HiText: Text Reading with Dynamic Salience Marking

Qian Yang
Tsinghua University
laraqianyang@gmail.com

Yong Cheng
Tsinghua University
chengyong3001@gmail.com

Sen Wang
Carnegie Mellon University
senw1@cs.cmu.edu

Gerard de Melo
Rutgers University
gdm@demelo.org

ABSTRACT

The staggering amounts of content readily available to us via digital channels can often appear overwhelming. While much research has focused on aiding people at selecting relevant articles to read, only few approaches have been developed to assist readers in more efficiently reading an individual text. In this paper, we present HiText, a simple yet effective way of dynamically marking parts of a document in accordance with their salience. Rather than skimming a text by focusing on randomly chosen sentences, students and other readers can direct their attention to sentences determined to be important by our system. For this, we rely on a deep learning-based sentence ranking method. Our experiments show that this results in marked increases in user satisfaction and reading efficiency, as assessed using TOEFL-style reading comprehension tests.

Keywords

Text visualization; Text skimming; Natural language semantics

1. INTRODUCTION

The Challenge of Modern Information Perusal. The increasing digitization of the world has radically transformed the way we consume information. Historically, the invention of the modern printing press enabled the spread of gazettes, newspapers, and magazines, which in turn led to massive information dissemination across space and time. The societal impact of this was so profound that freedom of the press has come to be considered a fundamental human right in many societies. Whilst until recently the majority of people would rely on at most a single daily newspaper to remain up to date, in today's world, information is spread in real-time across the globe. The resulting non-stop 24/7 stream of new articles, papers, and other documents far outpaces our human ability to peruse all relevant information. Hence, students need vital new skills to avoid what has been called "info-besity", perhaps in reference to the tolls of a time-consuming overconsumption of such content, which may have harmful effects on their lives and goals. New skills and tools are hence crucial for them to discern relevant insights in this data deluge.

One important strategy for helping students cope with information inundation is to draw on macro-level content selection at the level of documents, articles, or similar macroscopic content units. Given the large amounts of disparate content providers vying for our attention, aggregators such as Google News provide such selections, as do recommendation engines and document clustering tools. Unfortunately, this strategy alone is insufficient. While more modest-sized and perhaps even personalized selections of articles are certainly helpful, new forms of information dissemination such as blogging and posting on social media have democratized publishing, exacerbating the challenges. This has progressed to an extent that it is now typically futile to attempt to stay abreast of even just the narrower set of subject matters that are genuinely of interest to us. There is a steady supply of fresh material even about the most arcane choices of topics.

Overview. In this paper, we focus on a second, much less studied aspect of content selection, namely the micro-level selection of salient pieces of information *within* a given document. Although reading text is one of the most common uses of computing devices, only few techniques have been proposed to assist in making this more efficient.

Skimming is a reading technique that involves quickly glancing over a text and only reading selected parts of it fully. This may allow us to grasp the essence of a text in a fraction of the time that it takes for a regular line-by-line reading of the same text. The notion of speed reading is heavily based on skimming, combined with subvocalization elimination.

Despite the obvious benefits of being able to read more efficiently, only few people are effective speed readers. In fact, language learners are often held up by difficult words or sentences and may find themselves giving up, rather than seeking to drill down further only on those sentences that are crucial to get the gist of the text. Even for proficient readers, more often than not, skimming takes the form of glancing at somewhat randomly selected parts of the text. While in most cases this likely still will turn out to be more efficient than reading a text thoroughly, the haphazard nature of this process may lead to a hit-or-miss form of skimming. Fortunately, for electronic media, we have the opportunity of providing additional guidance with regard to the key parts of a given article.

