sandeep Kandregula

Venkata Sai Krishna Tejaswi Kasavajhula

Prepared for

CSE 535-Information Retrieval

## System Diagram:-



## Project description in detail :

### Dump Processing & Parsing:

- **News Articles:-**

For News Articles we have parsed news from 3 months of the New York Times (NYT) corpus for 2007. We have parsed news using SAX parser and extracted title, date, news body out of it. Below is the sample news.xml file.

<add>

<doc>

<field name="id">N\_1</field>

```
<field name="title">Paid Notice: Deaths BLUMENTHAL, MARTIN</field>
```

```
<field name="description">1/1/2007</field>
```

```
<field name="text"> News Body </field>
```

```
</doc>
```

```
</add>
```

Here, the data enclosed within <doc></doc> tags is a single page and there will be multiple Pages in the file. <add> </add> tags indicate the docs to be added to the index.

- **Wikipedia Articles:-**

We start with downloading enwiki dump of Wikipedia documents which is an XML file. We parsed the entire dump using SAX parser and extract title and article body in a specific Solr readable XML format, thereby reducing the size of the document to be handled.

The format of the Wiki.xml is given as:

```
<add>
```

```
<doc>
```

```
<field name="id">W_5006</field>
```

```
<field name="title">Tiger Woods</field>
```

```
<field name="text"> Wiki Article body </field>
```

```
</doc>
```

## Indexing:-

We have configured Schema.xml according to the requirements (discussed later) and news.xml and Wiki.xml is then placed in the SOLR\_HOME/example/exampledocs folder to proceed with indexing.

We start jetty by giving java -jar start.jar command and post the file to Solr for indexing with command java -jar post.jar \*.xml. If the indexing is successful we get following success message:

```
D:\IR Project\Solr2\solr-4.10.2\example\exampledocs>java -jar post.jar *.xml
SimplePostTool version 1.5
Posting files to base url http://localhost:8983/solr/update using content-type application/xml..
POSTing file news.xml
POSTing file solr.xml
POSTing file Wiki.xml
3 files indexed.
COMMITting Solr index changes to http://localhost:8983/solr/update..
Time spent: 0:00:03.455
D:\IR Project\Solr2\solr-4.10.2\example\exampledocs>_
```

## **Solr :**

We have indexed around 40,000 documents in a single core which includes both Wikipedia articles and News Stories.

To distinguish between Wiki articles and News stories, we are giving IDs to them. The IDs of Wiki articles are like W\_1, W\_2, .... and IDs for News stories are like N\_1, N\_2, .... .

### **Configuring:-**

In Solr an index is built of one or more Documents. A Document consists of one or more Fields. A field consists of a text, content, and metadata telling Solr how to handle the content.

Id is our unique key for each document.

```
<uniqueKey>id</uniqueKey>
```

```
<field name="id" type="string" indexed="true" stored="true" required="true"
multiValued="false" />
```

Our search is based on the title and text. Any data will be fetched with reference to title and text field.

```
<field name="title" type="text_general" indexed="true" stored="true"
multiValued="true" termVectors="true"/>
```

```
<field name="text" type="text_general" indexed="true" stored="true"
multiValued="true" termVectors="true"/>
```

Here both title and text field is given field type as text\_general.

This field type has a reasonable, generic, cross language defaults: it tokenizes with

StandardTokenizer, removes stop words from case-sensitive “stopwords.txt”.

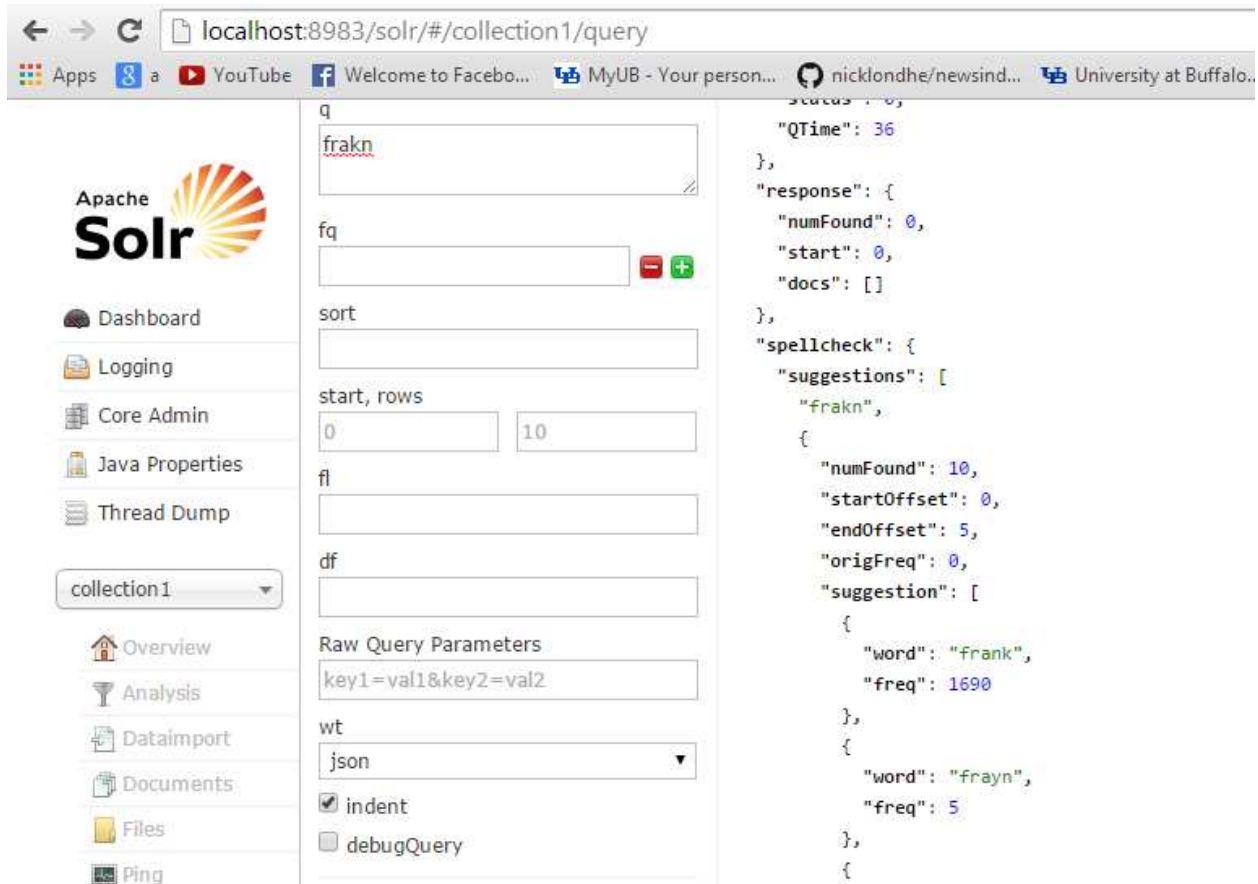
Following filters have been applied to text\_general field type:

- **solr.WhiteSpaceTokenizerFactory**
- **solr.WordDelimiterFilterFactory**
- **solr.StopFilterFactory**
- **solr.LowerCaseFilterFactory**

## Features Used of Solr :

### Spell Check (Did you mean)

If a user will enter an incorrect spelling in the query, the UI will suggest the correct spelling by asking 'did you mean' to the user. The output on Solr UI looks like this:



The screenshot shows the Apache Solr web interface. On the left is a sidebar with navigation links: Dashboard, Logging, Core Admin, Java Properties, Thread Dump, and a dropdown menu for 'collection1' containing Overview, Analysis, Dataimport, Documents, Files, and Ping. The main content area is divided into two panels. The left panel contains query input fields: 'q' with the value 'frakn', 'fq', 'sort', 'start, rows' (with values 0 and 10), 'fl', 'df', 'Raw Query Parameters' (with value 'key1=val1&key2=val2'), 'wt' (set to 'json'), and checkboxes for 'indent' and 'debugQuery'. The right panel displays the JSON response from the query. It includes a 'spellcheck' section with 'suggestions' for the word 'frakn', suggesting 'frank' (frequency 1690) and 'frayn' (frequency 5).

```
{
  "status": 0,
  "QTime": 36,
  "response": {
    "numFound": 0,
    "start": 0,
    "docs": []
  },
  "spellcheck": {
    "suggestions": [
      "frakn",
      {
        "numFound": 10,
        "startOffset": 0,
        "endOffset": 5,
        "origFreq": 0,
        "suggestion": [
          {
            "word": "frank",
            "freq": 1690
          },
          {
            "word": "frayn",
            "freq": 5
          }
        ]
      }
    ]
  }
}
```

Here a wrong spelling "frakn" is queried, in reply solr is suggesting most probable words like frank, frayn etc.

### Auto Suggest

When user will be in the middle of typing, the UI will suggest him/her the complete words which he/she might wish to write.

The screenshot shows the Apache Solr Admin UI in a web browser. The address bar displays `localhost:8983/solr/#/collection1/query`. The left sidebar contains navigation links: Dashboard, Logging, Core Admin, Java Properties, Thread Dump, and a dropdown for `collection1`. Below these are links for Overview, Analysis, Dataimport, Documents, Files, and Ping.

The main query interface includes the following fields:

- `/suggest` (Request Handler)
- `common` (Filter)
- `q` (Query): `fra`
- `fq` (Filter Query): (empty)
- `sort` (Sort): (empty)
- `start, rows` (Pagination): `0` start, `10` rows
- `fl` (Fields to List): (empty)
- `df` (Fields to Display): (empty)
- `Raw Query Parameters`: `key1=val1&key2=val2`
- `wt` (Wrapper Type): `json`

The right pane displays the JSON response for the suggest query:

```
{
  "responseHeader": {
    "status": 0,
    "QTime": 4
  },
  "spellcheck": {
    "suggestions": [
      "fra",
      {
        "numFound": 9,
        "startOffset": 0,
        "endOffset": 3,
        "suggestion": [
          "fran",
          "frat",
          "frau",
          "frawley",
          "fraxa",
          "fraxel",
          "fraxy",
          "fray",
          "fraz"
        ]
      }
    ]
  }
}
```

## More Like This

After clicking on a Wikipedia Article, the UI will suggest related news stories to it using `morelikethis` Request Handler.

The screenshot shows the Apache Solr Admin UI with the `morelikethis` request handler selected. The left sidebar is identical to the previous screenshot.

