Enhancing Web Page Skimmability

Chen-Hsiang Yu
MIT CSAIL
32 Vassar St
Cambridge, MA 02139
chyu@mit.edu

Robert C. Miller
MIT CSAIL
32 Vassar St
Cambridge, MA 02139
rcm@mit.edu

CHI'12, May 5–10, 2012, Austin, Texas, USA.
ACM 978-1-4503-1016-1/12/05.

Abstract

Information overload on the Web and limited reading time force users to skim read web pages. For non-native English readers, it is challenging to understand first-hand information written in English under time constraints. Traditional readability enhancement research has focused on enhancing reading comprehension and user satisfaction, but average reading times for non-native readers have remained the same or even worse. In this paper, we investigate useful techniques for readers when reading web pages under time constraints, i.e., having skim reading capability. We propose two techniques to help non-native readers to skim read web pages: (1) content spotlight, masking and filtering; and (2) semantic data extraction and in-place translation. Froggy GX is a prototype system that implements proposed techniques to provide skim-reading support for non-native readers.

Keywords

Skim reading, Froggy GX

ACM Classification Keywords

H.5.2 [Information Interfaces And Presentation]: User Interfaces - Interaction styles;

General Terms

Design, Measurement

Introduction

While there are about 328 million people in the world speaking English as their native language, there are 1.4 billion people speaking English as their second language. Although the Web offers an enormous set of reading materials, UNESCO's report in 2009 showed that more than 41% of web pages were written in English [6]. People speaking English as their second language (non-native readers) often need to read English web pages. For example, they might need to read first-hand information from English-speaking countries, such as the United States, or may have jobs that require them to understand English on the Web. In the past, reading English web pages has been identified as a difficult task for non-native readers, and researchers have tried to enhance web page readability for this group of users [7]. However, since there is excessive information to read on the Web, more and more people skim-read rather than read web pages in detail [1][3]. Unfortunately, skim reading on English web pages is not a common skill among non-native readers.

For decades, researchers have investigated how people read, but the study of skim reading is still in its infancy. The study done by Duggan et al. [2] show that native readers might use a certain kind of behavior, called *satisficing*, to read articles under time constraints. As they define it, satisficing is "reading through text until the rate of information gain drops below threshold and then skipping to the next section of text." However, it is not clear if the behavior of *satisficing* exists in non-native readers.

In this paper, we define *skimmability* as a combination of reading comprehension and user satisfaction under time constraints, which is different from the definition of readability defined by Yu et al. [7] To understand the difference in reading performance between native and non-native readers under time constraints, we conducted a preliminary study with six users (three native readers, three non-native readers) where each user read two articles. On average, there were 414 words in each article and each reader was given 30 seconds to read each article. The results showed that native readers are 20% better than non-native readers in reading comprehension under time constraints, as explained below.

In this paper, we focus on investigating useful techniques to enhance web page skimmability for non-native readers. There are two challenges in this research. First, although previous studies evaluated skim reading behavior for native readers, to the best of our knowledge there were no studies specifically done for non-native readers. Second, it is unknown what constitutes a good design for achieving the same skim reading effectiveness in non-native readers as native readers exhibit. We propose two techniques to enhance web page skimmability: (1) content spotlight, masking and filtering; and (2) semantic data extraction and in-place translation. Froggy GX is a prototype system that implements these two proposed techniques to provide skim-reading support for non-native readers.

Preliminary Study

To the best of our knowledge, there is limited work studying the difference between native and non-native readers in skim reading. We conducted a preliminary study to understand this issue. Six users (three native speakers and three non-native speakers) were recruited to join the study and, on average, they were 26 years old. There was only one condition investigated in the study: reading two articles (one a TOEFL article

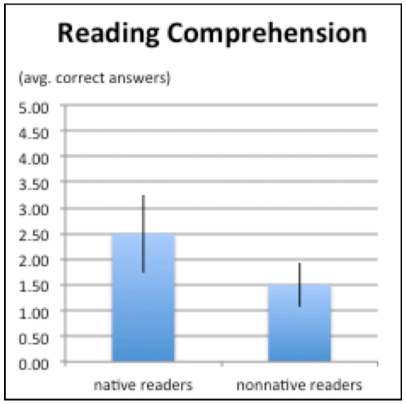


Figure 1. Reading Comprehension – it is measured by number of correct answer to a 5-question quiz.