WWW'17 Companion, April 3–7, 2017, Perth, Australia.
ACM 978-1-4503-4914-7/17/04.
http://dx.doi.org/10.1145/3041021.3054168

In this paper, we present HiText, a novel approach for supporting the reading process by specially marking on the screen those parts of the text that are likely to be salient. We rely on natural language analysis based on deep learning representations to identify important sentences and embed additional highlighting into the rendering of the original document. In order to remain unobtrusive and dynamic, only top-ranked key sentences are generally highlighted. The remaining text is selectively marked in further detail on demand, based on pointing device input, in accordance with the meta-guiding process of the reader. Our experiments show that our approach results in significant gains in efficiency and increased user satisfaction.

2. RELATED WORK

Given our earlier remarks about the helpful but insufficient nature of macro-level content selection at the level of choosing important documents, we now review in further detail some of the micro-level strategies for making the perusal of an individual document more efficient.

Excerpts and Summaries. In many information systems, a simple way of reducing the information load for a given document is to simply show a short excerpt of the original input text. This option, also known as *snippet generation*, is often invoked in text retrieval engines when providing search results in the form of a ranked list of relevant documents. For each document, a short snippet is displayed, which aids the user in judging whether a given document is likely to indeed satisfy a given information need. If this is the case, however, the user normally will have to select the link and consider the full text of the document.

Rather than showing a short excerpt reflecting just one single or in some cases two or three individual parts of the original document, a more thorough understanding can be achieved by supplying a concise summary that provides a brief sketch of the entire document. While certain genres such as scientific articles already typically come with short abstracts, and some online posters have resorted to providing a quick too-long-didn't-read (TL;DR) version of their message, the majority of online text does not come with pre-written summaries.

In natural language processing, a number of text summarization algorithms have been developed. For lack of space, we refer interested readers to one of several available surveys of this area [6, 19]. Related techniques can also be used to assess the quality of a given summary [31], or to generate summaries of structured data [29] or of videos [15]. Summaries are the most direct form of addressing the goal of providing a restricted amount of information to the reader and are often appropriate for mobile devices with limited screen sizes. The Yahoo! News Digest mobile app draws on technology initially developed by the start-up company Summly, acquired by Yahoo Inc. in 2013. The specific sentence ranking method that we rely on in this paper is a form of representation learning for natural language [14, 3, 16, 7].

Unfortunately, for many use cases, receiving just a summary is unsatisfactory. For one, natural language understanding is an AI-hard task, and automatic summarization systems are known to make mistakes that result in incoherent output summaries, sometimes even distorting the original message [10]. Moreover, different readers may exhibit different interests, which may also evolve dynamically during the process of reading [23]. Despite being very useful in a number of circumstances, static limited-length summaries thus do not do justice to the dynamics of attention and interest levels during the reading process. For instance, they do not provide a natural way for the reader to spontaneously decide to drill down on particular parts of the original text. Additionally, certain elements that have shown to be vital for comprehension, such as text structure [2], formatting, and figures and tables [32], may be lost when merely providing a short summary.

Reading and Comprehension. Several studies have shed further light on the way readers read an original text without any form of marking or highlighting. Kingery & Furuta [13] investigated the effects of font typeface and point size as well as screen resolution and monitor size on the legibility of text. Walsh [28] identified notable differences in the way readers read texts when comparing different media forms. Pitler [22] presented an automated method for assessing the readability (in terms of text quality) of a given text, which strongly correlates with human judgments of readability. Regarding skimming, Yi [32] conducted a study in which students were made to skim 100 CHI papers in a short amount of time (spending around 4.3 minutes per paper on average, for a total of around 7 hours). Duggan & Payne [8] used eye tracking to study the way readers skim text. Their study revealed three forms of behaviour that readers seem to combine when skimming under time pressure: (1) scanning, (2) satisficing, i.e., skipping ahead, possibly to the next paragraph, once the information gain drops below a threshold, and (3) sampling. These results suggest that salience-based marking of the form presented here could lead to gains in efficiency.