The main query interface includes the following fields:

- `Raw Query Parameters`: `key1=val1&key2=val2`
- `wt` (Wrapper Type): `json`
- `indent` (Checkboxes): ☒ indent, ☐ debugQuery
- `dismax` (Checkboxes): ☐ dismax, ☐ edismax, ☐ hl, ☐ facet, ☐ spatial, ☐ spellcheck
- `Execute Query` (Button)

The right pane displays the JSON response for the `morelikethis` query:

```
{
  "text": [
    "Sachin Tendulkar",
    "1/1/2007",
    "one of the greatest batsmen of all time.He took up cricket at the age of eleven, made his Test debut again"
  ],
  "description": "1/1/2007",
  "_version_": 1486854521939296300
},
{
  "response": {
    "numFound": 39334,
    "start": 0,
    "docs": [
      {
        "id": "H_22",
        "title": [
          "Sachin Tendulkar still lives in vacuum of own making"
        ],
        "name_autocomplete": [
          "Sachin Tendulkar still lives in vacuum of own making",
          "Everyone has his favourite Sachin Tendulkar story. A personal one is how he went incommunicado when appoi"
        ],
        "text": [
          "Sachin Tendulkar still lives in vacuum of own making",
          "1/1/2007",
          "Everyone has his favourite Sachin Tendulkar story. A personal one is how he went incommunicado when appoi"
        ]
      }
    ]
  }
}
```

## Highlighting

This feature of solr is used to generate a short snippet of the text. Using this we are displaying summary of both article and news story.



```
<?xml version="1.0"?>
<response>
  <lst name="responseHeader">...</lst>
  <result name="response" numFound="2514" start="0">...</result>
  <lst name="moreLikeThis">...</lst>
  <lst name="highlighting">
    <lst name="N_17604">
      <arr name="text">
        <str>
          Metro Briefing | New York: Manhattan: Man Charged With Posing As A <em>Lawyer</em>
        </str>
      </arr>
    </lst>
    <lst name="N_4795">
      <arr name="text">
        <str>
          An article on Tuesday about a controversy over the arrest of a political activist, Ke
          his age and misstated the surname of his <em>lawyer</em>. Mr. Krayske is 34, not 33,
        </str>
      </arr>
    </lst>
    <lst name="N_3370">
      <arr name="text">
        <str>
          Metro Briefing | New York: Brooklyn: <em>Lawyer</em> For City Contractors Charged
        </str>
      </arr>
    </lst>
  </lst>
</response>
```

## Application :

We have hosted our local website using Internet Information Services(IIS) manager. Our application is html based. We have used four html files, first html file is used to enter query, second is for Wiki articles with snippets, third is for displaying full Wiki article with News articles, social media posts to its side and a text box below the wiki article for user to comment on the article and fourth html file is used to display Full news article.

The interface between Solr and main application, navigation from one page to another, fetching Solr data and displaying it is written in javascript.

## Query Processing :

When a user enters a query, the documents are fetched from Solr in json format using following url :

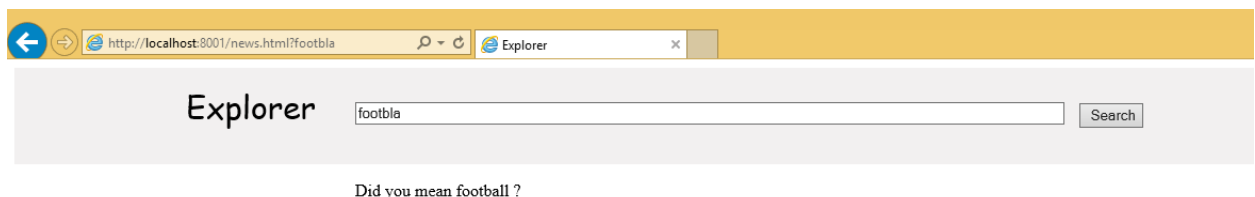
["http://localhost:8983/solr/spell?q="+x+"&fq=id%3AW\\*&spellcheck=true&spellcheck.collate=true&spellcheck.build=true&wt=json&indent=true"](http://localhost:8983/solr/spell?q=)

Above url also checks spellings of the entered query and suggests "Did you mean" if the spelling is wrong. json data returned from Solr is formatted using javascript and results of Wiki articles along with snippets is shown on second html file.

Screenshot of Homepage.

