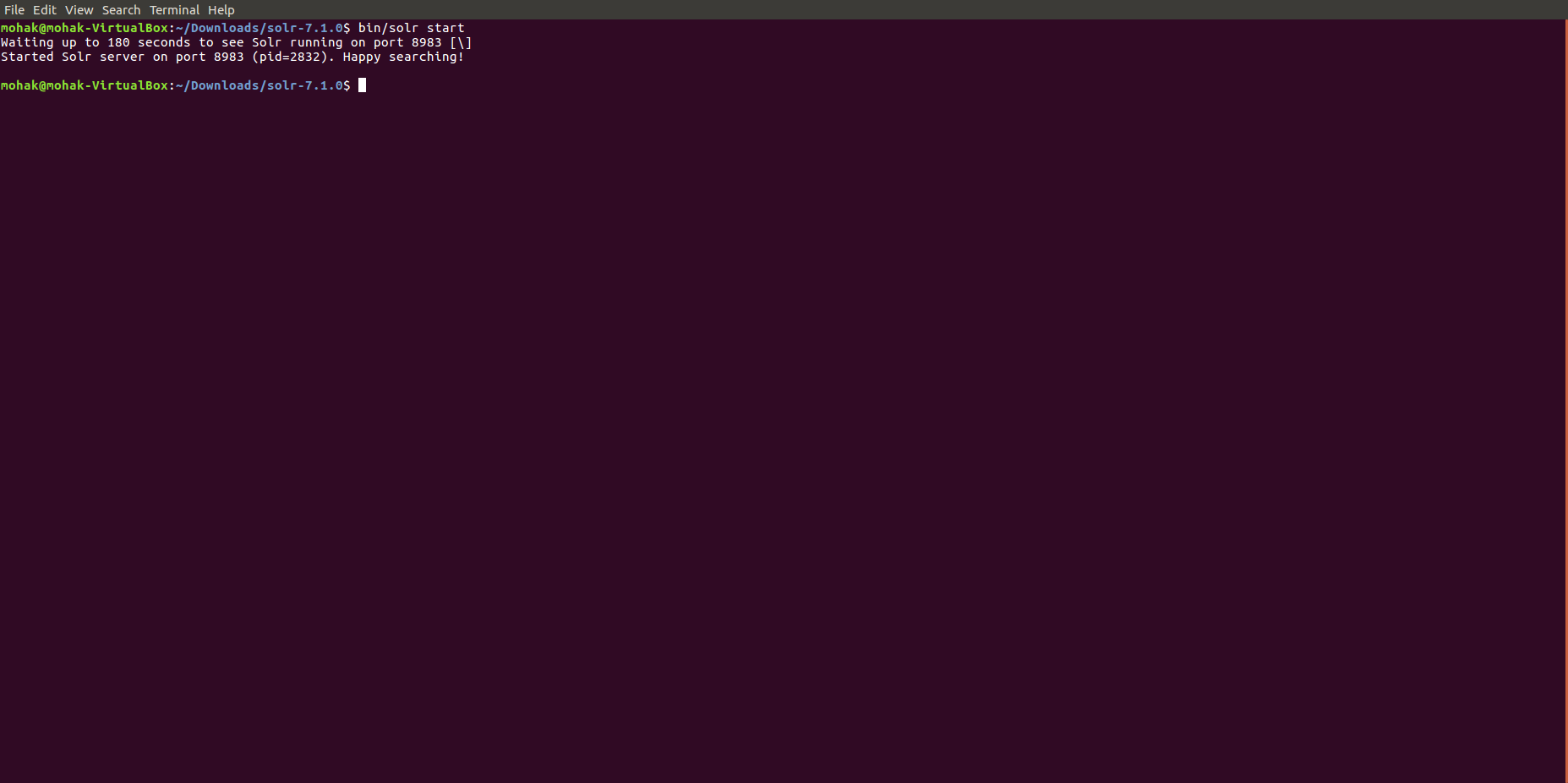
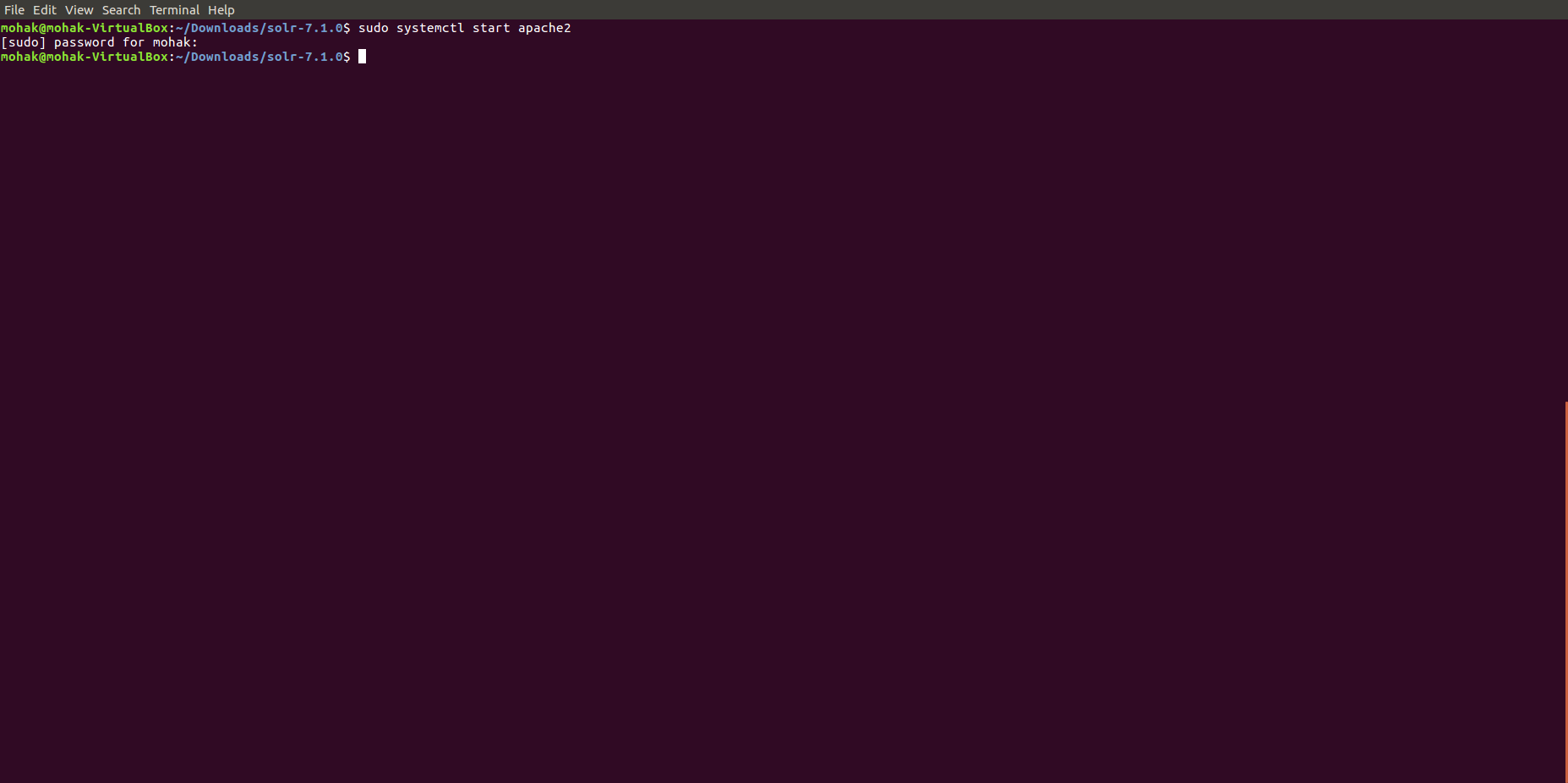
1. **Steps Taken to Complete the assignment:**
2. Install JAVA in Linux.
3. Install Solr in linux.
4. Installed Apache Server and PHP.
5. Start the server and create ‘csci572 cluster for use.
6. Make changes in “Managed-Schema” file corresponding to the cluster ‘csci572’ as per HW instructions.
7. Make the changes in the solrconfig.xml to incorporate \_text\_ searches as default.
8. Post the Data File corresponding to NYDailyNews into the cluster of indexing by following command:

