

---

# Responsive Snippets: Adaptive Skim-reading for Mobile Devices

**Luis A. Leiva**  
Sciling, SL  
46120 Valencia, Spain  
[name@sciling.com](mailto:name@sciling.com)

---

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

Copyright held by the owner/author(s).  
*MobileHCI '18 Adjunct*, September 3–6, 2018, Barcelona, Spain  
ACM 978-1-4503-5941-2/18/09.  
<https://doi.org/10.1145/3236112.3236159>

## Abstract

Responsive Snippets is the accompanying software of my recent work on responsive text summarization, an approach to web design aimed at allowing desktop web pages to be read in response to the size of the device a user is browsing with. Responsive Snippets allow designers to create HTML summaries quickly and effortlessly, by simply using CSS selectors and media queries. Responsive Snippets can be especially useful for news and blog sites, as a means to quickly allow users get a glimpse of the main body content, although they can be applied to any kind of HTML elements having text. The software is publicly available, so that others can build upon this work.

## Author Keywords

Responsive Web Design; Media Queries; Text Summarization; Graph algorithms; Skimming

## ACM Classification Keywords

H.5.2 [User Interfaces]: Screen design; H.3.3 [Information Search and Retrieval]: Information filtering

## Introduction

Users often do not read web pages; instead they scan them, trying to pick out a few sentences or even parts thereof to get the information they want [6, 16]. Clearly,