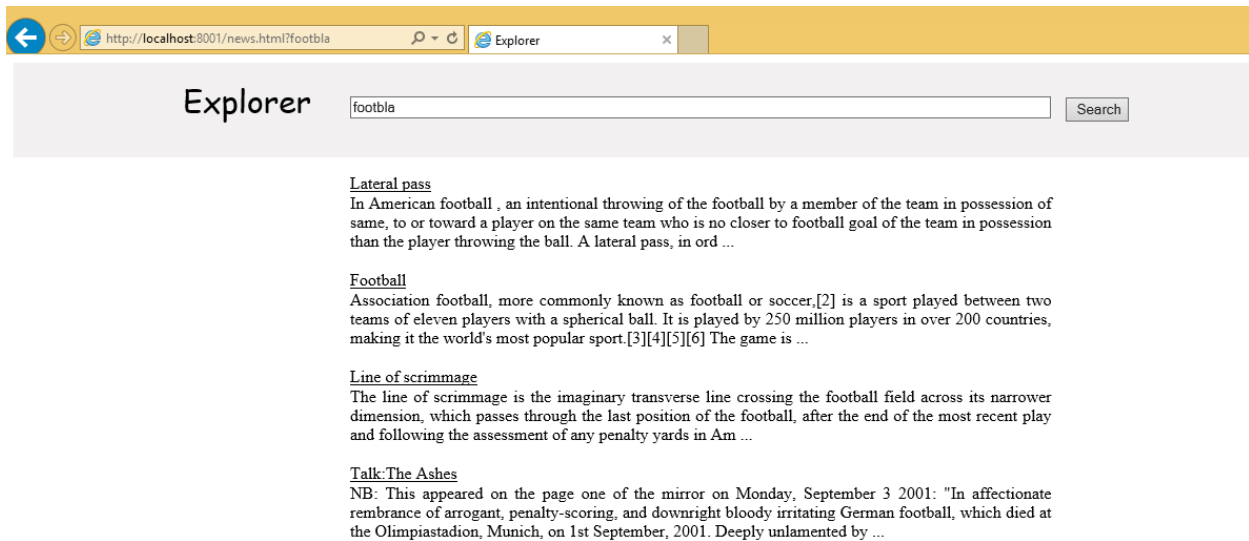


After giving a misspelled query, the system suggests the correct word.





After clicking on football, we get the desired results.

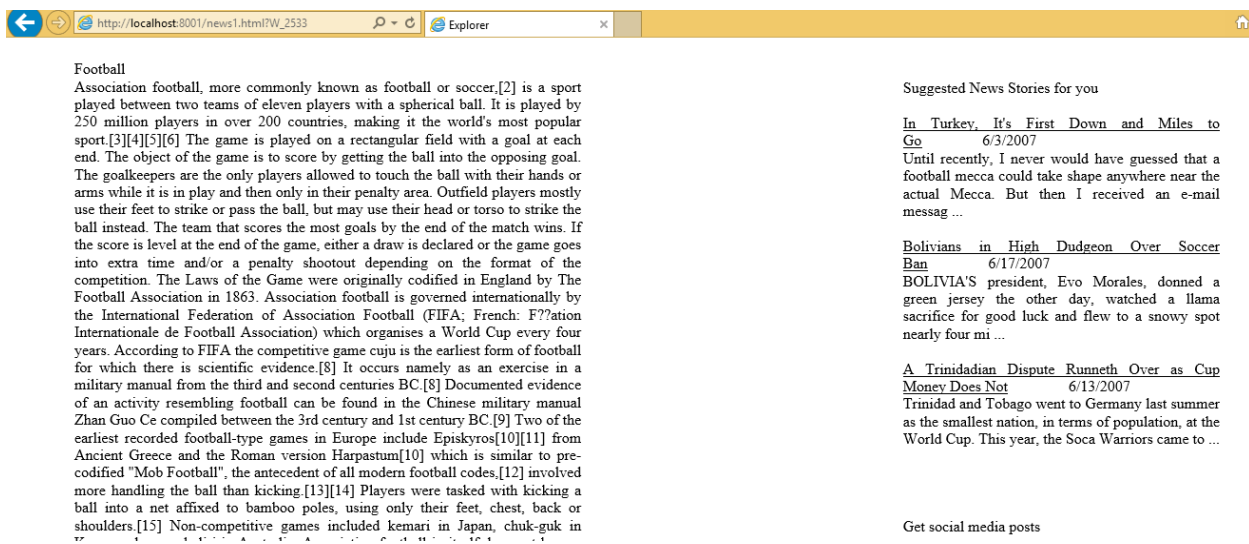


## Fetching Wiki article and related news stories :

When a user clicks on a wiki result, the full article gets opened on another page with relevant News Stories on the right side of main Wiki article. For showing relevant news stories, we have used "more like this" feature of Solr. The url we are using to fetch relevant news stories using mlt in json and then format by javascript is :

"[http://localhost:8983/solr/mlt?q=id:"+x+"&fq=id%3A%20\\*%26mlt.fl=text,title&mlt.mindf=1&mlt.mintf=1&wt=json&indent=true](http://localhost:8983/solr/mlt?q=id:)".

This query fetches only those documents whose id starts with N\_ ,i.e., news stories.

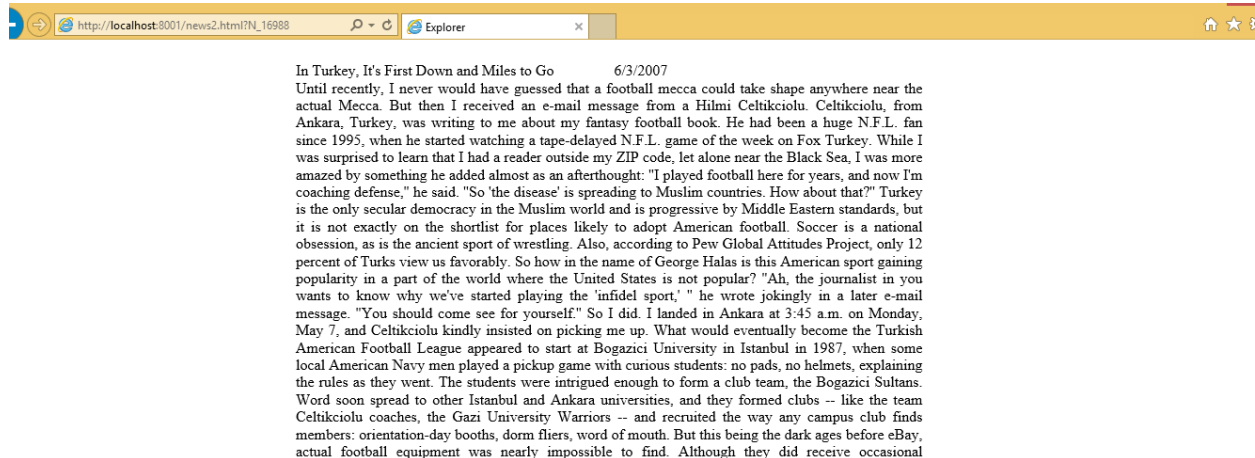


## Fetching whole news story :

When user clicks on some news story, the whole story is shown on a new page using this query :

"http://localhost:8983/solr/collection1/select?q=id%3A"+x+"&wt=json&indent=true"

where x contains the ID of news article the user has clicked on.



