

COMMAND LINE SEARCH ENGINE

OPERATING SYSTEM PROJECT REPORT

Submitted by

S. Prabhakaran

S. P. Vijay

TABLE OF CONTENTS

NO.	TITLE	PAGE NO.
1.	Introduction	2
2.	Features	2
3.	Usage	3
4.	Examples	4
5.	Troubleshooting	5
6.	Reference	6
7.	System Requirements	6

Introduction

Googler is a command-line power tool to search Google (Web & News) from the terminal. It shows the title, URL and text context for each result. Results are fetched in pages. Next or previous page navigation is possible using keyboard shortcuts. Results are indexed and a result URL can be opened in a browser using the index number. Supports sequential searches in a single instance.

Features

- Uses HTTPS connection
- Fetch n results in a go
- Start at nth result
- Fetch and navigate next or previous set of results
- Continuous search: fire new searches without exiting
- Reconnect on new search even if connection is closed due to inactivity
- Disable automatic spelling correction and search exact keywords (default: enabled)
- Easily open result URLs in browser from cmdline using index number
- Browser (Chromium and Firefox based) errors and warnings suppression
- Show full contextual text snippet in search results
- Specify search duration (in hours / days / weeks / months / years)
- Fetch results from Google News section
- Country/domain specific search (28 top-level domains supported, default: .com)
- Supports Google search keywords like filetype:mime, site:somesite.com etc.
- Open filetype specific links in browser, the links are handled by browser for the type
- Optionally open the first result directly in browser (as in *I'm Feeling Lucky*)
- Specify language preference for results
- Handle first level of Google redirections (reports IP blocking by Google)
- Unicode in URL works
- Skip links to Google News, Images or blank URLs in web search results
- UTF-8 request and response
- Fetch gzip compressed results

- Works with Python 2.7.x and 3.3.x or later
- Enable/disable color output (default: colorful)
- Enable/disable debug logs (default: disabled)
- Fast and clean (no ads or clutter)
- Minimal dependencies
- Open source and free

Usage

usage: googler [-s N] [-n N] [-N] [-c TLD] [-l LANG] [-x] [-C] [-j] [-t dN] [-d]
 KEYWORD [KEYWORD ...]

Perform a Google search and print results to stdout.

positional arguments:

KEYWORD search keywords

optional arguments:

-s N	start at the Nth result
-n N	show N results (default 10)
-N	show results from news section
-c TLD	country-specific search with top-level domain .TLD, e.g., 'in' for India (see https://en.wikipedia.org/wiki/List_of_Google_domains for a full list of TLDs)
-l LANG	display in language LANG
-x	disable automatic spelling correction
-C	disable color output
-j	open the first result in a web browser
-t dN	time limit search [h5 (5 hrs), d5 (5 days), w5 (5 weeks), m5 (5 months), y5 (5 years)]
-d	enable debugging

prompt keys:

g terms	initiate a new search for 'terms' with original options
n, p	fetch next or previous set of search results
1-N	open the Nth result index in browser
Enter	exit googler (same behaviour for an empty search)
*	any other string initiates a new search with original options

Examples

1. Google **hello world**:
\$ googler hello world
2. Fetch **15 results** updated within last **14 months**, starting from the **3rd result** for the string **cmdline utility** in **site** tuxdiary.com:
\$ googler -n 15 -s 3 -t m14 cmdline utility site:tuxdiary.com
3. Read recent **news** on gadgets:
\$ googler -N gadgets
4. Fetch results on IPL cricket from **Google India** server in **English**:
\$ googler -c in -l en IPL cricket
5. Search quoted text e.g. **it's a "beautiful world" in spring**:
\$ googler it's a "\"beautiful world\" in spring
6. Search for a **specific file type**:
\$ googler instrumental filetype:mp3
7. Disable **automatic spelling correction**, e.g. fetch results for googler instead of google:
\$ googler -x googler

8. **I'm feeling lucky** search:

```
$ googler -j leather jackets
```

9. **Website specific** search alias:

```
alias t='googler -n 7 site:tuxdiary.com'
```

10. Alias to find **definitions of words**:

```
alias define='googler -n 2 define'
```

11. Lookup n, p, g co or a number at the **navigation prompt**: As the navigation prompt recognizes n, p, g or numbers as commands, you need to prefix them with g, e.g., g n, g g keywords, g 1984.

12. Input and output **redirection**:

```
$ googler -C hello world < input > output
```

Note that -C is required to avoid printing control characters (for colored output).

13. **Piping** googler output:

```
$ googler -C hello world | tee output
```

14. More **help**:

```
$ googler
```

Troubleshooting

In some instances googler may show fewer number of results than you expect, e.g., if you fetch a single result (-n 1) it may not show any results. The reason is Google shows some Google service (e.g. Youtube) results, map locations etc. depending on your geographical data, which googler tries to omit. In some cases Google (the web-service) doesn't show exactly 10 results (default) on a search. We chose to omit these results as far as possible. While this can be fixed, it would need more processing (and more time). You can just navigate forward to fetch the next set of results.

Reference

https://en.m.wikipedia.org/wiki/Web_search_engine

<http://www.if-not-true-then-false.com/2009/google-search-from-linux-and-unix-command-line>

<https://developer.chrome.com/devtools/docs/commandline-api>

<http://xmodulo.com/access-popular-search-engines-command-line-linux.html>

<http://superuser.com/questions/47192/google-search-from-linux-terminal>

Reference Paper: The command line style search interface (CLI) Workshop conducted by Stanford, California, US copyrights 2014.

System Requirements

Processor	: Intel® Core™ i5-5200 CPU @ 2.20GHz 2.19GHz
Installed Memory(Ram)	: 4 GB
System type	: 64-bit Operating System
Operating System	: Ubuntu (Linux)

Code

<https://github.com/prabhakaran9397/OS-project/blob/master/googler>