“bin/post -c csci572 -filetype html NYD/NYD/”

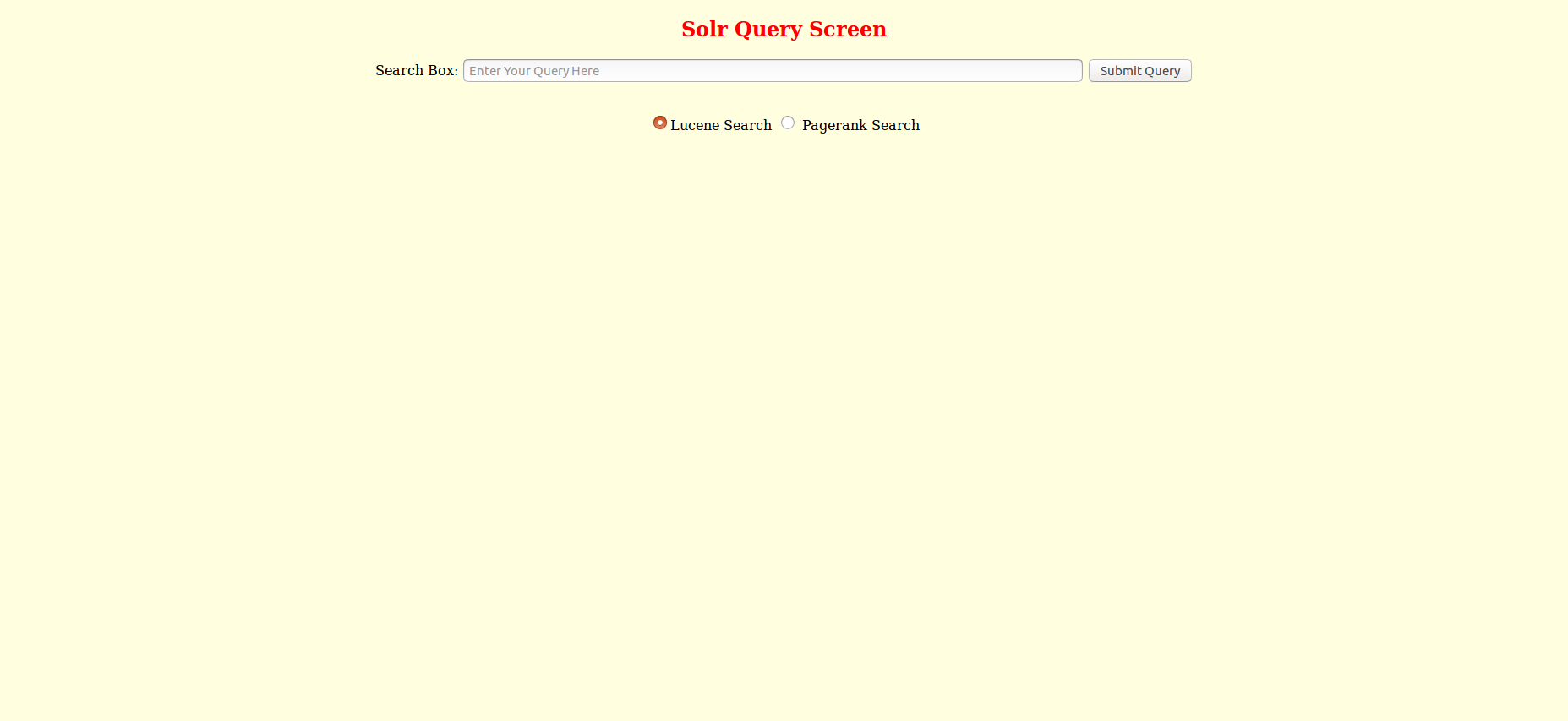
1. Created a program CreateMap.java which takes in input the “NYD Map.csv” as input and returns two Hash Map Data Structures. The function “GetURLMap” returns a hash map which has DocId as key and HTML file link as the value whereas the function ‘GetFileURL” returns a hash map which has HTML file link as key and DocId is the value.
2. Created a program ExtractingLinks which uses JSoup to find all the links in the HTML file and give it to us. The program takes in the input the entire Data File which has all the download HTML files and output a “edgeList.txt” file which has list of all incoming and outgoing links to the HTML file from all the HTML pages present in the Data File.
3. Then used this “edgeList.txt” file to create the network Graph using the python program “GraphMap” outputting the file “external\_pageRankFile. This “external\_pageRankFile” has the page ranking of all the HTML files that are present in the Data File using networkx python library.
4. Added fieldType and field into the Managed-Schema file to incorporate PageRank algorithm based search
5. Added listeners in the solrconfig.xml file of the cluster to allow solr to access the “external\_pageRankFIle” for the search purpose. The file is saved in the data folder of the cluster ‘csci572’.
6. Created a PHP page called solrquery.php which run on top of Apache Server. This php page uses Apache-solr-php client to access solr and based on query type i.e either Lucene based or Page Rank based, returns results of the query showing 10 at a time on the page.
7. Queried of all the 8 queries as per HW instructions and compared similarity to create an overlap graph based on the number of URL’s which were common among the two results.
8. Grabbed the screenshot of each results as per HW instructons.
9. **Flow of the Search Page Based on 1 query.**
10. Start Solr Server