## Social Media Posts :

When a user clicks on "Get social Media Posts", related social media posts from facebook are displayed. The facebook's Graph API is integrated with the main application using javascript. We have used a url given by facebook's Graph API to display relevant social media posts.

After clicking Get Social Media Posts, we related social media posts :

ball into a net affixed to bamboo poles, using only their feet, chest, back or shoulders.[15] Non-competitive games included kemari in Japan, chuk-guk in Korea and woggabaliri in Australia. Association football in itself does not have a classical history [16] Notwithstanding any similarities to other ball games played around the world FIFA have recognised that no historical connection exists with any game played in antiquity outside Europe.[17] The modern rules of association football are based on the mid-19th century efforts to standardise the widely varying forms of football played in the public schools of England. The history of football in England dates back to at least the eighth century AD.[18] The Cambridge Rules, first drawn up at Cambridge University in 1848, were particularly influential in the development of subsequent codes, including association football. The Cambridge Rules were written at Trinity College, Cambridge, at a meeting attended by representatives from Eton, Harrow, Rugby, Winchester and Shrewsbury schools. They were not universally adopted. During the 1850s, many clubs unconnected to schools or universities were formed throughout the English-speaking world, to play various forms of football. Some came up with their own distinct codes of rules, most notably the Sheffield Football Club, formed by former public school pupils in 1857,[19] which led to formation of a Sheffield FA in 1867. In 1862, John Charles Thring of Uppingham School also devised an influential set of rules. The laws of the game are determined by the International Football Association Board (IFAB).[24] The Board was formed in 1886[25] after a meeting in Manchester of The Football Association, the Scottish Football Association, the Football Association of Wales, and the Irish Football Association. FIFA, the international football body, was formed in Paris in 1904 and declared that they would adhere to Laws of the Game of the Football Association.[26] The growing popularity of the international game led to the admittance of FIFA representatives to the International Football Association Board

Get social media posts

The Giants are off to a 10-0 lead on the Titans, and they're doing it via the one consistent part of their game in recent weeks. Wide receiver Odell B ...  
<http://fb.nbcports.com/e1j>

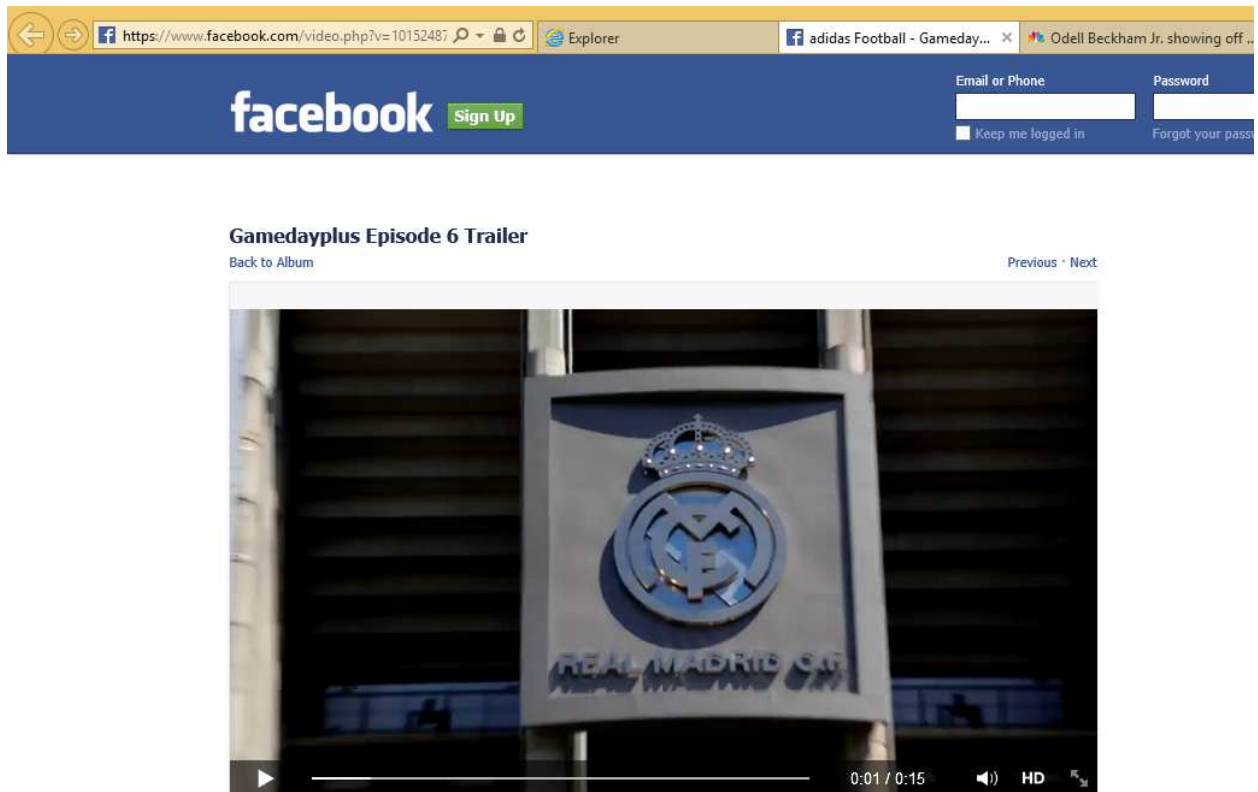
One of the day's most exciting games has been ... Jets-Vikings? Yes, Jets-Vikings. And Percy Harvin has been a major reason why. After the Jets took ...  
<http://fb.nbcports.com/nm1>

