Search Engine with AIML Bot

Assistant Professor Mrs. Bhagyashri Wakde
UG students Aakriti Upadhyay, Akriti A, Ansar Shaik, Syed Emaddudin Hussain

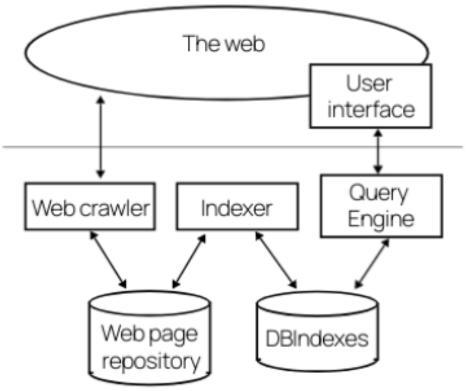
Department of Computer Science and Engineering
Rajiv Gandhi Institute of Technology
Bangalore-560032, Karnataka

Abstract: Our project is based on optimizing search engine results to provide relevant information to the user along with ad-free browsing experience. Main aim of this project is to develop a bot which will search in many different search engines and display top results which are more useful to the user. Later on, these results are stored in the database for further processing using Machine Learning. In the present trend, search engines are used by everyone for finding the required information on the web. Google is one of the most used search engines followed by Yahoo and Bing. For every search engine, the first few results are the sponsored results, where the website ranks because it is a paid advertise. This reduces the visibility of most organic websites with correct, accurate and more relevant information. So in this project, we collect results and display it without ads, for a better experience. Moreover, it will filter out the various irrelevant content and display only what is required. In addition to blocking ad tracking, it safeguards users from cross-site cookie tracking, bounce tracking, fingerprinting, phishing, and malware attacks. This paper contains everything we have learnt in the process.

INTRODUCTION

Nowadays, every user who uses the Internet wants to search for anything like educational colleges, about Information Technology, books, news etc., using Search Engines. A search engine is a software system designed to carry out web searches. They search the World Wide Web in a systematic way for particular information specified in a textual web search query. The search results are generally presented in a line of results, often referred to as search engine results pages (SERPs). When a user enters a query into a search engine, the engine scans its index of web pages to find those that are relevant to the user's query. The results are then ranked by relevance and displayed to the user. The information may be a mix of links to web pages, images, videos, infographics, articles, research papers, and other types of files. Some search engines also mine data available in databases or open directories. Unlike web directories and social bookmarking sites, which are maintained by human editors, search engines also maintain real-time information by running an algorithm on a web crawler. Any internet-based content that cannot be indexed and searched by a web search engine fall under the category of deep web. This is needed for everyone. The project is a search engine, which searches many search engines and selects the top links and then goes on those links. To develop this project, we applied a ranking algorithm to give better results to the user. We collect results and display it without ads, for a better experience. Moreover, it will filter out the various irrelevant content and display only what is required. This paper contains everything we have learnt in the process.

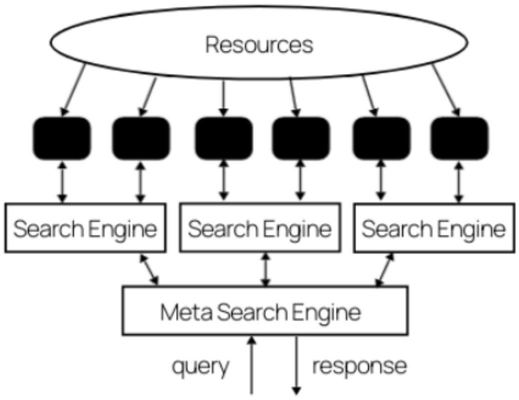
EXISTING ARCHITECTURE



General Search Engines

Search engine work through three primary functions:

Crawling: Scour the Internet for content, looking over the code/content for each URL they find.
Indexing: Store and organize the content found during the crawling process. Once a page is in the index, it’s in the running phase to be displayed as a result to relevant queries.
Ranking: Provide the pieces of content that will best answer a searcher's query, which means that results are ordered by most relevant to least relevant.
Crawling is the discovery process in which search engines send out a team of robots (known as crawlers or spiders) to find new and updated content. Content can vary — it could be a webpage, an image, a video, a PDF, etc. — but regardless of the format, content is discovered by links.
Googlebot starts out by fetching a few web pages, and then follows the links on those webpages to find new URLs. By hopping along this path of links, the crawler is able to find new content and add it to their index called Caffeine — a massive database of discovered URLs — to later be retrieved when a searcher is seeking information that the content on that URL is a good match for.



Meta Search Engine

In general terms, metasearch engines aggregate results from a variety of search engines. You can find meta search engines that compare prices on everything from clothing to electronics to airfare. Let’s say you’re in the market for a new set of Marshall’s noise-cancelling headphones. You could manually shop around to find the best price, or you could use a metasearch site like Google Shopping, which would show prices and links to buy Bose headphones on retail websites like Amazon, Best Buy, and Bose.com. You would then complete your purchase on one of those linked websites. Meta search engine gives more priority to the sponsored ads which are displayed at the top. As a normal psychological behaviour, people tend to click them first. This reduces the visibility of organic results. This is the main working of a general meta search engine. Our project intends to incorporates this search engine and optimize it for a better user experience.

PROPOSED ARCHITECTURE

Our main objective is to provide ad free browsing experience. From our research, we realized that there were few issues in the meta search engine as it would give priority to sponsored search results. Our search engine automatically blocks ads and ad trackers to protect users from annoying and potentially malicious ads. In addition to blocking ad tracking, it safeguards users from cross-site cookie tracking, bounce tracking, fingerprinting, phishing, and malware attacks.

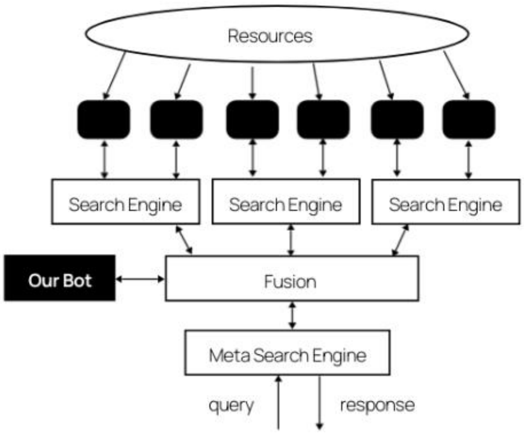
Speciality of our web search engine:

- Private browsing mode
- Block third party cookies by default
- Blocks social trackers

Features of our search engine include no ads or sponsored content, no cookies, no tracking or linking your personal IP address, no URL tracking tags, autocomplete search engine suggestions, provide light/dark system theme modes and provide optional location based searching.

The working of our search engine is the same as that of the meta search engine. However, we are implementing a bot that will filter the content and the search results, and provide only the relevant information.

The crawlers from our engine will scrape content and present it. Then comes the fusion phase where the results will be presented together and ranked based on the website's relevancy score. This relevancy score will be provided from out bot, using ML. The relevance score is calculated based on metrics like how many times the website is clicked etc. There are 3 modules in this engine. They are filters, search and storage. Filters are used to remove the items on the blacklist.txt from the pages. This is the main operation of our bot. It removes the ads using ML.



TECHNOLOGIES USED

The technologies we used for this project are:

- Flask framework for the web interface
- Python is used to code the backend.
- Panda library from python is used. Pandas is a software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series. Since we will be dealing with a lot of data and information, we use pandas for better manipulation.
- Requests module is also used. Requests is a HTTP library for the Python programming language. The goal of the project is to make HTTP requests simpler and more human-friendly. Since the crawlers will extract information from websites through URL, and process a lot of URLs, we require Requests module.
- Beautiful Soup is a Python library that is used for web scraping purposes to pull the data out of HTML and XML files. It creates a parse tree from page source code that can be used to extract data in a hierarchical and more readable manner. Web Scraping is the process of collecting data from the internet by using various tools and frameworks. Sometimes, it is used for online price change monitoring, price comparison, and seeing how well the competitors are doing by extracting data from their websites.