1. Start Apcahe Server



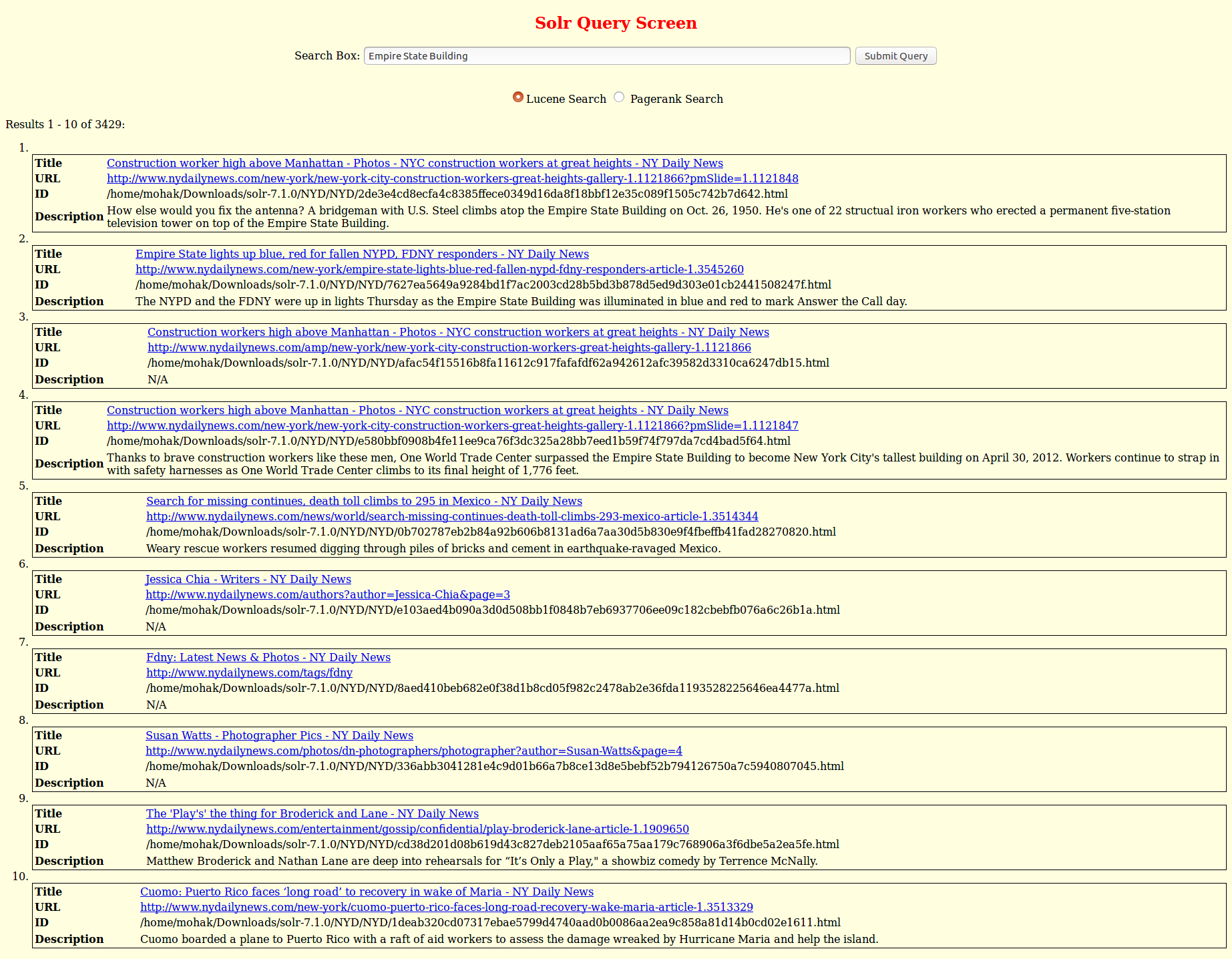
1. Open Query Page by going to “http://localhost/solrquery.php