Reading Assistance. The approach we follow is to highlight the key sentences in a text. The Semantize system [30] marked positive and negative sentiment words by underlining them with different colors. A study by de Paiva et al. [21] developed a way of highlighting text so as to distinguish different kinds of word types and named entities. The ScentHighlights system [4] marked sentences and keywords relevant to a given user query in a number of different colors. This solution thus applies when a reader seeks very specific information and can express this information need using a keyword query. The closest system we are aware of is the Nestor Highlighter Extension[1], a web browser plugin that first seeks to identify important sentences in a text and then highlights these with a yellow background. The main difference is that this sort of approach requires a hard binary choice between important and unimportant sentences. In our experiments, we show that an approach in which a graded view is provided dynamically on demand can be superior.

Another important research avenue is to devise specialized solutions for people with particular conditions. Ahmed et al. [1] conducted a study in which blind individuals used a speech synthesis-powered screen reading tool that allowed them to switch back and forth between a human-written summary and the original text. This solution allows for dynamic interactions. Due to the nature of the interface, it is centered around linear movements in the text. Yong et al. [33] presented reading aids for readers suffering from posterior cortical atrophy, which severely impacts text reading

[1] `http://www.nestorlabs.com/`

abilities. As their main challenge is the spatial layout of the text, the reading aids involved presenting either just individual words or just two words at a time to the reader.

3. DESIDERATA AND DESIGN

Considering the goal of aiding readers in skimming more efficiently while maintaining a natural unobtrusive interface, our analysis led us to derive the following requirements.

1. **Salience-Based Discrimination:** Given that our goal is to enable the reader to more quickly discern salient information from a text, we need to offer some way of distinguishing key information from less important parts of the text.

2. **Graded Salience:** Salience is an inherently graded property – The salience of a sentence may be more or less pronounced, so coercing it so as to establish a binary decision between salient and non-salient parts of the input may result in somewhat arbitrary choices. Different readers may desire reading the text at different levels of detail. Indeed, even a single reader may wish to read different parts of the same text at different levels of detail.

3. **Dynamic Interface:** Given the dynamically evolving nature of human interest during reading [23], the system is not fully able to predict in advance which pieces of information should best be presented to the reader. While a reader may initially only need the general gist of a text, this can easily spark further interest and curiosity, leading to a desire to drill down further on certain parts of the text. Moreover, current state-of-the-art methods in natural language understanding in general and text summarization in particular are prone to errors. It is thus imperative to enable dynamic exploration of the text. Among other things, this entails retaining access to the entirety of the original input text.

4. **Ergonomic Unobtrusiveness:** Reading is an activity that students and other readers engage in for significant amounts of time, often several hours per day. It is thus quite critical that the additional information be presented unobtrusively, while also avoiding a significant increase in eye fatigue, a problem that is said to lead to millions of eye examinations every year. While helpful for increased discriminability, overly colorful visualizations may hence not be desirable. Additionally, we aim at an interface that feels natural to users that are accustomed to and enjoy reading articles online.

4. THE HITEXT METHOD

Drawing on our analysis, we have developed HiText as a new method of providing text to a reader. As illustrated in Figure 1, our system consists of a document analysis method to extract salient sentences within the text, which we describe first. Subsequently, we detail our user interface, which highlights these sentences in accordance with their salience, yet seeks to remain unobtrusive and dynamic.

4.1 Document Analysis

Our system first reads the original document file. Our current implementation assumes a document provided in HTML, from which the text is extracted and then relevant units are scored.