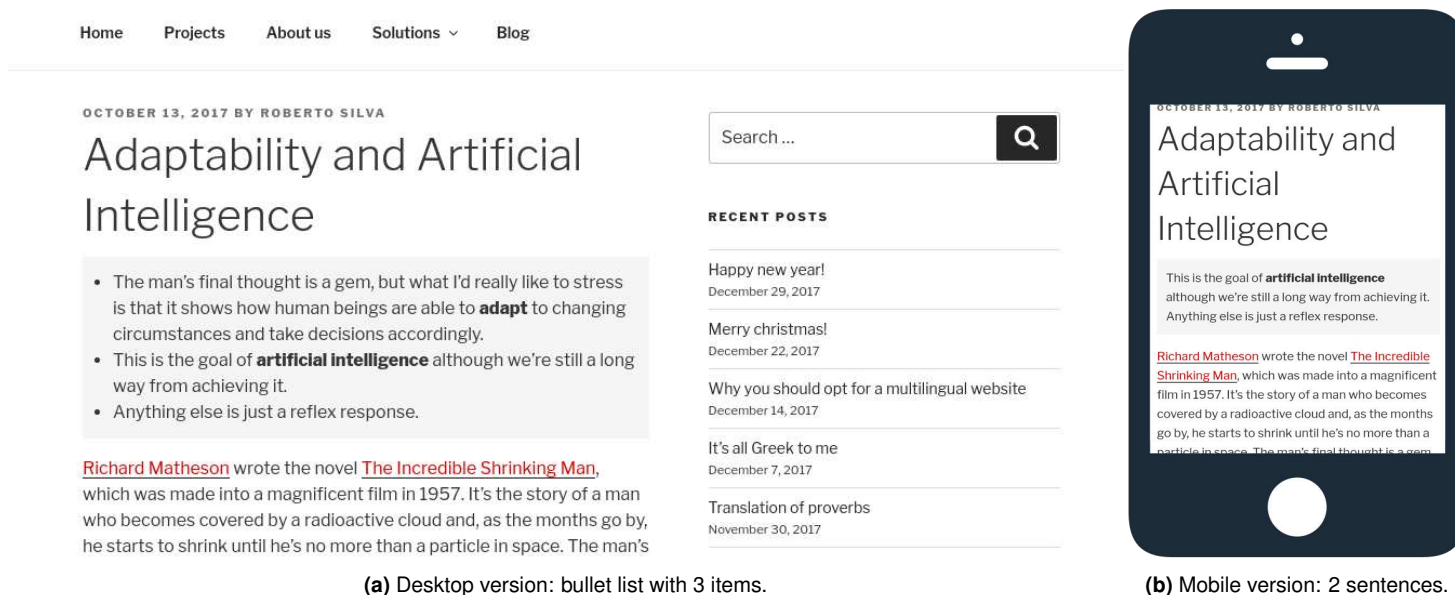


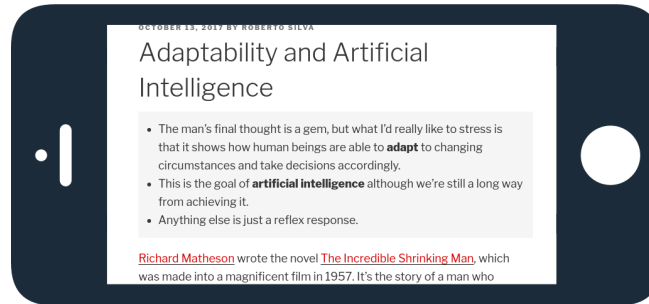
Figure 1: Implementation examples of Responsive Snippets on a live website.

nobody has time to read everything, yet we often have to make critical decisions based on what we are able to assimilate [14]. Text summarization is becoming indispensable for dealing with this problem, by distilling the most important information from a text source to produce a shorter, more manageable version. However, web designers do not have the right tools to achieve this goal at present. Rather, they usually delegate the implementation backbone to some skilled developer.

Responsive Web Design (RWD) suggests that design and development should respond to the user's behavior and environment based on screen size, platform, and orientation [8]. The practice consists of a mix of flexible grids and layouts, images, and an intelligent use of CSS media queries that can reformat web pages effortlessly (or automatically) without changing the original content itself or the HTML markup. This would eliminate the

need for a different design for each new browser-capable gadget on the market. Yet RWD is primarily targeted at modifying the graphical aspect of web pages rather than their contents. This work aims at bridging this gap.

In this paper, I present Responsive Snippets, the accompanying software of my recent work on responsive text summarization [13], an approach to web design aimed at allowing desktop web pages to be read in response to the size of the device a user is browsing with. As will be shown later, Responsive Snippets allow designers to create HTML summaries quickly and effortlessly, by simply using CSS selectors and media queries. The software is publicly available, so that others can build upon this work.



**Figure 2:** As soon as the mobile device goes into landscape mode, the user can read the same snippet shown in desktop mode (see Figure 1a), if the designer has decided to do so.

## Related Research

Responsive Snippets relate mostly to previous attempts to use text summarization for web browsing on handheld devices. For example, Lam and Baudisch [10] used text reduction heuristics to create readable overviews, and Buyukkokten et al. [5] investigated different methods for summarizing parts of web pages. Among other findings, text summaries provided significant improvements in terms of time and scrolls required to access the contents. Finally, Yu and Miller [17] developed a browser extension that transforms a web page into a format that is easier to skim-read for non-native readers.

In the past, researchers have proposed different ways of adapting web content, mostly involving end-user interaction. For example, Bolin and co-authors [4] developed a customization plugin for the Firefox browser that enabled a high-level scripting language, without having to access the source code of web pages. Kurniawan et al. [9] proposed to override the visual layer of a web page with custom CSS, although such updates

had to be performed by hand. Baudisch et al. [2] and Bila et al. [3] encouraged the user to actively modify the layout contents. Going further, Leiva [11, 12] leveraged implicit interactions (e.g., mouse movements or touch events) to automatically modify the visual appearance of web page elements.

Yet, none of these previous works have considered HTML content summarization from a RWD perspective. The closest attempt was a proof of concept which consisted in styling text paragraphs manually, by wrapping “unimportant” words or sentence parts in `<span>` elements that would be hidden on smaller screens.<sup>1</sup> This manual work is clearly too tedious to work at scale, since the text source has to be highly structured in order for it to be readable when parts of it are hidden on small devices. On the contrary, Responsive Snippets allow designers to achieve this goal effortlessly, by simply applying CSS selectors and media queries, as they would typically do in RWD.

## Implementation

Responsive Snippets follow an *extractive* summarization approach; i.e., they extract and present pieces of the original text, as opposed to the *abstractive* approach, where it is rephrased. The extractive variant has more sense for web contents, otherwise the web page could become confusing to the users. In fact, it has been shown that users generally prefer seeing summarized text in the form created by the author [1]. Additionally, extractive methods often perform better than abstractive methods according to several metrics [7].