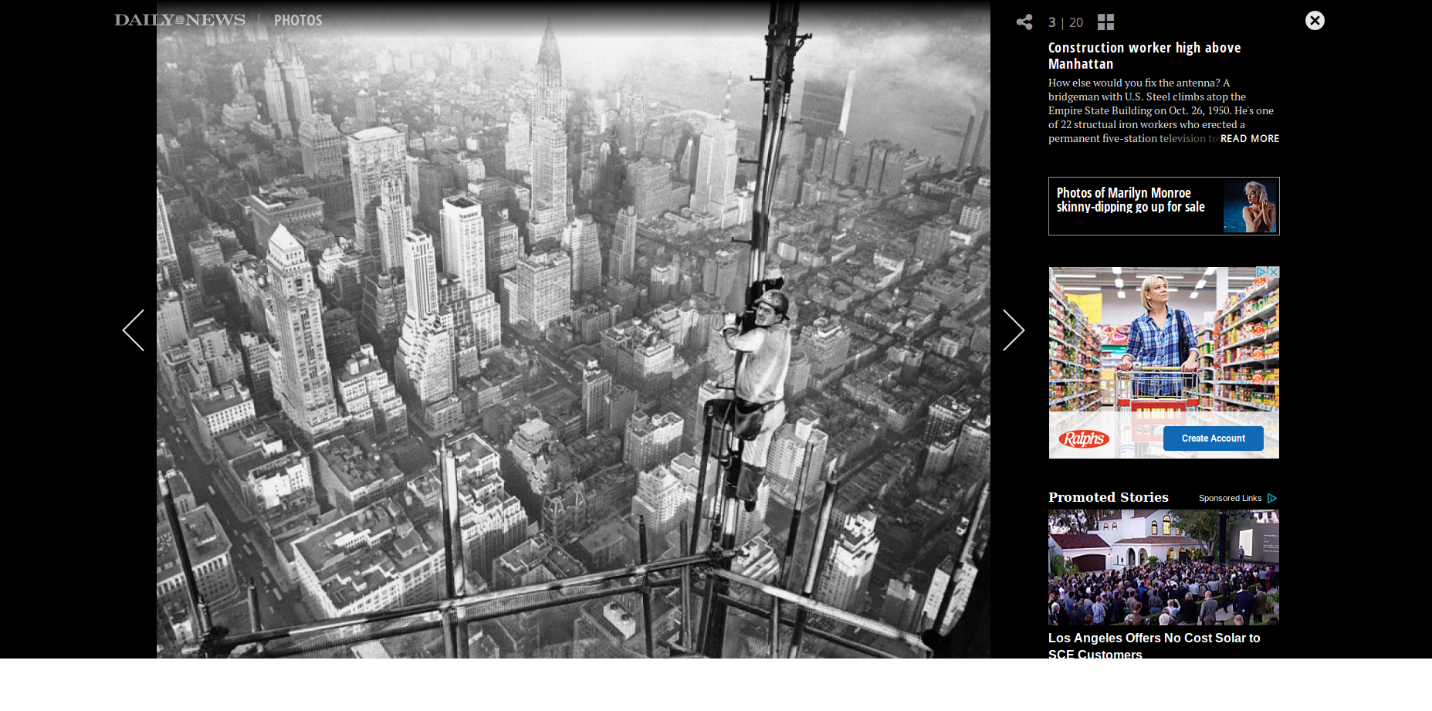
1. Enter the Query



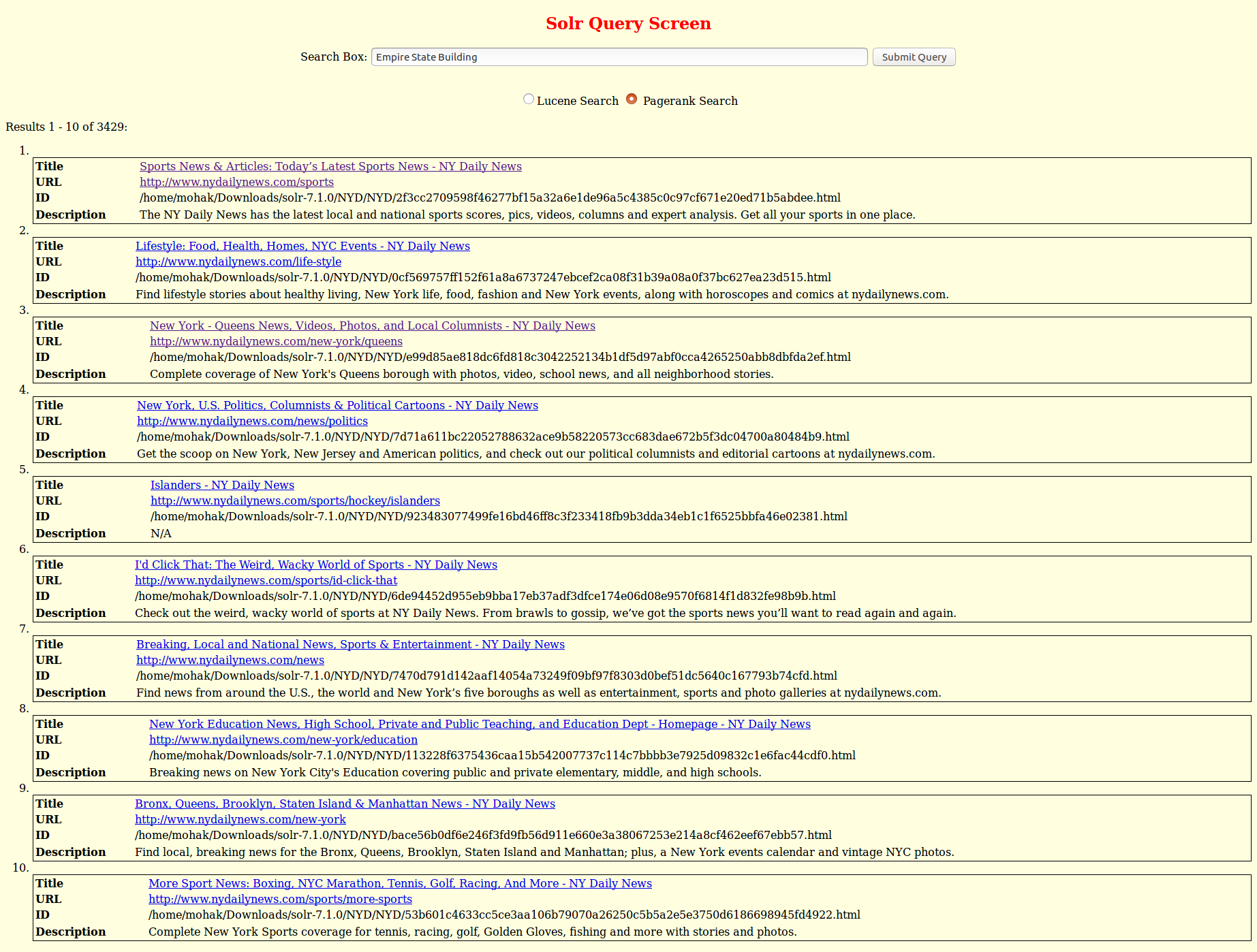
1. Lucene Query Result

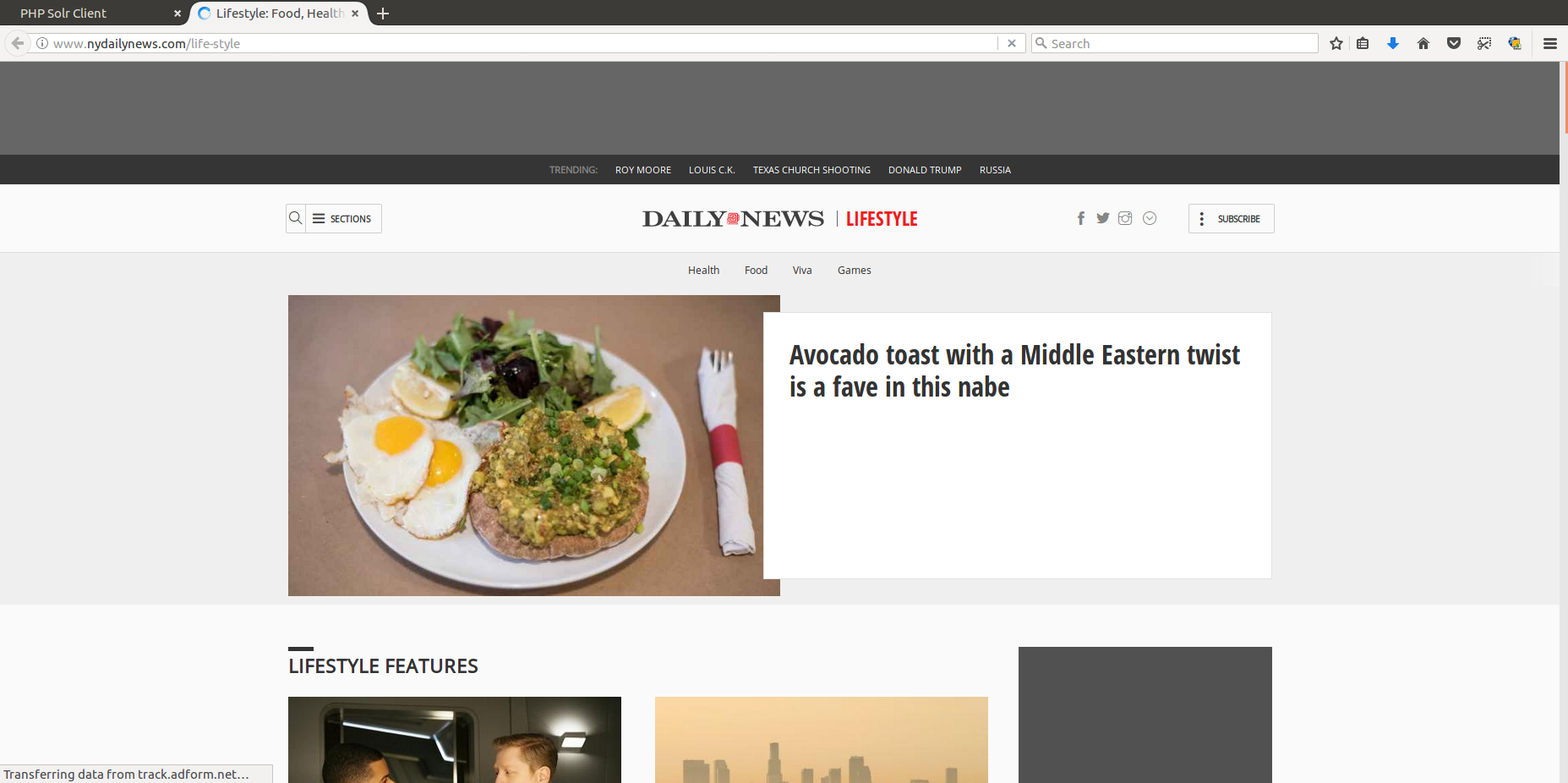