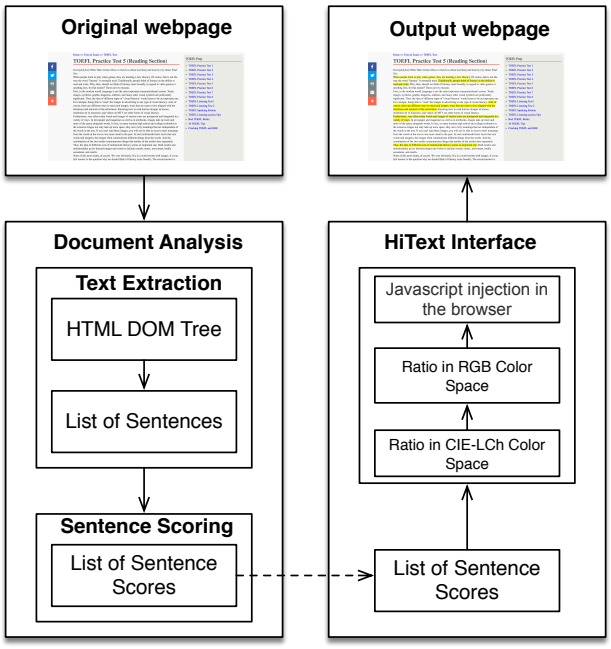

Figure 1: System overview

Text Extraction. The first step is to parse the input document with graceful handling of invalid HTML, similar to how modern Web browsers are able to cope with bad HTML code. Then, within the resulting HTML DOM tree, we generate a plaintext conversion of each text paragraph. However, since not all text elements in a given HTML page belong to the main text, we exclude navigational elements from consideration, seeking to identify them using a stop word ratio-based heuristic [25]. Finally, each paragraph is further split using a simple sentence splitting heuristic to obtain lists of sentences. In particular, we search for full stops but ignore those that appear to belong to abbreviations (such as *e.g.* or *U.K.*), which tend not to indicate sentence boundaries.

Sentence Scoring. Subsequently, each extracted sentence s is analysed and assessed using a deep learning-based scoring technique to produce a salience score $\sigma(s)$. As a first step, we generate a vector space representation of each sentence using a recurrent neural network (RNN) architecture trained without direct supervision, instead relying on sentence proximity as a proxy [14]. Recurrent neural networks are a form of neural network that can operate on variable-length sequential data. The basic ingredient of a recurrent neural network is a function f that computes a new hidden state vector $\mathbf{h}_i$ given the previous hidden state vector $\mathbf{h}_{i-1}$ and a new input vector $\mathbf{x}_i$ as $\mathbf{h}_i = f(\mathbf{h}_{i-1}, \mathbf{x}_i)$. If we assume every new input vector represents a word from the sequence of words in a sentence, then RNNs can be used to generate hidden vectors that represent sentences. Such sentence representations have also proven useful in machine translation [12, 24], significantly outperforming previous state-of-the-art systems on many language pairs [18].

In our case, the recurrent units of the RNN, i.e. the function f, consists of Gated Recurrent Units (GRUs) [5], which in turn were inspired by Long Short Term Memory units [11].

