

Examples of search algorithms used in information retrieval*

Jaroslav Ertl

Slovak University of Technology in Bratislava
Faculty of informatics and information technology
`xertlj@stuba.sk`

1. November 2023

Abstract

Many people use search engines daily for information retrieval, whether it's for school, work, or personal use, but few understand the complex algorithms that power these systems. In the realm of Information Retrieval (IR), search algorithms play a crucial role in delivering relevant results from vast datasets. There are many types of search algorithms used in IR, each of them having its unique purpose.

You can have a graph-based algorithm, a keyword-based algorithm, and many more. In this article, we will dive into the world of different search algorithms and their application in IR. In each section, we will discuss different search algorithms, how they work, and their purpose.

You can find the article that guided me towards the topic of search algorithms in references.[1]

1 Introduction

In the digital age, the internet has become a crucial part of our lives, and search engines have become our number one trusted companions. Whether we're seeking answers, writing research for academic purposes or simply satisfying our unending curiosity. But have you ever stopped and thought how that keyword entry turned into relevant results you were seeking? While the act of typing a keyword into a search bar may seem straightforward, beneath the surface lies a world of complexity, driven by search algorithms.

In this article, we embark on a voyage through the realm of search algorithms. Our first destination on this journey will be the island of graph-based algorithms, where we will delve into the details of Hyperlink-Induced Topic Search (HITS)[section 2.1] and the revolutionary PageRank[section 2.2] algorithm, developed by Google's founders, Larry Page And Sergey Brin. These algorithms have put down the foundation of web search and have transformed the way we discover information online.

*Term project in the subject Engineering Methods, ac. year 2023/24, advisor: Ing. Richard Marko, PhD.

From there, we will set sail for the world of keyword-based algorithms. At this stage of our adventure, we will uncover the mysteries of the Vector Space Model (VSM), a mathematical structure that underpins the way documents and queries are represented in the digital space. The VSM allows us to understand how documents are transformed into mathematical vectors and how these vectors are used to determine the relevance of documents to a search query. We will also navigate through the logic of Boolean Search, which includes logical operators to retrieve documents that correspond to our search criteria. And let's not forget the Term Frequency-Inverse Document Frequency (TF-IDF), a cornerstone of keyword-based algorithms. This algorithm has paved the way for innovative approaches in information retrieval. In particular, it laid the foundation for the BM25 search algorithm, which has become one of the most common and influential algorithms used in many search engines today. We will explore the core principles of TF-IDF and how BM25 builds upon this concept to deliver relevant search results to users.

2 Graph-based algorithms

2.1 Hyperlink-Induced Topic Search (HITS)

Our first stop on this journey is an island of Hyperlink-Induced Topic Search(HITS). HITS is an algorithm that is widely used in link analysis. It helps identify webpages across the internet that are related to your specific search. HITS was developed by Jon Kleinberg and is based on the concept that high-quality websites should both link to relevant sites and be linked to by other reliable pages. Imagine the internet as a giant network of interconnected webpages, with each webpage represented as a node and links between them as connections. Some webpages are more important and trustworthy than others, and HITS helps us figure out which ones they are. To understand how this algorithm works and why it is important for us we need to define what authority and hub is. To keep things simple, from now on we will refer to the webpage as a node. Every node is assigned two scores, one for authority and one for hub.

- **Authority** - A node is considered high-quality if many other high-quality nodes refer to it. This essentially means that if trustworthy and valuable nodes on a particular topic link to a specific node, this node is claimed as authoritative for that topic.
- **Hub** - A node is considered as high-quality if it refers to many other high-quality nodes. In other words, if a node links to multiple authoritative nodes, it is seen as a hub, connecting users to valuable information on various topics.

For A better understanding of authority and hub, you can take a look at Figure 1. Node 1 has the highest score for hub because it links to the most nodes in this diagram(2 and 4). On the other hand, node 3 has the highest score for authority because it is being linked to by the most nodes(2 and 4).

