Hyperrealised Semantic Search Specification

Mission

- Provide a search engine which aims to facilitate document search by allowing users to interactively remove concepts that do not relate to the content they are looking for.
- To optionally provide more context during search by referring to previous searches.
    - Reduce the number of keywords needed to find relevant documents.
    - Improve the accuracy of search by finding documents that are more relevant to the user.
- Improve the speed at which users can find the content they are after.
- Make the search experience more natural by interacting with the user.
- Allow search for documents related to content provided by the user.

Introduction

The main requirements of our semantic search software are divided into the sections of this document. These requirements are from the perspective of the end user and therefore written in user language without technical specification.

Hyperrealised Semantic Search Specification

Contents

Hyperrealised Semantic Search Specification

1. Functional Requirements

1.1. Semantic Search

Users should be able to search for entities (as defined below) related to the content they provide as a query.

1.1.1. Types of entities:

1.1.1.1. Wikipedia articles (primary requirement).

Articles from the Wikipedia encyclopaedia should be searchable.

Links to the current revision of the articles should be returned with results.

1.1.1.2. Related words (primary requirement).

Suggest alternative keywords to those provided by the user that may improve search results.

1.1.1.3. Documents provided by the user (secondary requirement).

Documents previously provided by a user that is not anonymous should be searchable by that user and be used to provide more context to subsequent searches made by the user if they so choose (see 1.2.1).

1.1.1.4. Documents on the user's computer (tertiary requirement)

Users should be able to search for documents on their computers based on the text contained within those documents.

All document types that contain text should be supported.

1.1.2. Types of queries:

1.1.2.1. Keywords (primary requirement).

The user can specify a list of keywords that somehow relate to the documents they are searching for.

The keywords specified by the user do not have to directly relate to the documents they are after but may be somewhat removed.

1.1.2.2. Documents provided by the user (secondary requirement).

Users should be able to provide URLs of documents on the internet as search queries or to provide extra context to searches the user may make.

Search results should prioritise documents related to the one provided.

All document types that contain text should be supported.

1.1.2.3. Uploaded documents provided by the user (tertiary requirement).

The user should be able to upload a document to perform the same function as requirement 1.1.2.2 if the document is not accessible on the internet.

1.2. User Accounts

1.2.1. Users can optionally create password protected accounts to keep track of their search history (secondary requirement).

The search results displayed to the user should be influenced by their previous searches and those searches results in order to improve relevance if they so choose.

1.2.1.1. The account should be uniquely identified by an email address supplied by the user.

Only one account can be associated with an email address.

1.2.1.2. The user should be able to change the email address associated with an account provided the email address they specify is not associated with another account.

1.2.1.3. The email address associated with the account should be verified upon creation of the account in order to prevent automated creation of accounts and to ensure that communication can be made with the user.

1.2.1.4. The user should specify a password in order to protect their account from access by others.

1.2.1.4.1. Eavesdropping to obtain a users password should not be feasible.

1.2.1.4.2. If the user forgets their password they should be able to reset it.

Confirmation should first be made by email in order to prevent the password being reset by others.

1.2.1.4.3. Password strength must be verified when set in order to ensure the security of user accounts.

Passwords must be at least 6 characters long.

1.2.1.5. Users should have access to their search history.

1.2.1.5.1. Recorded history should include previously made search queries and the results selected after making each query.

1.2.1.5.2. Users may clear their history at any time if they would like to remove any influence previous searches would have on subsequent searches.

1.2.1.6. Accounts may be terminated at any time by the user.

1.2.1.6.1. All information stored regarding the user should be removed from all active information stores in order to free the resources they required.

1.2.1.6.2. Once an account is terminated, the email address that was associated with it should be free to be associated with a new account.

1.2.2. Users do not have to create an account to use the services (primary requirement).

Current and future searches of anonymous users should not be influenced by their search history.

1.3. User Feedback

Users should be able to give feedback about the relevance of each search result.

1.3.1. Feedback should be used to adjust the search results.

1.3.1.1. Positive feedback should increase the relevance of the selected entity and closely related entities.

1.3.1.2. Negative feedback should significantly decrease the relevance of the selected entity and closely related entities.

1.3.2. Users should be able to specify feedback as a value on a continuum from “entirely irrelevant” to “exactly what I want”.

1.3.3. Feedback of all users should affect the relevance of entities and thus ordering of results.

1.4. User Interfaces

Users should be able to access the facilities provided by this hosted service through the internet.

1.4.1. A user interface should be accessible on many diverse internet connected devices.

1.4.1.1. There should be a web based user interface to provide access from internet connected devices with web browsers. (primary requirement)

The web based user interface should be standards compliant as to support as wide a range of web browsers as possible.

1.4.1.2. There should be support for a variety of pluggable user interfaces to both provide access from a greater range of devices and allow alternative user interfaces. (secondary requirement)

There should be an Application Programming Interface that is accessible from the most restricted of internet connections.

The API should be compatible with a wide variety of programming languages.

1.4.2. User interfaces should be minimal, streamlined and intuitive to use.

Users should not be delayed in performing their search by distractions.

Layout of interface should not be cluttered.

Novice users should have no trouble using user interfaces.

There should be simple messages and prompts to assist those with no previous experience.

1.4.3. Should support users ability to use features defined in section 1.1 (Semantic Search).

1.4.4. Should support users ability to use features defined in section 1.2 (User Accounts).

1.4.5. Should support users ability to use features defined in section 1.3 (User Feedback).

1.5. System Logs

Logs of service activity should be accessible for administration purposes and to help with system improvement.

1.5.1. Log files should contain access information.

1.5.1.1. Types of information that should be recorded should include:

1.5.1.1.1. The user involved (if not anonymous)

1.5.1.1.2. The IP address of the remote computer.

1.5.1.1.3. The date and time the activity occurred.

1.5.1.1.4. The activity performed.

1.5.1.1.5. The response time that was attained.

1.5.1.1.6. Whether any errors occurred while processing user activity.

1.5.2. Log files should be available in a common format, specifically XML, so that they can easily be read by a variety of software.

2. Non-Functional Requirements

2.1. Libraries

The libraries that should be used include the following:

2.1.1. ART

2.1.2. WordNet

Lexical database used to define relationships between words.

2.2. Scalability

2.2.1. The software should scale to the provided system resources in order to allow many users to perform queries.

2.2.2. The number of users should be limited to prevent unacceptable search times due to overloading of resources.

Users should be displayed a friendly apologetic message in the event that their search query could not be completed.

2.2.3. Performance should improve with the addition of system resources (hard drive space, main memory, CPU power and network speed).

2.3. Reliability

2.3.1. Malformed search queries should not disable the service but rather display an informative message as to why the query is malformed.

2.3.2. Service uptime should not be dependent on the software but rather outside influences such as power failures and network outage.

2.4. Performance (low priority)

2.4.1. The response time of a query given in keywords should not be significantly greater than the sum of network delays. The time taken to generate the response to a query should be negligible.

2.4.2. The response time of a query given as a URL or an upload document may be significantly longer as it is more dependent on network speeds.

2.4.2.1. The user should be notified of this added delay.

2.4.2.2. Users should be given the option to cancel the query and return to the search prompt.

2.4.2.3. The only limitation on performance should be the software libraries employed. There should not be an unjustifiable bottle neck within the software.