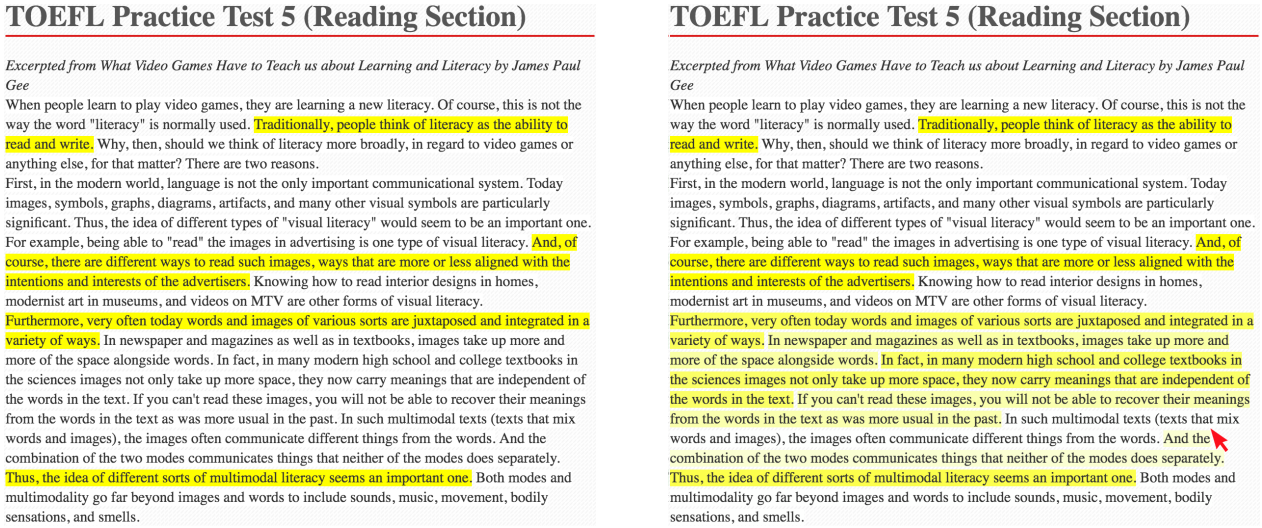

Figure 2: Left – Highlighting of top-k sentences selected by our method in top-k mode (left). Top-k sentences remain highlighted independent of pointing device movements. Right – HiText's on-demand graded highlighting of sentences (HiText reading mode). When the mouse hovers over any area in a paragraph, all sentences in that paragraph with sufficiently high scores are temporarily highlighted, with graded color intensities.

Initially, such a network would use random word vector representations and weights. However, when trained using gradient descent methods with backpropagation for several weeks on massive volumes of text, with the objective of obtaining vector representations that are predictive of vectors for its immediate neighbour sentences, then the word vectors and other network parameters adapt such that the resulting sentence vectors capture important semantic information [14].

We use a pretrained model for this architecture to produce a 4,800-dimensional real-valued vector representation $\mathbf{v}_s$ of every sentence s in the document, given by the final hidden vector representation of the RNN after appending an end-of-sentence marker. We then compute initial sentence scores as follows:

$$\sigma_0(s) = \sum_{s'} \frac{\mathbf{v}_s^t \mathbf{v}_{s'}}{||\mathbf{v}_s|| \, ||\mathbf{v}_{s'}||} \tag{1}$$

These scores compare the sentences pairwise in terms of the standard cosine measure. Each comparison yields values in $[-1, 1]$ and the sum indicates the global similarity to other sentences in the document. The intuition here is that sentences that relate more closely to the central overall message of the document are more relevant than sentences with more tangential content. As final scores, we compute

$$\sigma(s) = \max\left\{0, \; 1 - 2\frac{r(\sigma_0(s)) - 1}{n}\right\} \tag{2}$$

where $r(\sigma_0(s))$ denotes the rank of $\sigma_0(s)$ among all such similarity scores for different s in the document, n denotes the number of such sentences s, and the factor 2 ensures that 50% of sentences in the document obtain a score of 0.

4.2 The HiText Interface

Document Rendering. HiText's user interface is incorporated into the original text document, thus allowing the document to retain the majority of its original formatting. This enables HiText to naturally and unobtrusively blend into a reader's regular reading process. In particular, the original fonts, page layout, images, tables, and navigational elements such as hyperlinks are all retained, so the original document ergonomics and navigation remain fully functional. This also extends to attributes that may be significant when skimming, including stylistic elements such as bolding. Retaining the original font typefaces may also be important, for instance, for users that rely on fonts such as Dyslexie or OpenDyslexic that mitigate the effects of dyslexia. Nevertheless, HiText can easily be configured to modify the typeface and point size, if desired.

Implementation-wise, HiText first analyses the sentences in the document using the aforementioned neural network approach and then modifies the document's HTML DOM tree to incorporate the user interface. Specifically, it partitions paragraphs into individual spans of text corresponding to the previously identified sentences (using HTML span elements) and injects additional JavaScript code to enable a dynamic highlighting of these text spans.

A global button allows the reader to disable the highlighting entirely, so that HiText can disappear when not desired, again in accordance with our goal of remaining unobtrusive.

Top phrase highlighting. When enabled, the HiText approach highlights the top-k sentences in a document by modifying the background color of those sentences, as depicted in Figure 2 (left). The rationale for this is to enable readers to quickly identify salient sentences when glancing over the document. This mirrors the way in which readers of physical books often use highlighter pens during the reading process to mark important parts for possible revisiting.