CONCLUSION

Search engines offer users vast and impressive amounts of information, available with a speed and convenience few people could have imagined one decade ago. Their capabilities are expanding practically by the day. Soon it will seem routine to be able to search the contents of vast libraries of books; to find selected portions of video streams or audio recordings; to benefit from personalized searches that remember a user's preferences and keep track of changing geographical locations. Audio searching and search results will be available for the blind;

"implicit searching" will anticipate users' queries and have answers ready.

Today's internet users are very positive about what search engines already do, and they feel good about their experiences when searching the internet. They say they are comfortable and confident as searchers and are satisfied with the results they find. They trust search engines to be fair and unbiased in returning results. And yet, people know little about how engines operate, or about the financial tensions that play into how engines perform their searches and how they present their search results. Furthermore, searchers largely don't notice or understand or discern the different kinds of search results that are being served up to them.

This odd situation, in which a growing population of users relies on technology most of them don't understand, highlights the responsibility placed on search engine companies. They are businesses, in many cases extremely successful ones – but their effects on society are far more than merely commercial. One unexpected implication of our study is that search engines are attaining the status of other institutions – legal, medical, educational, governmental, journalistic – whose performance the public judges by unusually high standards, because the public is unusually reliant on them for principled performance.

REFERENCES

1. Berger, Sandy (2005). "Sandy Berger's Great Age Guide to the Internet". Que Publishing's 0-7897-3442-7
2. Jump up to: a b c "Architecture of a Metasearch Engine that Supports User Information Needs". 1999.
3. Ride, Onion (2021). "How search Engine work". onion ride.
4. Lawrence, Stephen R.; Lee Giles, C. (October 10, 1997). "Patent US6999959 - Meta search engine" – via Google Books.
5. Voorhees, Ellen M.; Gupta, Narendra; Johnson-Laird, Ben (April 2000). "The collection fusion problem".
6. "The Meta-search -- Search Engine History".
7. "Search engine rankings on Hot Bot: a brief history of the Hot Bot search engine".
8. Shu, Bo; Kak, Subhash (1999). "A neural network based intelligent metasearch engine". Information Sciences. **120** (4): 1–11. CiteSeerX 10.1.1.84.6837. doi:10.1016/S0020-0255(99)00062-6.
9. Kak, Subhash (November 1999). "Better Web searches and prediction with instantaneously trained neural networks" (PDF). IEEE Intelligent Systems.
10. "New kid in town".
11. "Rediff Search: Teens have.com of age!".
12. "Tazaa.com - About Tazaa.com".
13. "ABOUT US - Our history".
14. Spink, Amanda; Jansen, Bernard J.; Kathuria, Vinish; Koshman, Sherry (2006). "Overlap among major web search engines" (PDF). Emerald.
15. "Department of Informatics". University of Fribourg.
16. "Intelligence Exploitation of the Internet" (PDF). 2002.
17. HENNEGAR, ANNE (16 September 2009). "Metasearch Engines Expands your Horizon".
18. MENG, WEIYI (May 5, 2008). "Metasearch Engines" (PDF).
19. Selberg, Erik; Etzioni, Oren (1997). "The MetaCrawler architecture for resource aggregation on the Web". IEEE expert. pp. 11–14.
20. Manoj, M; Jacob, Elizabeth (July 2013). "Design and Development of a Programmable Meta Search Engine" (PDF). Foundation of Computer Science. pp. 6–11.
21. Jump up to: a b c d Manoj, M.; Jacob, Elizabeth (October 2008). "Information retrieval on Internet using meta-search engines: A review" (PDF). Council of Scientific and Industrial Research.