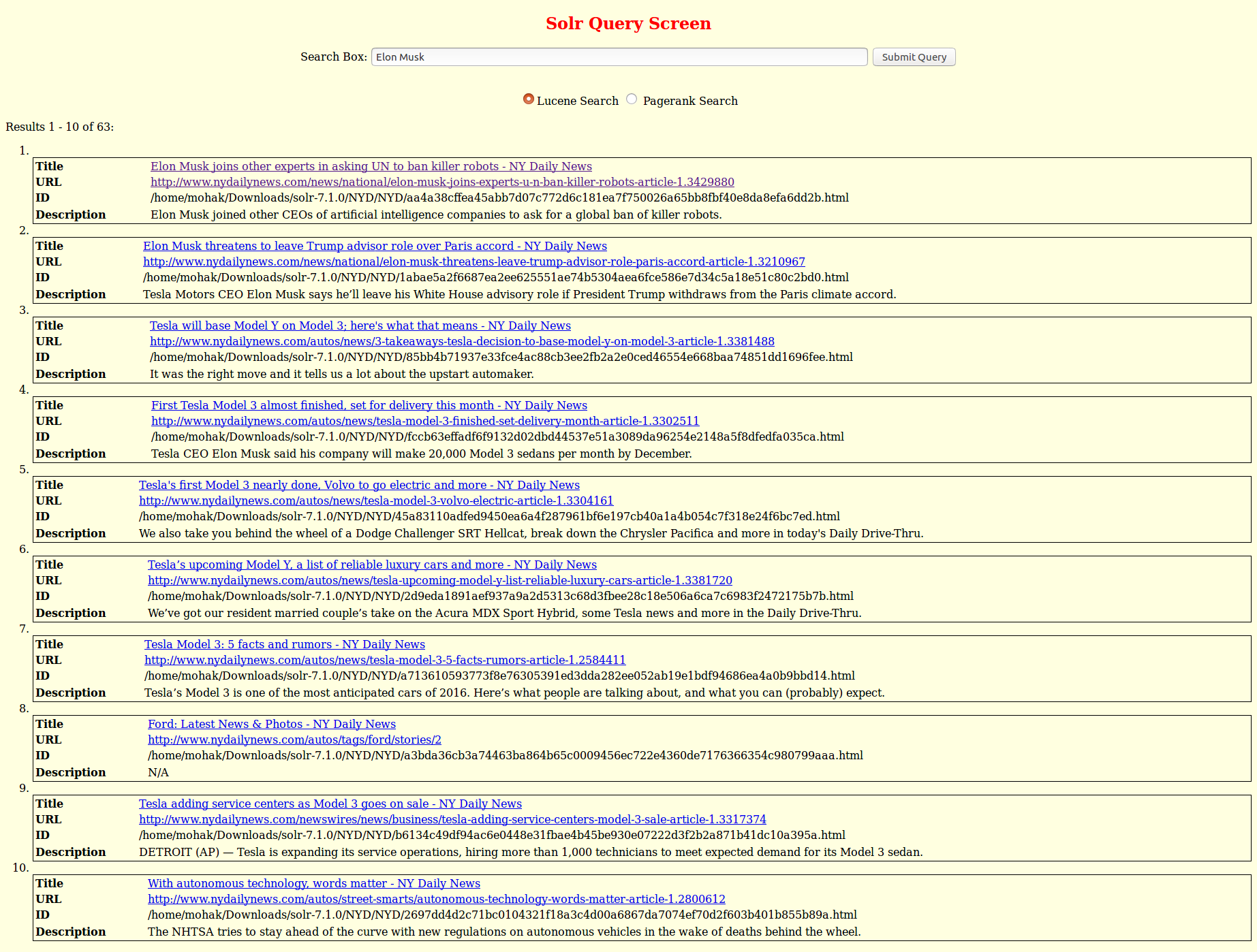
1. Result of First URL Clicked

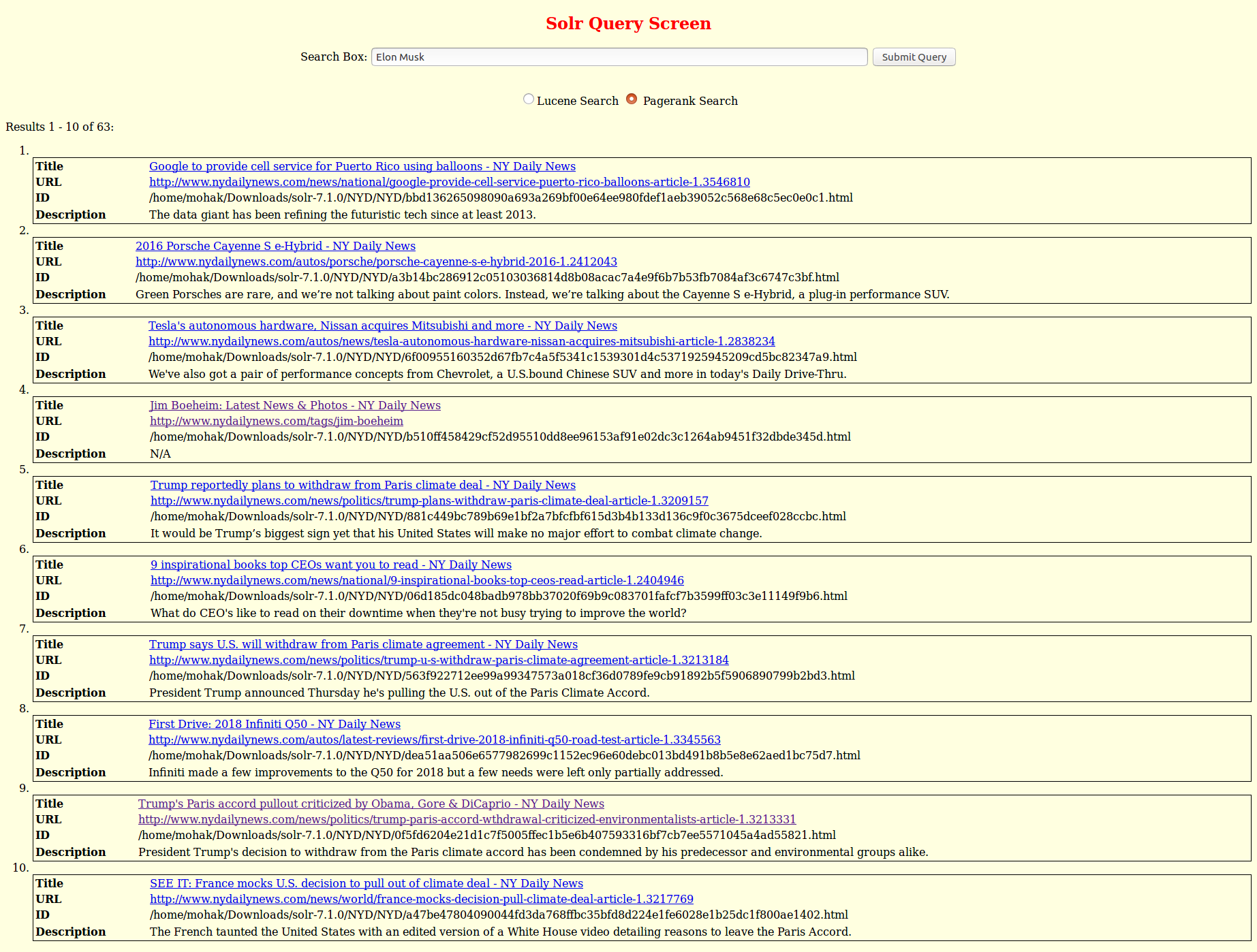


1. PageRank result of the query

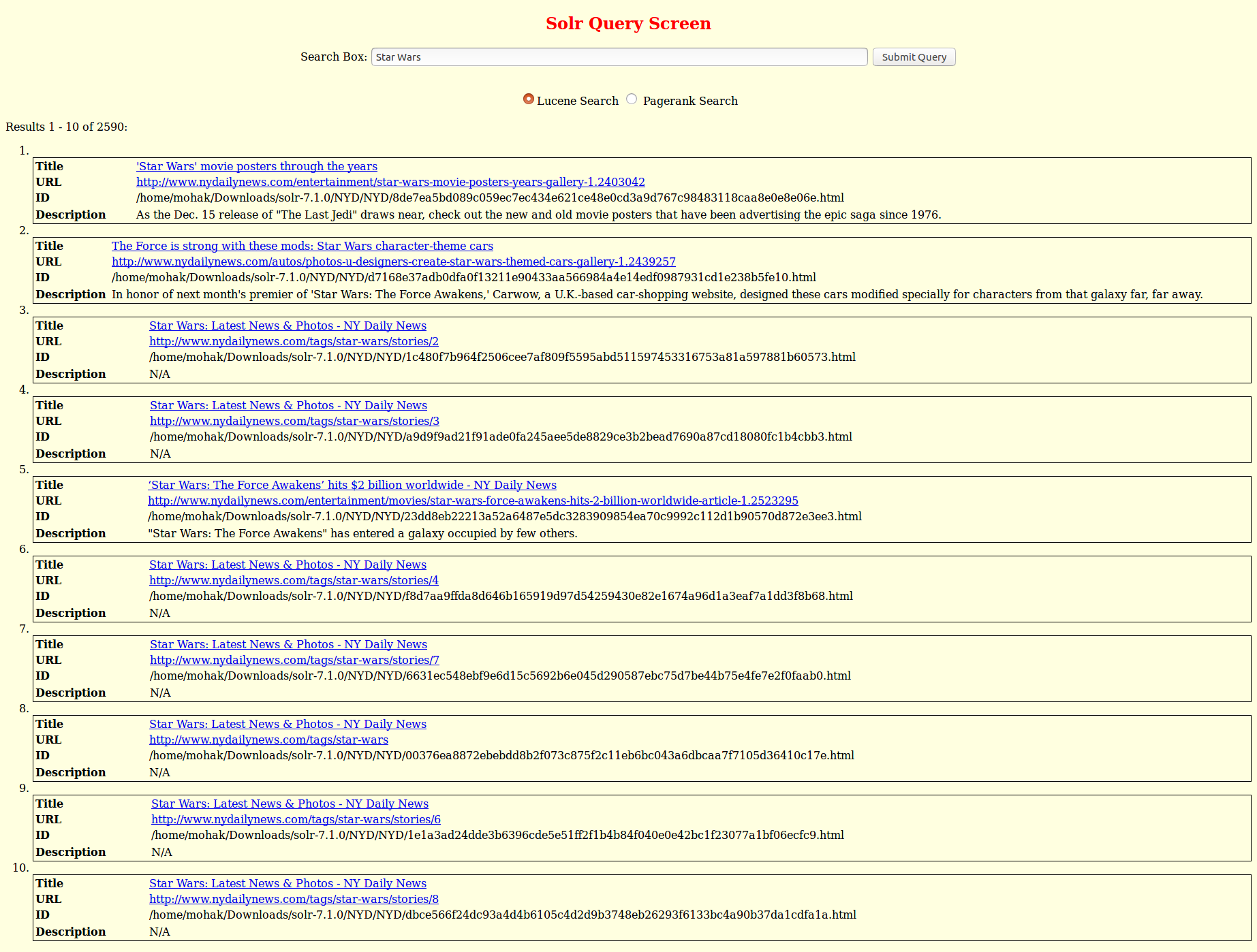


1. Result of 2nd URL clicked
2. **Results of the 8 queries.** First Screenshot will be of Lucene Result and 2nd will be of the pagerank result.
3. Elon Musk





