Wikipedia

Tom Rochette <tom.rochette@coreteks.org>

November 2, 2024 — 36c8eb68

0.1 Context

Wikipedia is the most popular and known encyclopedia of topics known to man. As such, it attempts to be complete and rigorous, which makes it a prime target as a source of knowledge.

Each article being uniquely identified by a URL, it allows us (humans) to *hopefully* uniquely identify knowledge topics using a common addressing scheme.

0.2 Learned in this study

0.3 Things to explore

- Build a network graph to explore Wikipedia
  - How should the graph be built?
  - Use tree/graph traversal algorithms to find the shortest path between two topics

1 Overview

The goal of this article is to find means for people to track their knowledge through Wikipedia as the source of common knowledge. As more and more people use a system in which this *knowledge* of knowledge is shared, it should be possible to develop tools such as "related pages" (a pool of pages users that have visited a certain page also visited), "stereotypical users" (a *user* that is interested in AGI will look into these articles) and more based on users profiles.

2 What should we track?

In order to determine what information should be tracked, one has to ask himself the type of queries he will want to execute in the future.

In our case, we want to be able to establish user profiles, that is, to characterize a user by the page he's visited.

Furthermore, one of the purposes of establishing a profile is to be able to tell its user what he has already explored.

Here's a non-exhaustive list of attributes we might be interested to track in order to establish such profile:

- View history
  - Article name
  - Date
  - Time
  - Origin/Referrer
- Some way to track the article was read (since viewing the page does not necessarily translate to having read it)

- Edited articles

Given this information for each user, it is possible to construct a tensor where the dimensions are $user \times user \times \boldsymbol{page}$, in other words, a matrix where each cell is a vector of all Wikipedia pages, and where there is a 0 if both users have not seen the page, and a 1 if they both have. The page vector may be flattened into a single number that represents the number of pages both users have seen in total, thus providing us with a matrix that indicates the total number of articles seen by user pairs.

Constructed this way, if an existing user has read all of Wikipedia (user A), and a new user has seen a single page (user B), then their shared cell will have a total value of 1. Furthermore, as the new user sees new pages, the more this value increases. If we were to compute a value between 0 and 1 using the number of pages seen by both users as the numerator and the number of unique pages seen by user A and user B as the denominator, then this number would slowly increase to 1.

In another case, where an existing user has seen a given set of Wikipedia pages and a new user begins browsing Wikipedia, this number will only grow as long as they see similar pages. When user A or B browses to a page neither have seen, this value decreases. If user A has seen page X, and then user B also navigates to this page, then this value increases.

$$0 \leq \frac{|\text{Pages seen by both A and B}|}{|\text{Pages seen by A or B}|} \leq 1$$

3 Relations within the content

Wikipedia contains a huge amount of internal links (links between articles) as well as external links that mostly serve as references to the content within a given page. Internal links can be extremely valuable to us since they can allow us to build a network of topics.

There are various types of internal links, which we'll describe below:

3.1 Content

Within the content of an article, there will often be terms which are themselves defined as an article.

A definition is generally based on other definitions which are themselves based on other definitions and so on.

As such, we can determine that internal links in the content of articles are reciprocal relations between the two articles. Another way to determine the relationship between two articles is to verify if both link to one another.

There are many types of relations and reasons that could explain these relations:

- A topic created another topic (parent-child, predecessor-successor)
- Topics are part of the same domain (siblings)
- A topic can be divided into sub-topics

3.2 See also

The *See also* section of articles contains a list of vetted related articles that may interest the reader of the current article. As such, it may have no relation to the current article other than being of a similar domain or of possible interest.

3.3 Categories

As the definition of *categories* of Wikipedia states very well:

> **Categories** are intended to group together pages on similar subjects.

The category itself though can be seen as a container of all the articles. In other words, it can serve as an abstraction of all the articles under its umbrella.

4 Additional relations

4.1 Redirects

Some concepts are known under various names, but in order to reduce the amount of confusion between them, it is possible to create redirections which will bring to the generally accepted term of a concept. Thus, this allows us to determine that if we've read the article a redirection points to, we probably can safely assume that the point of redirection is also something we *now* know.

5 Wikipedia exploration application

With all this information in hand, it is now time to design an application that will allow us to navigate through Wikipedia while allowing us to better guide our exploration.

One simple way to do so would be to develop a client/server system, where the client is a Chrome/Firefox extension that will add a minor UI to Wikipedia and allow the user to indicate if a page has been read. When clicked, a button would send an API call to a remote service that would track this information for the user. Furthermore, upon navigation, an API call would also be sent to this remote service, which would record which Wikipedia article was viewed. From the UI widget, the user would also be able to fetch reading recommendations from the remote service.

The service would act mostly as a storage service, as well as a platform where metrics would be computed to build a list of articles to suggest to a user.

6 See also

7 References

7.1 Wikipedia API endpoints

- https://www.mediawiki.org/wiki/API:Categories
- https://www.mediawiki.org/wiki/API:Links
- https://www.mediawiki.org/wiki/API:Linkshere
- https://www.mediawiki.org/wiki/API:Redirects

7.2 Wikipedia visualization tools

- http://sepans.com/sp/works/wikistalker/
- http://en.eyeplorer.com/show/
- http://www.idea.org/WikiNodes.htm

7.3 Wikipedia toolkits

- https://github.com/Wikidata/Wikidata-Toolkit

7.4 Mindmap

- https://github.com/kennethkufluk/js-mindmap

7.5 Collaborative filtering

- https://en.wikipedia.org/wiki/Collaborative_filtering