22. Wu, Shengli; Crestani, Fabio; Bi, Yaxin (2006). Evaluating Score Normalization Methods in Data Fusion. Information Retrieval Technology. Lecture Notes in Computer Science. Vol. 4182. pp. 642–648. CiteSeerX 10.1.1.103.295. doi:10.1007/11880592_57. ISBN 978-3-540-45780-0.

23. Manmatha, R.; Sever, H. (2014). "A Formal Approach to Score Normalization for Meta-search" (PDF). Archived from the original (PDF) on 2019-09-30. Retrieved 2014-10-27.

24. Najork, Marc (2014). "Web Spam Detection". Microsoft.

25. Vandendriessche, Gerrit (February 2009). "A few legal comments on spamdexing".

26. Wang, Yi-Min; Ma, Ming; Niu, Yuan; Chen, Hao (May 8, 2007). "Connecting Web Spammers with Advertisers" (PDF).

27. Trapani, Gina (4 May 2005). "Safari's private (porn) browsing mode". Lifehacker. Retrieved 11 April 2010.

28. Foley, Mary Jo. "Microsoft to roll out more granular 'porn mode' with IE 8". ZDNet. Retrieved 4 October 2008.

29. Sadighi, Lalee. "Microsoft's Internet Explorer 8 Goes 'Porn Mode'". Red Herring. Archived from the original on 12 September 2008. Retrieved 4 October 2008.

30. Kidman, Angus. "Microsoft releases IE8 beta 2: MS porn mode included". APC. Retrieved 4 October 2008.

31. "Adobe Flash 10.1 supports "private browsing"". The H. Retrieved 14 August 2019.

32. "Adobe Flash Player Private Browsing May Force Change in Fraud Fight". eWeek. Retrieved 14 August 2019.

33. Paul, Ian (11 March 2014). "Three practical reasons to use your browser's private mode". PCWorld. Retrieved 14 August 2019.

34. Jump up to:[a] [b] Brownlee, Chip (31 July 2019). "Google's Chrome Update Just Unlocked Lots of Newspapers' Metered Paywalls". Slate Magazine. Retrieved 14 August 2019.

35. Jump up to:[a] [b] Bursztein, Elie. "Understanding how people use private browsing". Retrieved 14 August 2019.

36. Espiner, Tom. "Private browsing tools still leave data trail". ZDNet. Retrieved 14 August 2019.

37. "Private browsing: 16 good reasons to use incognito mode". ZDNet. Retrieved 14 August 2019.

38. Ulmer, Hamilton (23 August 2010). "Understanding Private Browsing". Blog of Metrics. Mozilla Foundation. Retrieved 24 August 2010.

39. Parchisanu, Daniel (9 November 2018). "How to go incognito in all web browsers: Chrome, Firefox, Opera, Edge, and Internet Explorer". Digital Citizen. Retrieved 9 January 2019.

40. "Microsoft Announces Availability of Internet Explorer 8" (Press release). Microsoft. 19 March 2009. Archived from the original on 22 March 2009. Retrieved 16 December 2011.

41. "Mozilla Cross-Reference mozilla1.9.1". Mozilla Foundation. Retrieved 26 May 2009.

42. Mateu, Roberto. "Opera 10.5 pre-alpha for Labs". Opera Software. Archived from the original on 24 August 2011. Retrieved 22 December 2009.

43. "Private Browsing for Amazon Silk". Amazon Inc. Archived from the original on 22 December 2014. Retrieved 18 November 2014.

44. Jump up to:[a] [b] Grothaus, Michael (12 April 2019). "Incognito mode won't keep your browsing private. Do this instead". Fast Company. Retrieved 14 August 2019.

45. "Incognito mode while browsing - Myths Busted - Privacyflake". www.