Responsive Snippets implement various text summarization algorithms, among which TextRank [15]

<sup>1</sup>[http://www.frankieroberto.com/responsive\\_text](http://www.frankieroberto.com/responsive_text)

is the default and recommended one, since it is an unsupervised language-independent graph-based ranking model that produces summaries very similar to what humans would produce [15] and it is still considered one of the best performers overall [7].

Another important feature of Responsive Snippets is that they preserve the HTML markup of the original text source. This is accomplished via a simple string matching algorithm, although, to the best of my knowledge, no other text summarization technique achieves this effect.

The reader may consult the previous publication on responsive text summarization [13], the technique on which Responsive Snippets are based, to get into the technical details as well as an empirical evaluation in terms of both feasibility and performance.

## Usage

Responsive Snippets rely on traditional CSS selectors and media queries, so they can be easily used by web designers and developers. Figure 3 provides the CSS code shown in Figure 1.

Any HTML element that may contain text can be summarized using the following CSS properties:

- **text-summary** (*float|percentage|string*) indicates the length of the summarized text. Default: *none* (the source element is not summarized).
- **text-summary-target** (*string*) indicates the target element that will display the text summary. Default: *none* (targets the source element itself).
- **text-summary-algorithm** (*string*) indicates the summarization algorithm. Default: *textrank*.

```
/* CSS selector. Targets any device. */
.single-post article .entry-content {
  --text-summary: 3 sentences;
  --text-summary-list: unordered;
  --text-summary-target: #tldr-snippet;
}

/* CSS media query. Targets mobile devices. */
@media (max-width: 640px) {
  .single-post article .entry-content {
    --text-summary: 2 sentences;
    --text-summary-list: none;
  }
}

@media (max-width: 640px)
and (orientation: landscape) {
  .single-post article .entry-content {
    --text-summary: 3 sentences;
    --text-summary-list: unordered;
  }
}
```

**Figure 3:** Working examples from Figure 1. Since the **text-summary\*** properties are non-standard, they must be prefixed with two dashes (--).

- **text-summary-list** (*string*) displays the sentences as a list, either ordered or unordered (itemized). Default: *none* (displays plain HTML).
- **text-summary-skip-min-words** (*number*) filters out sentences having less words than the given number. Default: *none* (no sentences are filtered).
- **text-summary-skip-max-words** (*number*) filters out sentences having more words than the given number. Default: *none* (no sentences are filtered).

The reader may consult <https://luis.leiva.name/rts/> for more information regarding usage details and code documentation, including minimal working examples for each of the available properties.

## Validation

Responsive Snippets were brought to production on a corporate blog website on March 1, 2018. In this section, I validate its impact according to the Google Analytics reports pertaining two months of usage, both before and after having included Responsive Snippets on the blog website.

Table 1 shows the results. The “Before” condition includes visitors from January 1 to February 28 and the “After” condition includes visitors from March 1 to April 30.

Metric	Before	After	Diff
Unique Users	434	597	↑ 163
New Users	422	585	↑ 163
Sessions	529	714	↑ 185
Num. Sessions per User	1.22	1.20	↓ 0.02
Pageviews	917	1,388	↑ 471
Pages/Session	1.73	1.94	↑ 0.21
Avg. Session Duration	01:11	01:30	↑ 00:29
Bounce Rate (%)	70.51	67.51	↓ 3

**Table 1:** Validation results on a production website, both before and after having included Responsive Snippets.

As can be noticed, users now spend more on time on the website and visit more pages on average. The bounce rate (percentage of visitors who navigate away after viewing only one page) has improved as well, suggesting that now blog visitors can easily go straight to the posts they like to read the most.

It is important to notice that no new blog posts have been published on the corporate blog since December 2017, and the only change implemented on the website has been the Responsive Snippets software itself. Therefore, the results support the Responsive Snippets technique and serve as an in-the-wild validation.

## Conclusion

Responsive Snippets are the first attempt to create HTML summaries from a truly RWD perspective. The value of this work lies in the fact that it is possible to create summarized contents with traditional CSS selectors and media queries. This allows web designers to display different summaries of a body text at different levels of granularity. What is more important, summarization happens automatically, without changing the original content itself or the HTML markup of the original web page.

