The Sum of All Human Knowledge in Your Pocket: Full-Text Searchable Wikipedia on a Raspberry Pi

Jimmy Lin
The iSchool — College of Information Studies
University of Maryland, College Park
jimmylin@umd.edu

ABSTRACT

We demonstrate a prototype that takes advantage of open-source software to put a full-text searchable copy of Wikipedia on a Raspberry Pi, providing nearby devices access to content via wifi or bluetooth *without requiring internet connectivity.* This short paper articulates the advantages of such a form factor and provides an evaluation of browsing and search capabilities. We believe that personal digital libraries on lightweight mobile computing devices represent an interesting research direction to pursue.

Categories and Subject Descriptors: H.3.7 [Information Storage and Retrieval]: Digital Libraries

Keywords: low-power devices; mobile computing

1. INTRODUCTION

Wikipedia aspires to provide every single person on the planet free access to the sum of all human knowledge.[1] According to Alexa, it is the 7th-most visited website in the world, with approximately 40% of its traffic coming from search engines (as of December 2014). Today, accessing Wikipedia requires internet connectivity for most users, but what if this weren't the case? We have assembled a prototype that takes advantage of existing open-source software to provide a full-text searchable copy of Wikipedia on a Raspberry Pi, an inexpensive computer the size of a deck of playing cards. This provides any nearby device access to Wikipedia either via wifi or bluetooth *without requiring internet connectivity.* Before describing our prototype, we first address two obvious questions:

Why is this better than using a web search engine? Today, internet connectivity is ubiquitous and Wikipedia is easily accessible from a web browser. What's compelling about Wikipedia on a Raspberry Pi?

First, ubiquitous connectivity is an illusion frequently disrupted by dead zones, data limits, and network congestion. This is especially true in the developing world, where many users do not have access to (or cannot afford) the internet. Second, full-text search on Wikipedia via a web search engine lacks the flexibility of open software running on a device under one's control. Our prototype runs a completely open software stack, which provides a platform for arbitrary customization. With full control over both the hardware and software, users are no longer beholden to external services.

Finally, having a local resource eliminates the possibility of third parties collecting access statistics, interaction data, and query logs. Users could issue sensitive queries without privacy concerns, e.g., targeted ads that start appearing in a cloud-based email service. A self-contained search engine also opens the door to new search personalization strategies based on personal data that users might not feel comfortable sharing with a third party.

Why is this better than using a mobile phone? Why couldn't the functionalities described here be implemented on a mobile phone, e.g., as an Android or iOS app? We see a few advantages of the Raspberry Pi, the biggest of which is flexibility. The machine runs a full-featured operating system, which provides much greater control over the hardware. It is also significantly more expandable via numerous USB ports and other custom connector ports. The storage requirements of Wikipedia and the full-text index still represent a substantial fraction of mobile devices' total capacity today: on a storage per unit cost basis, the Raspberry Pi is cheaper, and so it makes sense to offload Wikipedia onto a separate device. Finally, users today have multiple devices, e.g., mobile phones and tablets, and may desire access from all of them. Rather than duplicating content, it makes sense to hold a single copy of the data on a stand-alone device.

2. SYSTEM DESIGN

The Raspberry Pi (Figure 1) is an inexpensive, single-board computer the size of a deck of playing cards originally developed to promote computer science education, but has gained popularity in the maker community. The machine is based on the Broadcom BCM2835 system on a chip, which runs a 700 MHz processor from the ARM11 family. The B+ model used in our experiments has 512 MB RAM, and storage is provided via a microSD card.

The starting point of our demonstration is the open-source Kiwix package, which is an offline reader for web content that supports the ZIM format. The "Kiwix plug" project began in 2012 to provide offline Wikipedia access in a small form factor, specifically for developing countries in Africa.[2]

[1] `http://en.wikiquote.org/wiki/Jimmy_Wales`

[2] `https://blog.wikimedia.org/2012/06/28/`

JCDL'15, June 21–25, 2015, Knoxville, Tennessee, USA.
Copyright is held by the owner/author(s). Publication rights licensed to ACM.
ACM 978-1-4503-3594-2/15/06 ...$15.00.
DOI: http://dx.doi.org/10.1145/2756406.2756938.

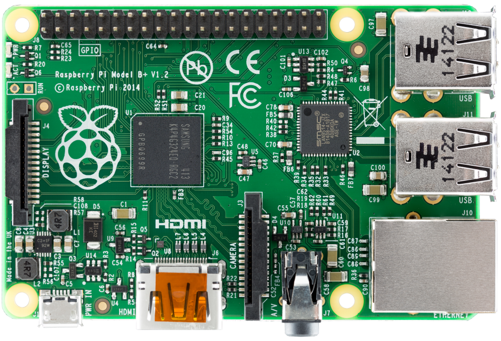

Figure 1: Raspberry Pi B+ Model; peripherals connect via USB ports (right) and HDMI (bottom).