1. Star Wars





1. North Korea



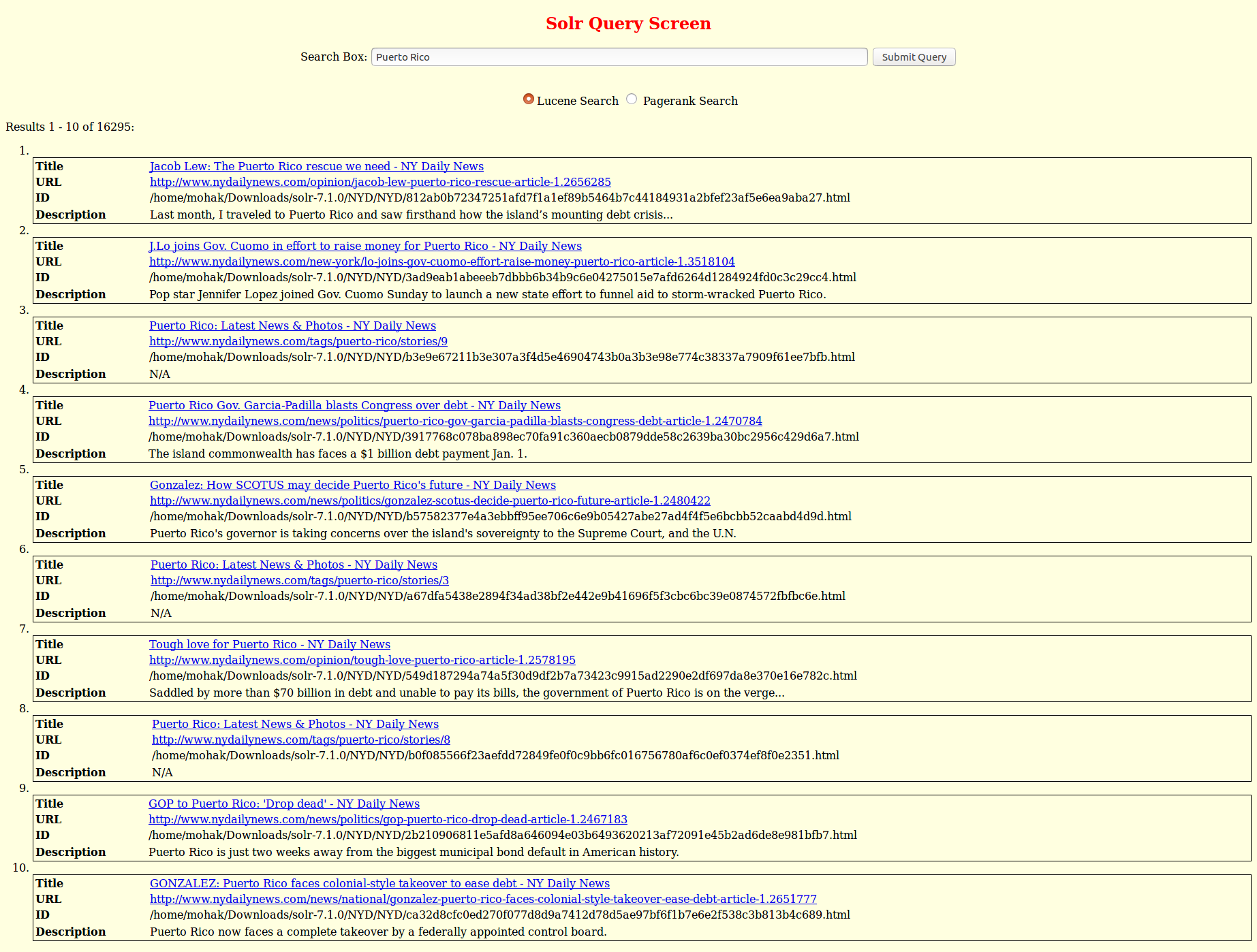


1. LA Dodgers



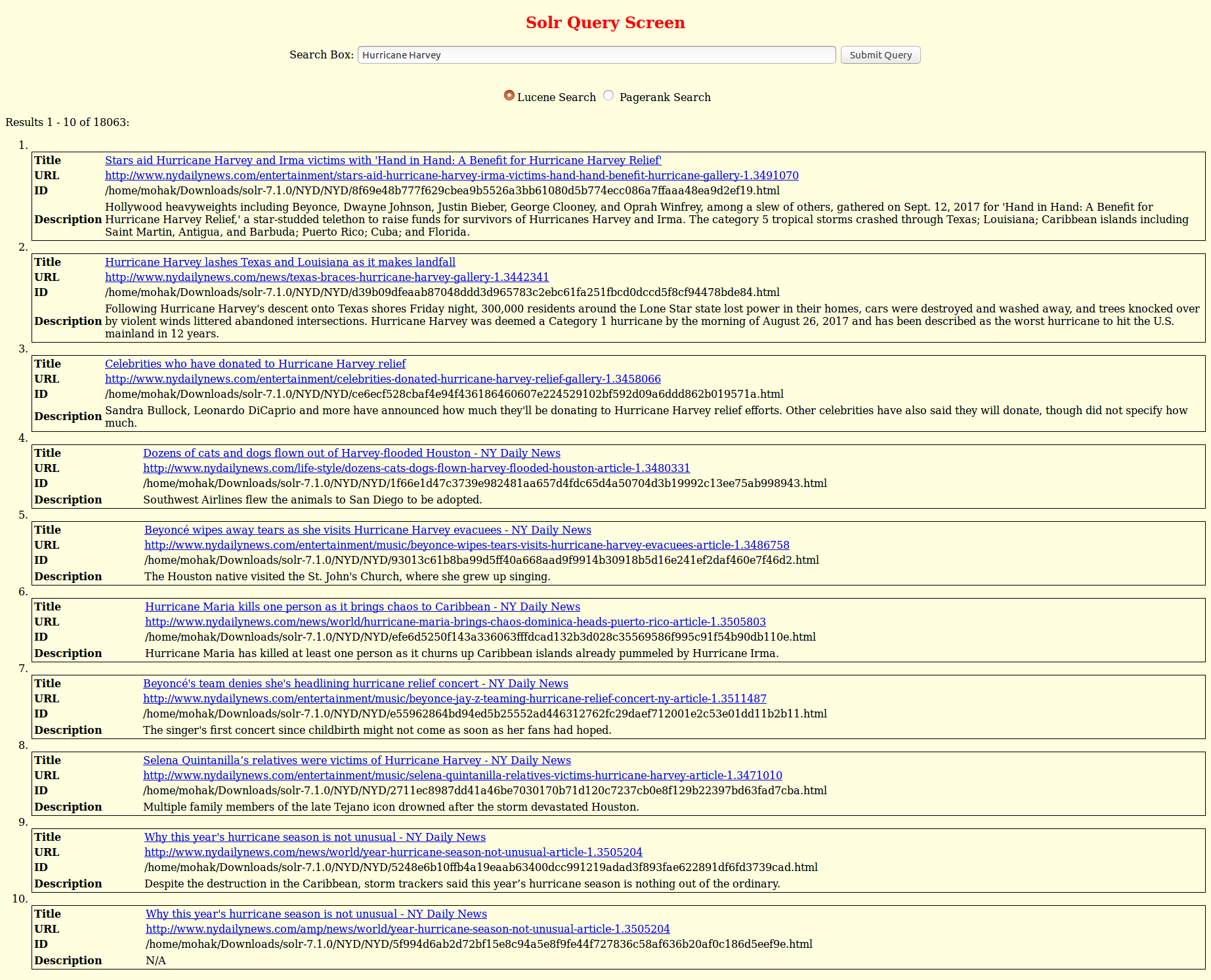


1. Puerto Rico



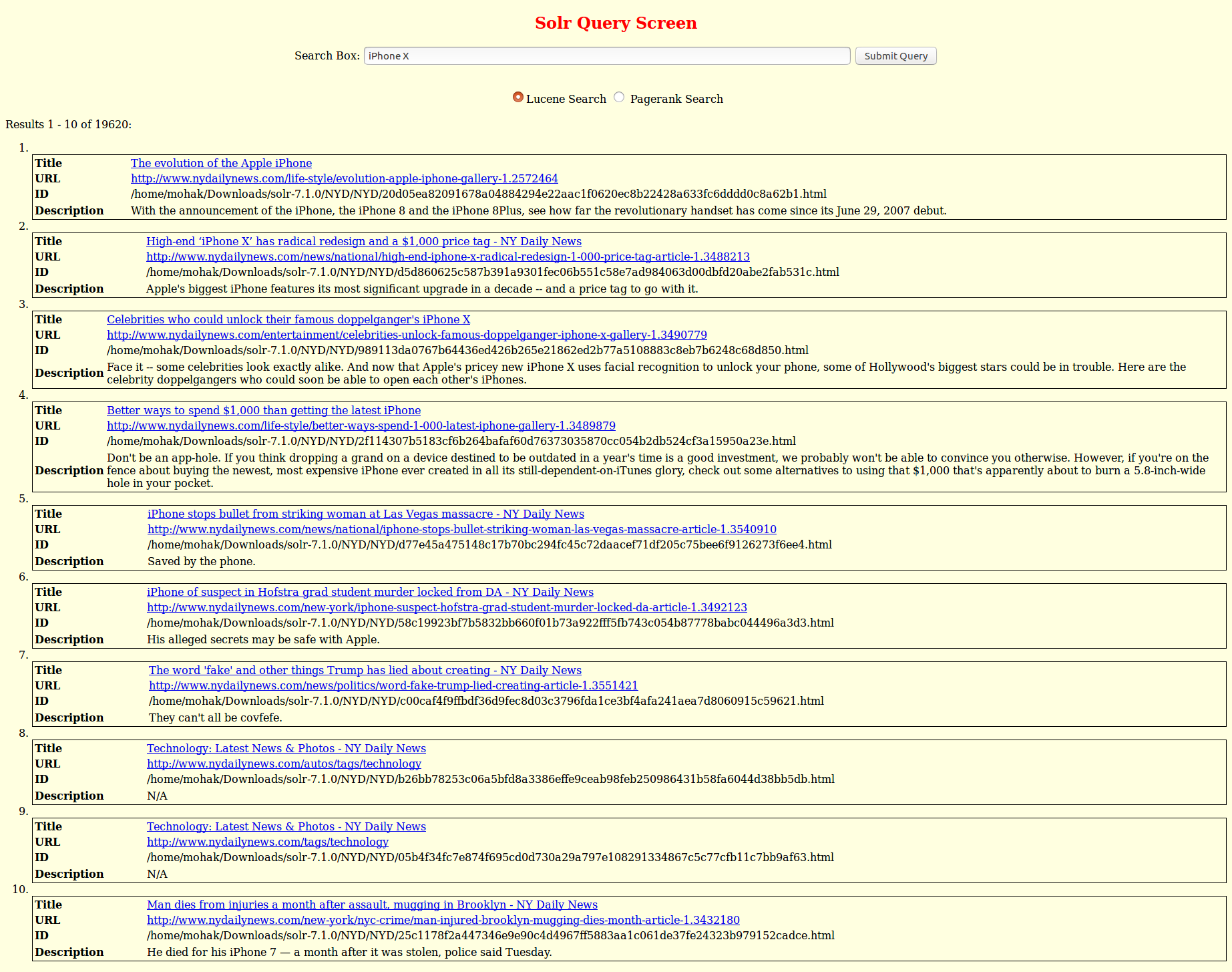


1. Hurricane Harvey



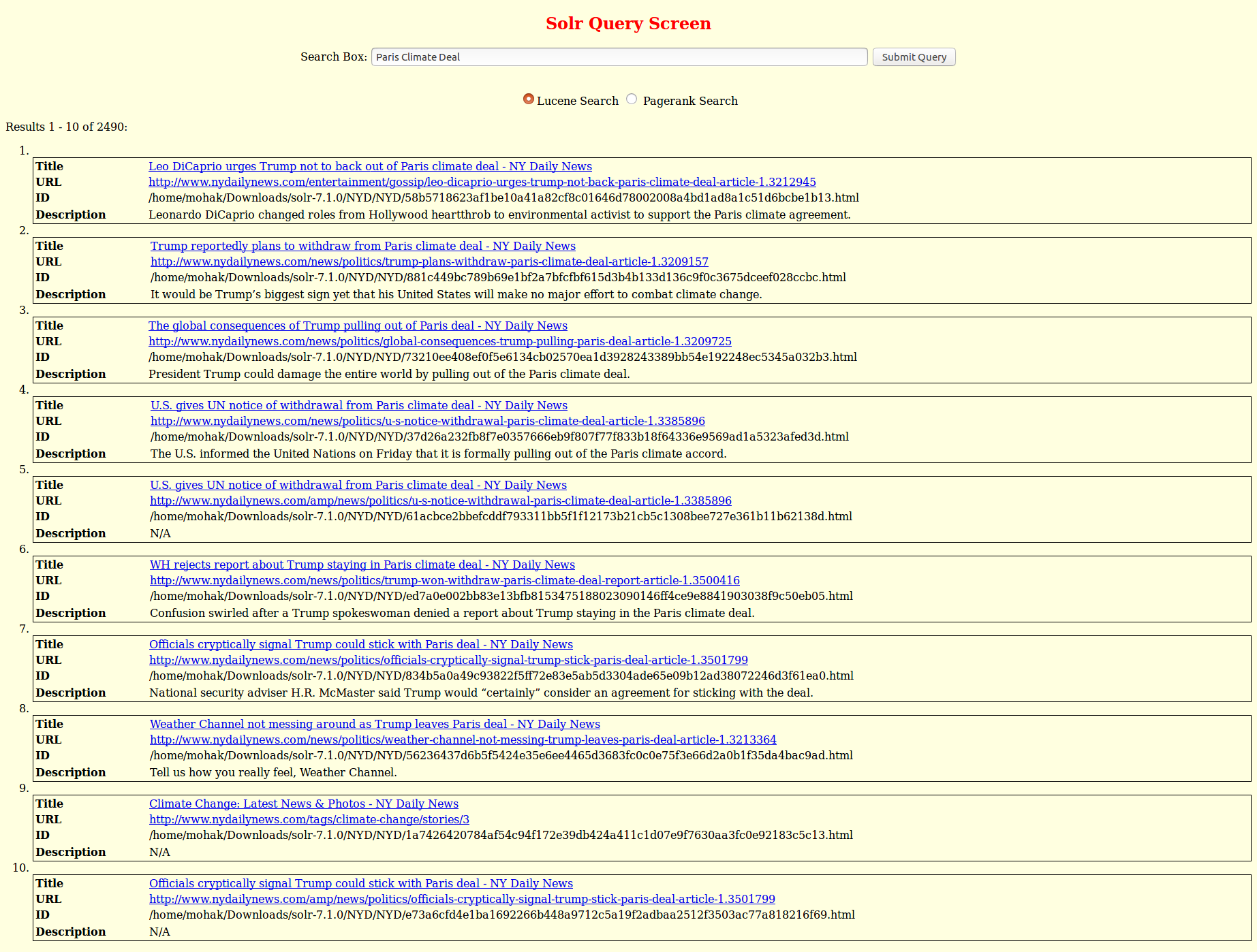


1. iPhone X





1. Paris Climate Deal





1. **Why PageRank of webpages vary?**

The Page Rank of a page depends on page rank value of all the pages that are pointed to this page. The page rank of a page may be lower if the page pointing to this page has a lower page rank score and with the effect of damping factor the page rank of the page can decrease. Thus, if a page has multiple pages pointing to this page, then the page will have a high Page Rank numbers and if it has less number or zero pages pointing to it, then, its value will be less.

1. **Overlap Graphs**

The results produced by the Lucene and Page Rank results for the 8 queries mentioned didn’t had any URL’s in common from NYDaily and hence the overlap graph is empty.

|  |  |
| --- | --- |
| Query | No. of URL Common |
| Elon Musk | 0 |
| Star Wars | 0 |
| North Korea | 0 |
| LA Dodgers | 0 |
| Puerto Rico | 0 |
| Hurricane Harvey | 0 |
| iPhone X | 0 |
| Paris Climate Deal | 0 |