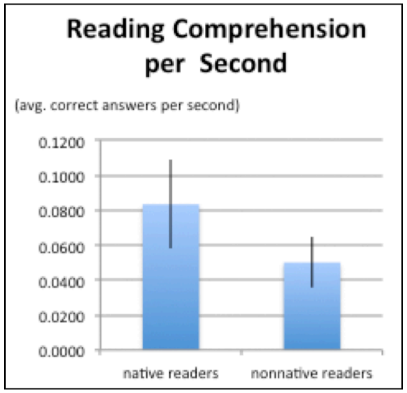


Figure 2. Reading Comprehension per Second.

and the other a GRE article), with a 30-second time limit per article. On average, three non-native readers have learned English for 13.6 years and spent 2.83 years living in the United States. After reading each article, we asked each user to answer five single selection questions, which are used to measure the user's reading comprehension. At the end of the reading, we interviewed each user with a few proposed changes to both of user interface and the content to understand their opinions.

The study measured users' reading comprehension and reading comprehension per unit time. For analysis, we categorized the results of six users into two groups: native readers and non-native readers. Figure 1 shows the analysis of the results in reading comprehension, which was measured by the number of correct answers in a five-question quiz about the article. The results indicate readers' reading comprehension is 20% better than non-native readers' under 30 seconds time constraint in reading.

Figure 2 shows the analysis of reading comprehension per second between the two groups. The results reveal that the group of native readers has a 66.67% higher reading comprehension per second than the group of non-native readers.

After reading two articles, we presented four new content transformations on paper to have readers' feedbacks. The four presented formats are: (1) key sentence highlighting, (2) key sentences shown as bullet points in a popup window, (3) key sentences translated into a reader's native language, and (4) presenting translated key sentences at the beginning of the article. In sum, all non-native readers like to read filtered text, such as key sentences, which allows them to finish reading under time constraints and feel comfortable reading filtered text (key sentences) written in their native language

The results of this preliminary study indicate that there exists a difference between native and non-native speakers in reading comprehension under time constraint. In addition, non-native readers prefer to read filtered information, such as key sentences, and feel comfortable to read this information written in their native language. Non-native readers also reported that they sometimes search for a certain keyword on the content and read the enclosing sentence.

System Design

Based on the results and feedbacks of the preliminary study, we believe that it is important to enhance web page skimmability for non-native readers. Froggy GX is a prototype system developed to enhance web page skimmability for non-native readers.

Design Principles

There are a few design principles behind the system. First, a simple interaction is necessary. Because users read web pages under time constraints, they do not have enough time to control widgets for reading, such as buttons, sliders and textbox. Therefore, the design of the tool needs to be simple. Secondly, because non-native readers' reading speed is slow in general [7], the tool needs to provide a good reading strategy for readers to finish reading within the time constraints, such as providing filtered information. Moreover, as we learned, because non-native readers feel comfortable reading content written in their native language we can take this idea into consideration. However, we should not overuse it because, as we observed, current machine translation is only good enough for translating individual words and phrases, but not the entire article.

Lastly, non-native readers tend to search for a keyword and read the enclosing sentence, but time constraints might have an impact on choosing a keyword. Preprocessing the content for extracting semantic data might help. Based on these three principles, we developed a prototype system and conducted several iterations with five non-native users. The final prototype system, named Froggy GX, is a new Firefox extension that has two features: (1) content spotlight, masking and filtering; and (2) semantic data extraction and in-place translation.

User Interface Design

The design of Froggy GX includes only one button for the skim mode trigger and one drop down menu for a user to select his native language. When a reader clicks on the Skim Mode button, the drop down menu shows up. (Figure 3) Currently, the system supports English, Chinese, Korean and Japanese.