In comparison with approaches that only display raw short summaries, our approach may require scrolling over the document. In return, this form of visualization can feel more natural to users, as it directly corresponds to the normal

scrolling behaviour when skimming over an unannotated document. Importantly, readers maintain a clear sense of how much text they are skipping over. The reader also has the ability to read additional parts of the text instead of just the highlighted ones, without losing track of the original order and structure of the text. Implementation-wise, this process simply requires enabling custom background colors for the text spans corresponding to the top-ranked sentences.

Dynamic graded highlighting. In order to account for the graded nature of salience and to dynamically support the reading process when the reader decides to drill down on parts of the text, we propose to dynamically highlight further parts of the text in a more fine-grained manner. In the interest of unobtrusiveness, this is performed only on demand. Fortunately, when reading, people often rely on their fingers or a pen to direct their attention across the page. On computers with pointing devices, this *meta-guiding* has a natural analogue: Readers can use their pointing device to trace and guide their attention. We draw on this natural behaviour by capturing the location of the pointing device so as to determine the current paragraph of interest.

For this currently active paragraph, additional sentences not among the previously selected top-k sentences are highlighted as well, while for all other paragraphs only the top-k sentences are highlighted. As shown in Figure 2 (right), the degree of highlighting in the currently active paragraph is determined in accordance with the degree of salience. For HiText's graded display, we determine the minimum and maximum salience scores, respectively, over all sentences:

$$\sigma_{\min} = \min_s \sigma(s)$$
$$\sigma_{\max} = \max_s \sigma(s)$$

We assign two background colors $C_{\min}$, $C_{\max}$ to sentences with $\sigma(s) = \sigma_{\min}$ and $\sigma(s) = \sigma_{\max}$, respectively. Normally, $C_{\min}$ is set to the regular background color of the document in order to maintain an unobtrusive non-highlighting of minimally salient sentences. $C_{\max}$ is normally set to the same highlighting color as for the top-k sentences, so that their background color remains unchanged when the paragraph is selected by the user.

All other sentences are assigned an interpolated color between $C_{\min}$ and $C_{\max}$. For this, we first convert their color representations to the CIE-LCh color space, in order to account for human perception of color differences. In this color space, we can linearly interpolate between $C_{\min}$ and $C_{\max}$ while maintaining saturation and brightness. Thus, any score $\sigma(s)$ can be mapped to an interpolation

$$\frac{\sigma(s) - \sigma_{\min}}{\sigma_{\max} - \sigma_{\min}} (C_{\max} - C_{\min}) + C_{\min}.$$

The resulting CIE-LCh space color representations can then be converted back to the RGB color space for rendering. Figure 3 provides an example of HiText's ergonomic mode, which relies on a more subtle color scheme for highlighting.

5. INTERFACE EVALUATION

We conducted a series of experiments to provide a multifaceted assessment of HiText in practice. Experiment 1 evaluated reader satisfaction and efficiency, while Experiment 2 considered efficiency and effectiveness of our interface based on reading comprehension tests, for a more objective assessment. Additionally, in Experiment 3 we analysed the text summarization method.

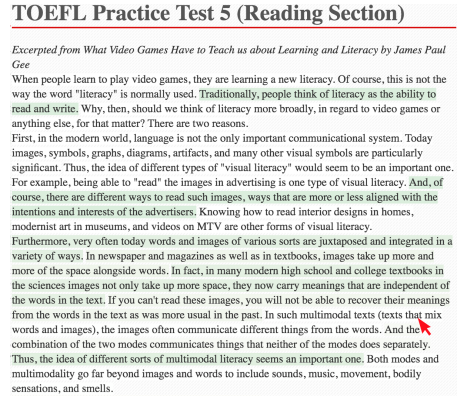

Figure 3: HiText's alternative ergonomic color mode, showing graded highlighting of sentences when hovering on paragraph 3

5.1 Experiment 1: Comprehension Survey

Evaluating comprehension is non-trivial, as it is impossible to repeat a given task with different methods under equal conditions: Once a text has been read by a participant using one set of parameters, a repeated reading by the same participant under a second set of parameters would be biased by the first. One way to overcome this problem is by finding two participants with comparable reading abilities. In our experiment, we make this more robust by relying on two groups (with counterbalancing) chosen such that all participants have the same level of education and the same English certificate.