<https://www.facebook.com/video.php?v=10152487081723531>

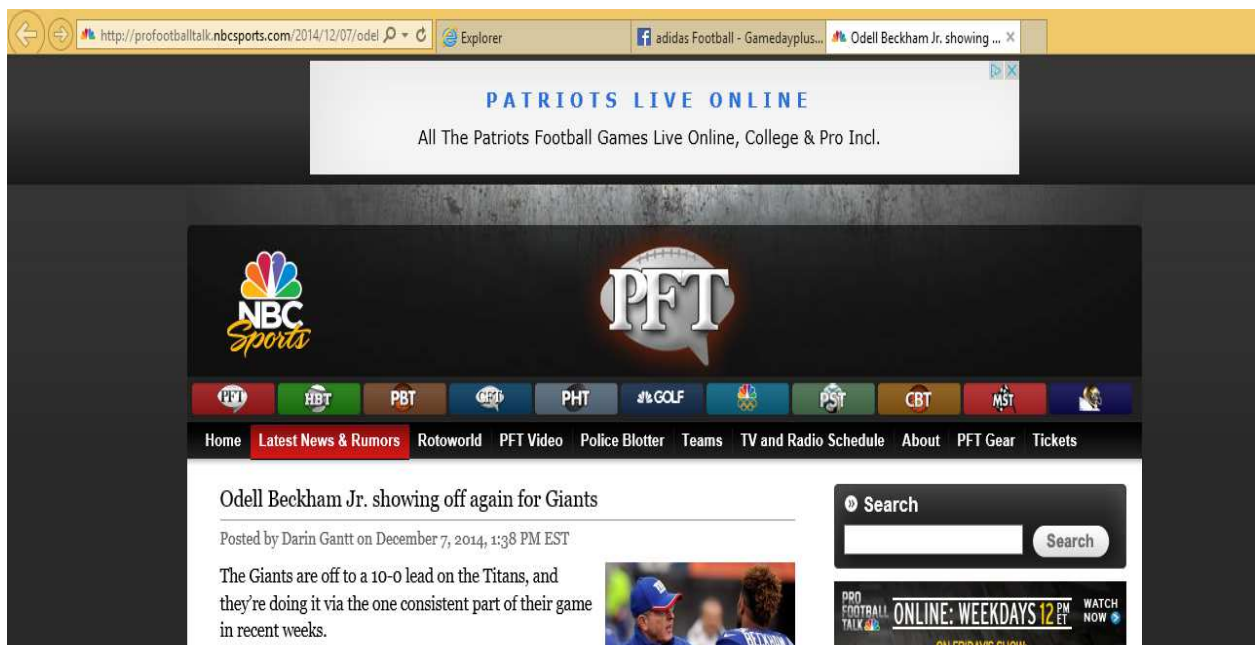
The Panthers, with one win in their last 10 games, have rocketed to a 17-0 lead against the Saints. But the bigger story comes from the brouhaha that ...  
<http://fb.nbcports.com/rGO>

Geno Smith's struggles persist. The Jets' starting quarterback was picked by Vikings linebacker Gerald Hodges on the game's first offensive play on Su ...  
<http://fb.nbcports.com/GWw>

Results of Social Media - When user clicks on a link in social media posts, it opens in a new tab :



Above shown is a social media post with YouTube video.



Above shown is a social media post with news.

## Commenting on an article :

We have added an extra feature where a user can comment on a Wiki and News article. Whenever a user comments on some article, we are creating a text file with name as Article id (W\_10 or N\_10 etc.) and storing name of the user and user's comment in that text file.

Here is the screenshot how comments are displayed in UI.

pluralism, however, was challenged in the early 1940s by a new Muslim nationalism which was demanding a separate Muslim homeland carved out of India.[6] Eventually, in August 1947, Britain granted independence, but the British Indian Empire[6] was partitioned into two dominions, a Hindu-majority India and Muslim Pakistan.[7] As many displaced Hindus, Muslims, and Sikhs made their way to their new lands, religious violence broke out, especially in the Punjab and Bengal. Eschewing the official celebration of independence in Delhi, Gandhi visited the affected areas, attempting to provide solace. In the months following, he undertook several fasts unto death to promote religious harmony. The last of these, undertaken on 12 January 1948 at age 78,[8] also had the indirect goal of pressuring India to pay out some cash assets owed to Pakistan.[8] Some Indians thought Gandhi was too accommodating.[8][9] Nathuram Godse, a Hindu nationalist, assassinated Gandhi on 30 January 1948 by firing three bullets into his chest at point-blank range.[9] Indians widely describe Gandhi as the father of the nation.[10][11] His birthday, 2 October, is commemorated as Gandhi Jayanti, a national holiday, and world-wide as the International Day of Nonviolence. He was the mentor of Indira Gandhi.