As Figure 4 illustrates, when a reader clicks on Skim Mode button, Froggy GX will transform the original page (Figure 4-(a)) to a new format for skim reading purposes (Figure 4-(b)). Three actions are executed when the Skim Mode button is clicked. First, the tool spotlights the main reading area and masks unimportant content. Second, the whole text context of a web page is extracted and analyzed for semantic data extraction. Third, all key sentences are highlighted with partial translation support. The user can click on a sentence to have in-place translation support and double click on a paragraph to expand it into Jenga format dynamically [7].

Content Spotlight, Masking and Filtering

When a web page is loaded, Froggy GX starts a sequence of actions in the background to prepare a skim reading environment, including parsing the content, tagging all paragraphs and sentences, and calculating an absolute position and dimension of each paragraph. Based on this information, Froggy GX can find a container node for spotlight purposes. The container node covers the dimensions of all paragraph nodes and is the closest node to all paragraph nodes in the DOM tree. Froggy GX creates a new <SPAN> node with a transparency value = 0.1 to be a mask and attaches it to the end of the <BODY> node. In order to have a spotlight effect, Froggy GX increases the z-order of the container node and sets its background color to white.

To have a clean reading environment, the GX uses a technique similar to existing systems (e.g. Safari 5 Reader and Firefox Readability and Froggy [7] extensions) to detect possible annoying content and make it proportionally smaller and more transparent. Next, we use static approach, including human-rated data, to find the salient sentences of the whole article and make all non-salient sentences transparent. A few well-known existing algorithms, including Clair libraries [6], is under evaluation.

Semantic Data Extraction and In-Place Translation

After Froggy GX spotlights the web page, it also compiles all extracted data and uses the OpenCalais Web Service [5] to perform content analysis and semantic data extraction. The OpenCalais Web Service provides a feature to annotate a given content and return it with semantic data, such as entities (e.g., company, country and people), events or facts (e.g., acquisition, alliance and bankruptcy).