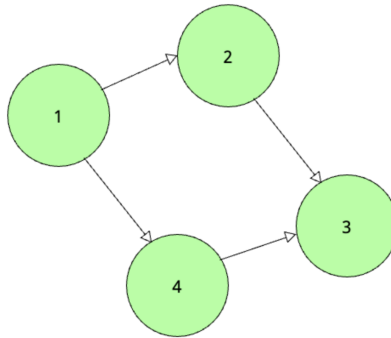


Figure 1: Explanation of hub and authority

Now let's take a look at how the HITS algorithm works:

1. **Initialization** - We start by assigning each node a score for both authority and hub. For simplicity, we often begin with an equal score, typically set to 1.
2. **Iteration** - Then we iterate through the nodes updating their authority and hub scores.
3. **Updating the score**
 - **Updating the authority** - For the current node, we calculate its authority score by summing the hub scores of all the nodes that link to it. In other words, if several webpages with high hub scores link to the current node, its authority score increases.
 - **Updating the hub** - For the current node, we calculate its hub score by summing the authority scores of all the nodes it links to. In other words, if the current connects to authoritative webpages its hub score increases.
4. **Repetition** - We repeat this process until the scores no longer change significantly, indicating that we've reached a stable state.
5. **Ranking** - The final authority and hub scores are used to rank the nodes. Nodes with higher scores are considered more relevant and trustworthy, allowing us to present users with the most valuable web content.

It's essential to note that the HITS algorithm's performance can vary depending on the size of the nodes it analyzes. The number of iterations required and the time taken to reach a stable state is strongly influenced by the number of edges and nodes within the network. On a larger scale, more iterations are necessary for the score of each node to stabilize, therefore causing more computation time. For instance, consider Figure 2, which represents a relatively small network consisting of 7 nodes and 18 edges. In this context, the HITS algorithm would require approximately 5 iterations to reach a stable state. The computation time in such a scenario is relatively short. However, in contrast, imagine a

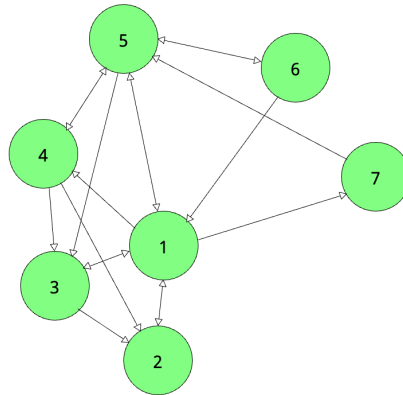


Figure 2: Iteration time example by T.Chou[2]

bigger network with 1000 nodes and 5000 edges. Even after 500 iterations of the HITS algorithm, this larger network might still not have reached a stable state due to its complexity.

Although HITS is playing significant role in information retrieval and in improvements of search engines, it has some challenges associated with itself.

- **Low Computational Efficiency** - As we have already mentioned, the HITS algorithm's computation efficiency varies depending on the size of the network and the data it works with. When working with huge amounts of data it can take vast amount of time to complete the results.
- **Results Can Be Manipulated** - The HITS algorithm is vulnerable by cheaters. For instance, a cheater can build a website that links to other high-quality websites, this website then gets a good hub score. Then the cheater allows this website link to his own website. This way he can increase the authority score of his website.

Yet, despite these challenges, the HITS algorithm remains a foundation tool in the realm of information retrieval, helping us to find relevant information on the web with precision and accuracy.[3]

2.2 PageRank

References

- [1] Weitong Zhang, Ronghua Shang, Zhiyuan Li, Rui Sun, and Jun Du. Personalized web page ranking based graph convolutional network for community detection in attribute networks. *IEEE Access*, 11:84270–84282, 2023.
- [2] T. Chou. Hits algorithm: Link analysis explanation and python implementation. <https://towardsdatascience.com/hits-algorithm-link-analysis-explanation-and-python-implementation-61f0762fd7cf>, 2019.

- [3] Weiming Yang. An improved hits algorithm based on analysis of web page links and web content similarity. In *2016 International Conference on Cyberworlds (CW)*, pages 147–150, 2016.