Melany Paola Rivera Lores

NAO ID: 3332

Date: September 30th, 2025

Pathway: Technoready Bécalos

Challenge: Server and Database Commands

Technical Report

Introduction

The present technical report provides an overview of the Google Scholar API through SerpAPI. It summarizes aspects required for integration into applications and systems, focusing on endpoints, authentication, query parameters, response formats, usage limits, and examples.

Endpoints

The API provides access to Google Scholar data through different endpoints. The most relevant include:

- google_scholar: General search for publications.
- google_scholar_author: Retrieve author profiles and publication details.

Authentication Methods

Access to SerpAPI requires an API key. The key must be included as a parameter in all requests to authenticate and authorize usage.

Query Parameters

The API supports various parameters to customize searches:

- q: Search query string.
- author_id: Unique identifier for author profiles.
- hl: Language of the results (e.g., 'en').
- start: Defines the starting index for pagination.
- num: Number of results to return per request.

Response Formats

Responses are returned in JSON format, structured for easy parsing and integration. The JSON objects include metadata such as titles, authors, publication dates, and links.

Usage Limits

SerpAPI enforces request quotas depending on the selected plan, but in this case, free plan to offer 100 requests per month.

Code Examples

Below is an example of how to use the Google Scholar API with SerpAPI in Java:

Biology

```
Map<String, String> parameter = new HashMap<>();

parameter.put("engine", "google_scholar");
parameter.put("q", "biology");
parameter.put("api_key",
"3c80b3f19d74557d14f0bdd138545d703f072aca20e602649f471e2a387a987a");

GoogleSearch search = new GoogleSearch(parameter);

try {
  JsonObject results = search.getJson();
  var organic_results = results.get("organic_results");
} catch (SerpApiSearchException ex) {
  System.out.println("Exception:");
  System.out.println(ex.toString());
}
```

Machine Learning in Healthcare

```
Map<String, String> parameter = new HashMap<>();

        parameter.put("engine", "google_scholar");
        parameter.put("q", "machine learning in healthcare");
        parameter.put("hl", "en");
        parameter.put("api_key", "YOUR_API_KEY_HERE");

        GoogleSearch search = new GoogleSearch(parameter);

        try {
            JsonObject results = search.getJson();
            var organicResults = results.get("organic_results");
            System.out.println("Search Results: " + organicResults);
        } catch (SerpApiSearchException ex) {
            System.out.println("Exception:");
            System.out.println(ex.toString());
        }
```

Conclusion

The Google Scholar API through SerpAPI provides an important way to integrate academic data into applications and systems. By using authenticated requests, customizable query parameters, and structured JSON responses, developers can efficiently manage and analyze scholarly information.