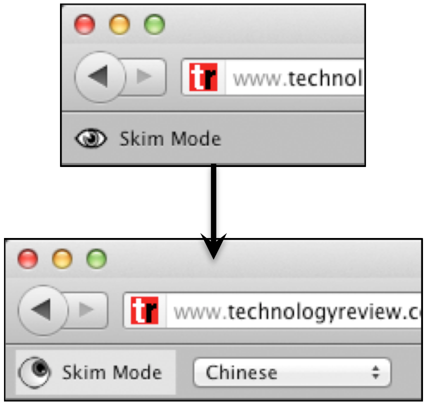
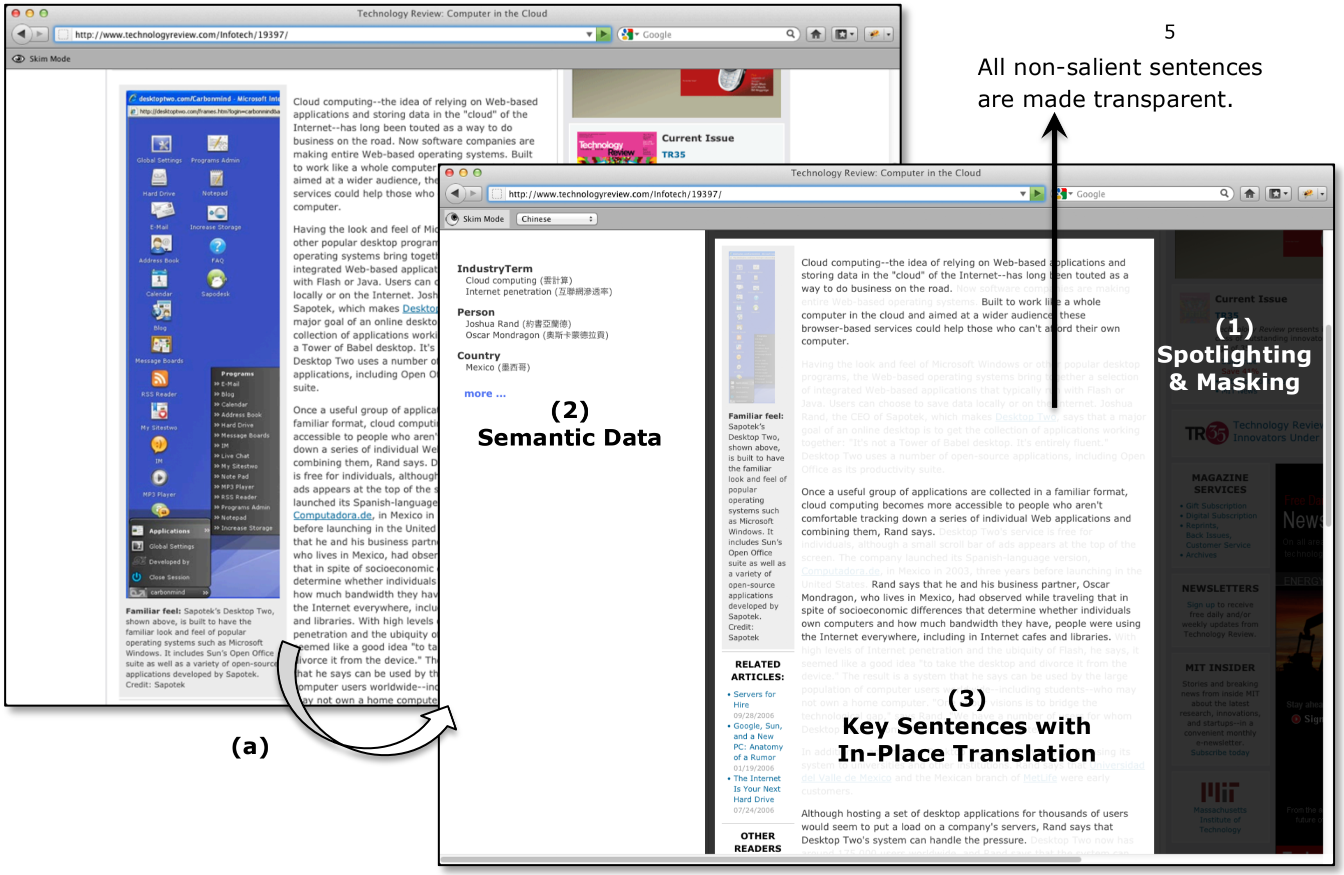


Figure 3. Froggy GX - an extension of Firefox browser. (Left: Default state; Right: Skim mode is triggered)


All non-salient sentences are made transparent.

(1) Spotlighting & Masking

(2) Semantic Data

(3) Key Sentences with In-Place Translation

(a)

(b)

Technology Review: Computer in the Cloud

http://www.technologyreview.com/Infotech/19397/

Skim Mode · Chinese

IndustryTerm
Cloud computing (雲計算)
Internet penetration (互聯網滲透率)

Person
Joshua Rand (約書亞蘭德)
Oscar Mondragon (奧斯卡蒙德拉貢)

Country
Mexico (墨西哥)

more ...

Cloud computing--the idea of relying on Web-based applications and storing data in the "cloud" of the Internet--has long been touted as a way to do business on the road. Now software companies are making entire Web-based operating systems. Built to work like a whole computer in the cloud and aimed at a wider audience, these browser-based services could help those who can't afford their own computer.

Having the look and feel of Microsoft Windows or other popular desktop programs, the Web-based operating systems bring together a selection of integrated Web-based applications that typically run with Flash or Java. Users can choose to save data locally or on the Internet. Joshua Rand, the CEO of Sapotek, which makes Desktop Two, says that a major goal of an online desktop is to get the collection of applications working together: "It's not a Tower of Babel desktop. It's entirely fluent." Desktop Two uses a number of open-source applications, including Open Office as its productivity suite.