The software has since been ported to the ARM architecture, and specifically, to the Raspberry Pi.

Kiwix comes with a stand-alone HTTP server that allows any connected device to access Wikipedia content; it also provides built-in full-text search. We evaluated the performance of both capabilities under two different connectivity options: wifi and bluetooth. Note that wifi is needed only to connect to the Raspberry Pi; external connectivity to the internet is not required. We also compared the performance of Kiwix's full-text search to ATIRE [3], a state-of-the-art search engine written in C.

3. EVALUATION

Our hardware configuration is as follows: a Raspberry Pi B+ model overclocked to "turbo" at 1 GHz, a 64 GB microSD card, and external USB dongles for wifi and bluetooth. To avoid idle power draw, in the wifi configuration the bluetooth dongle was removed, and vice versa. We used Kiwix 0.9 with the pre-built version of English Wikipedia from January 2014 (containing all text but no images). The article content occupies 12 GB and the full-text index another 11 GB. We used the latest version of ATIRE; its index is substantially more compact at 2.0 GB (Wikipedia articles were first cleaned to remove wiki markup; terms were stemmed but no stopwords were removed). In addition to measuring performance in terms of latency, we also measured power usage with an electricity usage monitor. The cost of our setup is as follows (as of Dec. 2014, in USD): the Raspberry Pi costs $35 and the 64 GB microSD card costs $25 (a 32 GB microSD card costs $13 and provides sufficient space). The AC power supply costs $9, or alternatively, one can purchase battery packs of varying capacities. These costs do not include the display and peripherals.

Our first experiment evaluated the performance of the device for serving Wikipedia article content. We sampled 600 "popular" Wikipedia articles as a query set using the following procedure: from Wikipedia page view statistics,[3] we downloaded all hourly logs for December 1, 2014; for each hour, we retained the top 1000 articles, and from the union, we sampled 600 articles total. We then accessed these articles from a client laptop connected to the Raspberry Pi (either wifi or bluetooth). Results averaged over three runs are shown in Figure 1, which lists article access latency (in seconds) as well as the power consumption during the experiments (in Watts). For reference, while idle, the Raspberry Pi draws 2.1 W using wifi and 1.6 W using bluetooth.

	latency (s)	power (W)
Wifi	0.88	2.7
Bluetooth	3.57	1.8

Table 1: Latency and power consumption of Wikipedia article access using Kiwix.

	Kiwix	ATIRE
Wifi	1.94	1.04
Bluetooth	2.46	1.08

Table 2: Query latency (seconds) for full-text search, comparing Kiwix and ATIRE.

Results show that while bluetooth consumes less power than wifi (which is consistent with previous work [1]), page access is substantially slower. Using wifi, the device is reasonably responsive and appears sufficient to support interactive browsing. The data transfer rate over bluetooth is substantially lower than over wifi, which explains slower page accesses. Whether the tradeoff of speed for lower power is worthwhile depends on user preferences.

Our second experiment compared full-text search provided by Kiwix internally and the ATIRE search engine. This evaluation focused only on efficiency (query latency) and left aside the issue of effectiveness (result quality), since we did not have relevance judgments to assess search output. We used 300 queries from the web track of the Text Retrieval Conferences (TRECs) from 2009 to 2014 [2]. As with before, queries were issued from a laptop connected to the Raspberry Pi. Results averaged over three runs are presented in Table 2, showing query latencies for both Kiwix and ATIRE using wifi and bluetooth.

We see that ATIRE is substantially faster than Kiwix's built-in search (and takes less space in terms of index size), although both are slower than what users have come to expect today (query latencies of a couple hundred milliseconds). The performance of Kiwix degrades substantially when switching from wifi to bluetooth, whereas ATIRE performance is about the same. Using wifi, the Raspberry Pi draws 2.7 W during the search experiments with both systems, and 2.2 W using bluetooth. These results show that there is a substantial performance difference between a "black box" search engine and a state-of-the-art system. Nevertheless, more improvements are still needed to provide a truly responsive, interactive search experience.

In conclusion, our experiments show that low-power computing devices today are affordable and achieve sufficient performance to warrant more exploration of personal mobile digital libraries. This represents an interesting future research direction.

Acknowledgments: This work was supported by NSF under IIS-1218043 and Verisign. Any opinions, findings, conclusions, or recommendations expressed are those of the author and do not necessarily reflect the views of the sponsors.

4. REFERENCES

[1] R. Balani. Energy consumption analysis for bluetooth, wifi and cellular networks, 2007.

[2] C. Clarke, N. Craswell, and I. Soboroff. Overview of the TREC 2009 web track. *TREC*, 2009.

[3] A. Trotman, X.-F. Jia, and M. Crane. Towards an efficient and effective search engine. *SIGIR 2012 Workshop on Open Source Information Retrieval*, 2012.

[3] https://dumps.wikimedia.org/other/