The software is publicly available at <https://luis.leiva.name/rts/> so that others can build upon this work. Responsive Snippets can be easily incorporated into production-ready websites, and in fact they are currently being used on a corporate blog site. I hope this work will inspire others to build better tools to deliver web contents that are easier to assimilate for their visitors.

## REFERENCES

1. Aris Anagnostopoulos, Andrei Z. Broder, Evgeniy Gabrilovich, Vanja Josifovski, and Lance Riedel. 2011. Web Page Summarization for Just-in-Time Contextual Advertising. *ACM Trans. Intell. Syst. Technol.* 3, 1, Article 14 (2011), 32 pages.
2. Patrick Baudisch, Xing Xie, Chong Wang, and Wei-Ying Ma. 2004. Collapse-to-zoom: Viewing

- Web Pages on Small Screen Devices by Interactively Removing Irrelevant Content. In *Proc. ACM Symposium on User Interface Software and Technology (UIST '04)*. 91–94.
3. Nilton Bila, Troy Ronda, Iqbal Mohamed, Khai N. Truong, and Eyal de Lara. 2007. PageTailor: Reusable End-user Customization for the Mobile Web. In *Proc. Intl. Conf. on Mobile Systems, Applications and Services (MobiSys '07)*. 16–29.
  4. Michael Bolin, Matthew Webber, Philip Rha, Tom Wilson, and Robert C. Miller. 2005. Automation and Customization of Rendered Web Pages. In *Proc. ACM Symposium on User Interface Software and Technology (UIST '05)*. 163–172.
  5. Orkut Buyukkokten, Hector Garcia-Molina, and Andreas Paepcke. 2001. Seeing the Whole in Parts: Text Summarization for Web Browsing on Handheld Devices. In *Proc. Intl. Conf. on World Wide Web (WWW '01)*. 652–662.
  6. Geoffrey B. Duggan and Stephen J. Payne. 2011. Skim Reading by Satisficing: Evidence from Eye Tracking. In *Proc. SIGCHI Conf. on Human Factors in Computing Systems (CHI '11)*. 1141–1150.
  7. Max Grusky, Mor Naaman, and Yoav Artzi. 2018. NEWSROOM: A Dataset of 1.3 Million Summaries with Diverse Extractive Strategies. In *Proc. Conf. North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL '18)*. 1–12.
  8. Kayla Knight. 2011. Responsive Web Design: What It Is And How To Use It. Smashing Magazine. (2011). Available at <https://smashingmagazine.com>.
  9. S. H. Kurniawan, A. King, D. G. Evans, and P. L. Blenkhorn. 2006. Personalising Web Page Presentation for Older People. *Interact. Comput.* 18, 3 (2006), 457–477.
  10. Heidi Lam and Patrick Baudisch. 2005. Summary Thumbnails: Readable Overviews for Small Screen Web Browsers. In *Proc. SIGCHI Conf. on Human Factors in Computing Systems (CHI '05)*. 681–690.
  11. Luis A. Leiva. 2011. Restyling Website Design via Touch-based Interactions. In *Proc. Intl. Conf. on Human Computer Interaction with Mobile Devices and Services (MobileHCI '11)*. 599–604.
  12. Luis A. Leiva. 2012. Automatic Web Design Refinements Based on Collective User Behavior. In *Proc. Extended Abstracts on Human Factors in Computing Systems (CHI EA '12)*. 1607–1612.
  13. Luis A. Leiva. 2018. Responsive Text Summarization. *Inf. Process. Lett.* 130, 1 (2018), 52–57.
  14. Inderjeet Mani. 1999. *Advances in Automatic Text Summarization*. MIT Press.
  15. Rada Mihalcea and Paul Tarau. 2004. TextRank: Bringing Order into Texts. In *Proc. Empirical Methods in Natural Language Processing (EMNLP '04)*. 404–411.
  16. Jakob Nielsen and John Morkes. 1997. Concise, SCANNABLE, and Objective: How to write for the Web. Nielsen Norman Group. (1997). Available at <https://nngroup.com>.
  17. Chen-Hsiang Yu and Robert C. Miller. 2012. Enhancing Web Page Skimmability. In *Proc. Extended Abstracts on Human Factors in Computing Systems (CHI EA '12)*. 2655–2660.