Once a useful group of applications are collected in a familiar format, cloud computing becomes more accessible to people who aren't comfortable tracking down a series of individual Web applications and combining them, Rand says. Desktop Two's service is free for individuals, although a small scroll bar of ads appears at the top of the screen. The company launched its Spanish-language version, Computadora.de, in Mexico in 2003, three years before launching in the United States. Rand says that he and his business partner, Oscar Mondragon, who lives in Mexico, had observed while traveling that in spite of socioeconomic differences that determine whether individuals own computers and how much bandwidth they have, people were using the Internet everywhere, including in Internet cafes and libraries. With high levels of Internet penetration and the ubiquity of Flash, he says, it seemed like a good idea "to take the desktop and divorce it from the device." The result is a system that he says can be used by the large population of computer users worldwide--including students--who may not own a home computer. "One of our visions is to bridge the technological gap," says Rand. "We have a number of users for whom Desktop Two ...

In addition ... Desktop Two ... licensing its system to universities and other institutions. Rand says that Universidad del Valle de Mexico and the Mexican branch of MetLife were early customers.

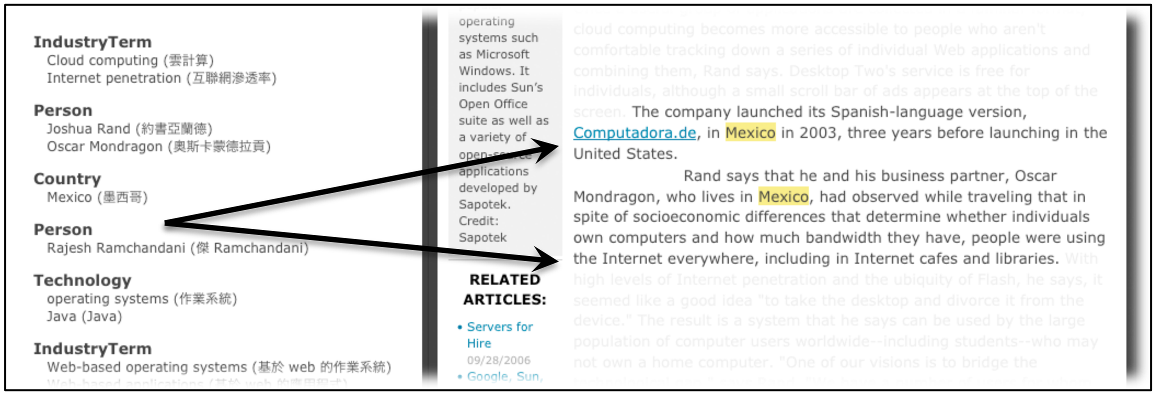
Although hosting a set of desktop applications for thousands of users would seem to put a load on a company's servers, Rand says that Desktop Two's system can handle the pressure. Desktop Two now has around 175,000 users worldwide, and Rand says that the system can

Familiar feel: Sapotek's Desktop Two, shown above, is built to have the familiar look and feel of popular operating systems such as Microsoft Windows. It includes Sun's Open Office suite as well as a variety of open-source applications developed by Sapotek. Credit: Sapotek

RELATED ARTICLES:
- Servers for Hire 09/28/2006
- Google, Sun, and a New PC: Anatomy of a Rumor 01/19/2006
- The Internet Is Your Next Hard Drive 07/24/2006

OTHER READERS

Figure 4. Using Froggy GX to transform the original web page into a format that is easier to skim read for non-native readers: (a) original web page, and (b) a transformed format.


IndustryTerm
Cloud computing (雲計算)
Internet penetration (互聯網滲透率)

Person
Joshua Rand (約書亞蘭德)
Oscar Mondragon (奧斯卡蒙德拉貢)

Country
Mexico (墨西哥)

Person
Rajesh Ramchandani (傑 Ramchandani)

Technology
operating systems (作業系統)
Java (Java)

IndustryTerm
Web-based operating systems (基於 web 的作業系統)

operating systems such as Microsoft Windows. It includes Sun's Open Office suite as well as a variety of open-source applications developed by Sapotek. Credit: Sapotek

RELATED ARTICLES:
- Servers for Hire 09/28/2006
- Google, Sun,

cloud computing becomes more accessible to people who aren't comfortable tracking down a series of individual Web applications and combining them, Rand says. Desktop Two's service is free for individuals, although a small scroll bar of ads appears at the top of the screen. The company launched its Spanish-language version, Computadora.de, in Mexico in 2003, three years before launching in the United States.