good-ankit

testing comments-sagar

sagar
testing comments
Submit

Get social media posts

<https://www.facebook.com/WomanKnowThyself/photos/a.31/?type=1>

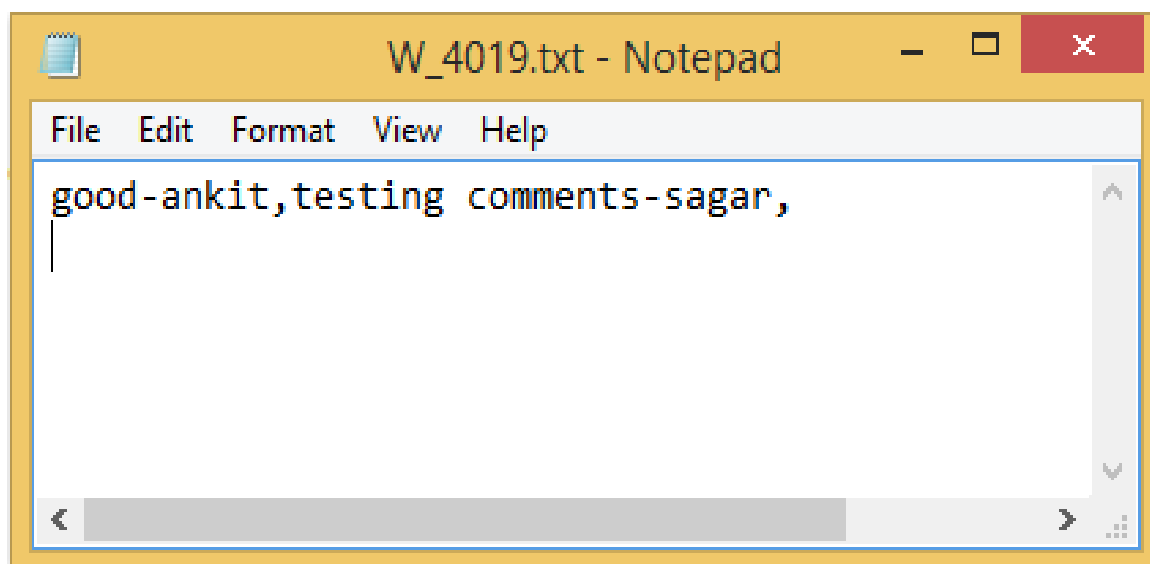
<https://www.facebook.com/ilovejesuschristpage/photos/a.1015?type=1>

<https://www.facebook.com/WomanKnowThyself/photos/a.31/?type=1>

<https://www.facebook.com/photo.php?fbid=57967222179567&set=pcb.579673682179421&type=1>

<https://www.facebook.com/photo.php?fbid=1501953750077733&set=pcb.1501954983410943&type=1>

Screenshot of text file.



# SOLR STATISTICS

## Overview:

localhost:8983/solr/#/collection1

Welcome to Facebook - Log In, Sign Up or Learn More  
https://www.facebook.com

**Apache Solr**

- Dashboard
- Logging
- Core Admin
- Java Properties
- Thread Dump
- collection1
- Overview
- Analysis
- Dataimport

**Statistics**

Last Modified: 6 minutes ago  
Num Docs: 40493  
Max Doc: 40524  
Heap Memory: 769784  
Usage:  
Deleted Docs: 31  
Version: 552  
Segment Count: 19  
Optimized:   
Current:

**Instance**

CWD: D:\IR Project\Solr2\solr-4.10.2\example  
Instance: D:\IR Project\Solr2\solr-4.10.2\example\solr\collection1  
Data: D:\IR Project\Solr2\solr-4.10.2\example\solr\collection1\data  
Index: D:\IR Project\Solr2\solr-4.10.2\example\solr\collection1\data\index  
Impl: org.apache.solr.core.NRTCachingDirectoryFactory

**Replication (Master)**

	Version	Gen	Size
Master (Searching)	1417974970353	151	289.22 MB
Master (Replicable)	1417974970353	151	-

**Healthcheck**

Ping request handler is not configured with a healthcheck file.

## Select Query: (Average Response Time):

**/select**

class: org.apache.solr.handler.component.SearchHandler  
version: 4.10.2  
description: Search using components:  
query  
facet  
mlt  
highlight  
stats  
expand  
spellcheck  
debug  
src: null

---

stats:

handlerStart:	1417974918138
requests:	17
errors:	2
timeouts:	0
totalTime:	2963.92364
avgRequestsPerSecond:	0.002769416150385179
5minRateReqsPerSecond:	0.010011965810882713
15minRateReqsPerSecond:	0.007562148034267018
avgTimePerRequest:	174.3484494117647
medianRequestTime:	0.882636
75thPcRequestTime:	3.1115285000000004
95thPcRequestTime:	2242.707892
99thPcRequestTime:	2242.707892

## Suggest Query (Average Response Time):

### /suggest

class:	org.apache.solr.handler.component.SearchHandler	
version:	4.10.2	
description:	Search using components: suggest	
src:	null	
stats:		
handlerStart:	1417974918169	
requests:	3	
errors:	0	
timeouts:	0	
totalTime:	10.110444	
avgRequestsPerSecond:	0.0004887219241273287	
5minRateReqsPerSecond:	3.984135194187845e-7	
15minRateReqsPerSecond:	0.0001085394252025552	
avgTimePerRequest:	3.370148	
medianRequestTime:	2.278432	
75thPcRequestTime:	6.755559	
95thPcRequestTime:	6.755559	
99thPcRequestTime:	6.755559	
999thPcRequestTime:	6.755559	

