COVER PAGE

Student Information

Name:	Alexander Baldwin
Email:	Baldwin_alexander@student.ceu.edu
Student ID:	182491
Supervisor:	Prof. Laszlo Sallo

Abstract

The following report summarises the business need for a keyword density calculator, and the app created as a response. This project was undertaken in pursuit of an MSc in Business Analytics from the Central European University, and represents the Capstone Project concluding the programme.

The business need considered is in the field of Search Engine Optimisation, and the ever-changing environment that entails. One factor, of many, is the content of a page or article; particularly the keyword or phrase a user searches for. This app is designed to scrape the highest-performing pages, calculate the keyword density in each, aggregate this information, and provide recommendations to the user. This is designed to streamline a task that must otherwise be done manually. While tools exist to scrape and analyse single sites, there is no equivalent market for a system to compile and analyse information from a number of sites simultaneously. Finally, this report will consider the limitations of the app in its current prototype form, and the opportunities and limitations in the wider business context.

Contents

Capstone Project – Keyword Density Calculator

Business Environment

Digital Content

Digital content is an exponentially growing field, offering products and services, off-the-shelf and bespoke, business-to-business, and business-to-customer. In addition to commercialised content there are academic publications, government reports and statistics, news media content, personal blogs, and social media. Beyond this there are games, videos, image-boards, and media distribution services.

At time of writing, there are nearly two billion websites, and 4 billion internet users in the world[1]. In a single day, nearly 5 million blog posts, 600 million tweets, and 61 million Instagram photos will be created. All of this content is far beyond the capability of an individual, or even a collaborating group of people, to monitor and record.

Search Engines

In order to make use of the content that is generated, users must be able to find *relevant* content to their interests or needs. When the internet was in its early stages, information could be shared by word-of-mouth; users could be directed to sources and repositories of information by others who knew of the content and quality of specific sites.

As the scope and scale of the internet increased, automated tools were required to record what content could be found, and where. "Scrapers" or "spiders" are the name given to the programmes that examine the content of webpages and follow the hyperlinks contained within, creating a map of interconnected content. Once this was recorded, it could be used to direct users to content related to search queries.

Search Engine Algorithms

Search engine algorithms continuously increase in complexity, with both progressive and radical changes to behaviour, particularly in response to attempts to exploit the setup. For example, originally a page was ranked for the frequency of a search-term appearing. This was easily exploited by repeating a word far more than good use of language would allow. This reduced the user experience and could easily prioritise irrelevant content. Then the algorithm might be adapted to focus on "related words", so the system might be exploited by using many synonyms.

Search engines had to find relevant, quality content, while avoiding exploitation that would allow bad actors to profit by disadvantaging legitimate search engine users. The practice of adapting web content to perform well in the search engine algorithm is referred to as SEO or Search Engine Optimisation.

Search Engine Optimisation

Deliberate exploitation of an algorithm to artificially increase search engine performance is referred to as *Black Hat SEO*. This consists of providing low-quality content that reduces the user experience in order to gain more exposure, generate sales activity, or for some other purpose.

However, while exploiting the algorithm is frowned upon, combated, and sometimes punished, search engine providers often offer guidelines for good-practice, or *White Hat SEO* that raises the performance of high-quality content. Examples of such activity include appropriate use of heading tags in HTML, use of keywords in appropriate places and at a reasonable frequency and density, as well as use of alt-text, hyperlinks, and submission of supporting documents like XML webmaps. This range of good-practice tasks and practices has created both a business function of SEO specialism, and an industry of SEO service providers.

[1] http://www.internetlivestats.com/ - Retrieved at 19:14 - 12/12/18

Content Creation

Content creators who wish to comply with good SEO practice have a combination of specific and vague guidelines. Some are performed rarely, or are easily automatable, such as uploading an XML sitemap, or creating a header and footer field for a website. However, specific unique content is harder to optimise.

When writing articles, one must consider how a web-scraper will recognise the topic, and thus flag it as relevant content when a user searches for a related query. Elements such as header tags, alt-text on images, and *snippets* accessible by Google's scraper can all support this. Additionally, raw text analysis can provide a measure of value. Content is often clustered, and the metrics for high-performing content can be contained within a topic. For example, in a mathematical paper, one might expect a high density of technical terms, and a good-quality piece of content would be expected to have high keyword density. By contrast, an article discussing hill-walking in Scotland is unlikely to require the repetition of many key phrases, but might instead be better evaluated by the variety of phrases used. Thus a low keyword density might be an indicator of good content.

For this reason, when preparing content, a creator should examine the existing market for a search term and identify the trends of high-performing content in that field. The total word-count and keyword density of high performing content might be quite similar, in which case one can infer that it would be advisable to work to a similar standard to generate content that the search algorithm will flag as relevant.

Manual Process

A variety of services exist to calculate the keyword density of a given keyword on a given page. However, few services are available that examine the general market. When working in the industry, the author of this piece had to manually gather information from the top-performing sites from a search, then record and manually calculate the average wordcount and keyword density for a given topic. This process proved effective, and optimised content frequently performed well in search engine results, however the process was time-consuming.

The task frequently took an hour or more, involving using a tool to evaluate a page, copying the results into a spreadsheet, and using the spreadsheet to calculate summary statistics. This allowed for user-error to creep in, including copying incomplete links, or inaccurately transposing data.

Automated Process

Thus, there was a business need for the task, but also an incentive to automate and streamline the process. Google provides a Search Engine API that produces easily analysed search engine results, and the entire scraping and analysis process could be automated. The initial code for this task could be run on a standard computer in less than two minutes. However, this required the installation and managing of R, R Studio, and 10 dependent packages.

Making this tool a standalone app, hosted on an instance accessible by a URL makes the benefits of the tool available to users without the requirement to either understand or maintain the running environment.

Deliverable

The output deliverable of this project represents a minimum viable product, suitable for a private user to generate the results and summary of the top 10 results from the Google Search Engine in less than a minute. This reduces the labour required for a repetitive task and represents a major efficiency saving for business.

This is not a product ready for commercial distribution, due to limitations in the hosting environment, the rudimentary user interface, and periodic bugs in the analysis and summarisation functions. However, the application works in the majority of test-cases, and provides a consistent, clear output. This can be recorded and the results analysed to identify trends in performance or trends over time for a single keyword.

Additional and in-depth analysis could be added to the application, however such specification creep was considered to be unwarranted in the case of this capstone project. The project deliverable was defined as a private solution to an individual problem, and this could be used either by a solo content creator, or as an internal tool for a small-business. If a company wished to develop this into a commercially distributed tool it would require more thorough examination of the legal and technical environment in which it would be launched.

As a Capstone

This deliverable is particularly representative as a capstone project because it demonstrates the broad range of technical skills developed through the MSc Business Analytics programme. The author initially had minimal experience with R as a functional programming language, and limited experience with statistics. In addition to building on these skills, he was introduced to Python as a second programming language, and the use of APIs to access data from external sources, as well as the use of Shiny as a package to generate standalone applications.

This deliverable is therefore a combination of the use of APIs, the processing of untidy text data, the quantitative analysis of that data, and the generation of visualisations and summary statistics, hosted on a cloud-computing service.

This demonstrates not only the material learned but also the synthesis of disparate elements into a single project that meets a business need that the author was unable to meet prior to this course.