Rand says that he and his business partner, Oscar Mondragon, who lives in Mexico, had observed while traveling that in spite of socioeconomic differences that determine whether individuals own computers and how much bandwidth they have, people were using the Internet everywhere, including in Internet cafes and libraries. With high levels of Internet penetration and the ubiquity of Flash, he says, it seemed like a good idea "to take the desktop and divorce it from the device." The result is a system that he says can be used by the large population of computer users worldwide--including students--who may not own a home computer. "One of our visions is to bridge the

Figure 5. Clicking an item on the floating menu and showing corresponding sentences on the reading content.

However, the amount of returned data varies. In order to prevent information overload, which makes reading worse, we sort the data based on its relevance scores and provide selected entities at the beginning. We also offer an option at the end to expand the information.

All semantic data, including entities, events and facts, are grouped together by their categories, such as Joshua Rand and Oscar Mondragon in the Person category. The semantic data are implemented as a floating menu that can dynamically move along with the reader's scrolling. In addition, each data point has a translation enclosed within parentheses at the end, and the native language selected in the drop down menu decides the translation language. Froggy GX uses Microsoft Translator API [4] to perform the data translation. When a reader clicks on a specific data (entity, event or fact) on the floating menu, all the sentences containing this data will be emphasized, and the data is marked in yellow. Furthermore, if sentences that contain this highlighted data are connected sequentially, they will be separated as the Jenga format [7], which was invented to provide a better readability for non-native readers. (Figure 5)

Conclusion and Future Work

Reading web pages under time constraints is a common practice, but research into skim reading behavior on the Web is still in its infancy. Although Duggan et al.'s work [1][2][3] have presented possible strategies used by native readers, we are still not clear if these practices are also useful for non-native readers. In this research, we not only conducted a preliminary study to understand the difference between native readers and non-native readers in reading under time constraints, but we also investigate the possible solutions for non-native to use when reading web pages. Froggy GX is a Firebox browser extension that implements the ideas we propose: (1) content spotlight, masking and filtering, and (2) semantic data extraction and in-place translation. We are working on iterations and planning a user study for the tool.

Acknowledgements

We thank valuable suggestions from Eunsuk Kang and MIT UID group members. This work is supported in part by Quanta Computer as part of the TParty project.

References

[1] Duggan, G.B. and Payne, S.J. How much do we understand when skim reading? *CHI'06 Extended Abstracts on Human Factors in Computing Systems – CHI'06*, ACM Press, pp. 730-735, 2006.

[2] Duggan, G.B. and Payne, S.J. Skim Reading by Satisficing: Evidence from Eye Tracking. *In Proceedings of CHI2011*, ACM Press, pp. 1141-1150, 2011.

[3] Duggan, G.B. and Payne, S.J. Text Skimming: The Process and Effectiveness of Foraging Through Text Under Time Pressure. *Journal of Experimental Psychology. Applied 15*, 3, pp. 228-242, 2009.

[4] Microsoft Translator API: http://www.microsofttranslator.com/dev/

[5] OpenCalais: http://www.opencalais.com/

[6] Pimienta, D. et al. Twelve years of measuring linguistic diversity in the Internet: balance and perspectives. Paris: UNESCO. http://unesdoc.unesco.org/images/0018/001870/187016e.pdf

[7] The Clair libraries: http://www.clairlib.org/index.php/Main_Page

[8] Yu, C.H. and Miller, R.C. Enhancing Web Page Readability for Non-native Readers. *In Proceedings of CHI2010*, ACM Press, 2010.