## More Like This Query (Average Response Time):

### /mlt

class:	org.apache.solr.handler.MoreLikeThisHandler	
version:	4.10.2	
description:	Solr MoreLikeThis	
src:	null	
docs:	<a href="http://wiki.apache.org/solr/MoreLikeThis">http://wiki.apache.org/solr/MoreLikeThis</a>	
stats:		
handlerStart:	1417974918169	
requests:	12	
errors:	2	
timeouts:	0	
totalTime:	4375.096562	
avgRequestsPerSecond:	0.001954890730038597	
5minRateReqsPerSecond:	0.008547468930994586	
15minRateReqsPerSecond:	0.006246740339672311	
avgTimePerRequest:	364.5913801666667	
medianRequestTime:	186.84446250000002	
75thPcRequestTime:	443.45120475	
95thPcRequestTime:	1737.848855	
99thPcRequestTime:	1737.848855	
999thPcRequestTime:	1737.848855	



## Hits (Query Hits):

### queryResultCache

class:	org.apache.solr.search.LRUCache
version:	1.0
description:	LRU Cache(maxSize=512, initialSize=512)
src:	null

stats:	lookups:	123
	hits:	30
	hitratio:	0.24
	inserts:	93
	evictions:	0
	size:	93
	warmupTime:	0
	cumulative_lookups:	123
	cumulative_hits:	30
	cumulative_hitratio:	0.24
	cumulative_inserts:	93
	cumulative_evictions:	0

## Hits (Document Hits):

### documentCache

class:	org.apache.solr.search.LRUCache
version:	1.0
description:	LRU Cache(maxSize=512, initialSize=512)
src:	null

stats:	lookups:	307
	hits:	121
	hitratio:	0.39
	inserts:	186
	evictions:	0
	size:	186
	warmupTime:	0
	cumulative_lookups:	307
	cumulative_hits:	121
	cumulative_hitratio:	0.39
	cumulative_inserts:	186
	cumulative_evictions:	0

## Warm up Time:

### searcher

class:	org.apache.solr.search.SolrIndexSearcher	
version:	1.0	
description:	index searcher	
src:	null	
stats:	searcherName:	Searcher@550cbff8[collection1] main
	caching:	true
	numDocs:	40493
	maxDoc:	40524
	deletedDocs:	31
	reader:	StandardDirectoryReader(segments_46:552:nrt_3t(4.10.2):C32/1:delGen=1_1v(4.10.2):C80_43(4.10.2):C59_2m(4.10.2):C275_3a(4.10.2):C9613/30:delGen=1_3b(4.10.2):C4911_3c(4.10.2):C5422_3d(4.10.2):C3998_3e(4.10.2):C8563_3f(4.10.2):C4324_3g(4.10.2):C3123_3x(4.10.2):C13_44(4.10.2):C6_45(4.10.2):C10_46(4.10.2):C10_47(4.10.2):C4_48(4.10.2):C3_49(4.10.2):C3_4a(4.10.2):C75)
	readerDir:	org.apache.lucene.store.NRTCachingDirectory:NRTCachingDirectory(MMapDirectory@D:\IR Project\Solr2\solr-4.10.2\example\solr\collection1\data\index lockFactory=NativeFSLockFactory@D:\IR Project\Solr2\solr-4.10.2\example\solr\collection1\data\index; maxCacheMB=48.0 maxMergeSizeMB=4.0)
	indexVersion:	552
	openedAt:	2014-12-07T17:56:11.649Z
	registeredAt:	2014-12-07T17:56:13.027Z
	warmupTime:	1

## Core:

CACHE

CORE

HIGHLIGHTING

OTHER

QUERYHANDLER

QUERYPARSER

UPDATEHANDLER

Watch Changes

Refresh Values

Searcher@550cbff8[collection1] main

core

class: collection1

version: 1.0

description: SolrCore

src: null

stats:

coreName: collection1

startTime: 2014-12-07T17:55:17.935Z

refCount: 2

instanceDir: D:\IR Project\Solr2\solr-4.10.2\example\solr\collection1\

indexDir: D:\IR Project\Solr2\solr-4.10.2\example\solr\collection1\data\index/

aliases: collection1

searcher



## **Future Work :**

- In future we intend to implement rating also in the same way, we have implemented comments.
- Currently we are only using Graph API for facebook, we intended to integrate twitter API also which we were not able to do in this project due to some integration issues. We also researched on Quora API and discovered Quora API is not available due to lack of consumer base.
- Right Now suggest is only working in solr, we intend to implement it in UI also.
- In future we will also try to categorize documents so that user can see his choice of articles according to category using Faceting.

## **Member Contribution :**

GUI design: Sandeep and Sagar

Hosting local website using IIS manager: Sagar

Query Parser : Sagar

Snippet formation : Sagar

Integration of Solr data, features in GUI : Sagar

Navigation from one page to another : Sagar

Research into APIs : Sandeep and Teja

Integrating API with main application : Sagar, Sandeep and Teja

Commenting on articles : Sandeep and Sagar

Parsing Wikipedia and News Articles : Ankit and Teja

Indexing in Solr : Ankit and Teja

Solr features : Ankit