Participants. We used a pool of 10 participants (4 female, 6 male), with ages ranging from 20 to 26. These were recruited from a number of universities and are not from our lab. All are proficient non-native speakers of English with the same English certificate.

Materials. For our experiment, we chose two articles from popular online journals[2]. Specifically, the title of Article 1 is "A New Clue Suggests Biden May Run" (The New Yorker, October 8, 2015) with 543 words. The title of Article 2 is "There's Already Life on Mars, and We Put It There" (The New Yorker, October 8, 2015) with 1,260 words.

Procedure and Measures. As mentioned above, we avoid a within-participants design, instead opting for a between-group design, in which the pool of participants is randomly divided into two groups (Group A and Group B, with 5 participants each) to read a given text under two different conditions in a counterbalanced order:

1. The first condition (Top5), for the control group, is that of having just the top-5[3] sentences highlighted. This is the obvious way of highlighting text using the output of a text summarization engine, as performed by the Nestor Highlighter mentioned in the Related Work section.

[2]Online at `http://www.larayang.com/hitext/`
[3]5 is set heuristically, based on the typical number of sentences in a regular human-written summary for news.

2. The second condition (Top5+Graded) involves reading the text using our proposed method HiText, with top-5 highlighting and mouse hover-controlled graded highlighting of additional sentences. To create the sentence vectors $\mathbf{v}_s$, we rely on the neural model by Kiros et al. [14], trained on a large corpus of 74,004,228 sentences from 11,038 books [34].

In order to better account for differing reading preferences, we swapped the group assignment for the second article. Thus, half of the subjects (Group A) read Article 1 (Top5 condition), and Article 2 (Top5+Graded condition), while the others (Group B) read Article 1 (Top5+Graded) and Article 2 (Top5). The relevant statistics for Top5+Graded were considered the HiText group, while those for Top5 as the control group. The students were instructed to skim a given article at their own pace until they felt they had grasped the main points. We measured the time taken until this point. In a post-experimental survey, we independently collected qualitative feedback from every participant, asking them whether they believe HiText improves their reading speed (Q1), whether HiText improves their reading experience (Q2), and whether they recommend HiText to others (Q3). Specifically, we asked these questions, and let participants choose between *Strongly Agree*, *Agree*, *Not sure*, *Disagree*, *Strongly Disagree*.

Results and Discussion. The results are given in Table 1. The HiText group needed less time to read the article. In Figure 4, we provide as an example the specific times taken by readers for the first article in our evaluation set. To enable an easier comparison, the readers here are sorted by the time taken, in descending order. The control group is shown on the left (1-5 in red), while the HiText group is shown on the right (6-10 in black).

	Article 1	Article 2
HiText Group	86 (40.37, 18.06)	187 (183.15, 81.91)
Control Group	127 (66.86, 29.90)	234 (121.57, 54.37)
Time saved	32.28%	20.09%

Table 1: Efficiency analysis (Experiment 1), given as means (std. deviation, std. error) in seconds.

The results of the post-experimental survey are given in Table 2. We observe that 10% reported that they agree and 90% reported that they strongly agree with the idea of the user experience being better using HiText. One of the main reasons they provided was that the use of color (in particular the dynamic highlighting based on their mouse movements) helped guide their field of view. Likewise, 90% of participants reported that they strongly agree with the thought of recommending HiText to others, while the remaining 10% agree as well.

One participant suggested first showing a short abstract and then the HiText-enabled main text. This appears to be a useful idea, as it would resemble the way human-written abstracts are often presented. Another participant suggested a global option to enable the graded display of the entire text. While our intention had been to avoid an overly colorful graded display of all text in order to remain unobtrusive, the idea of making this available as a global option is a natural extension to the current global enabling/disabling of HiText.

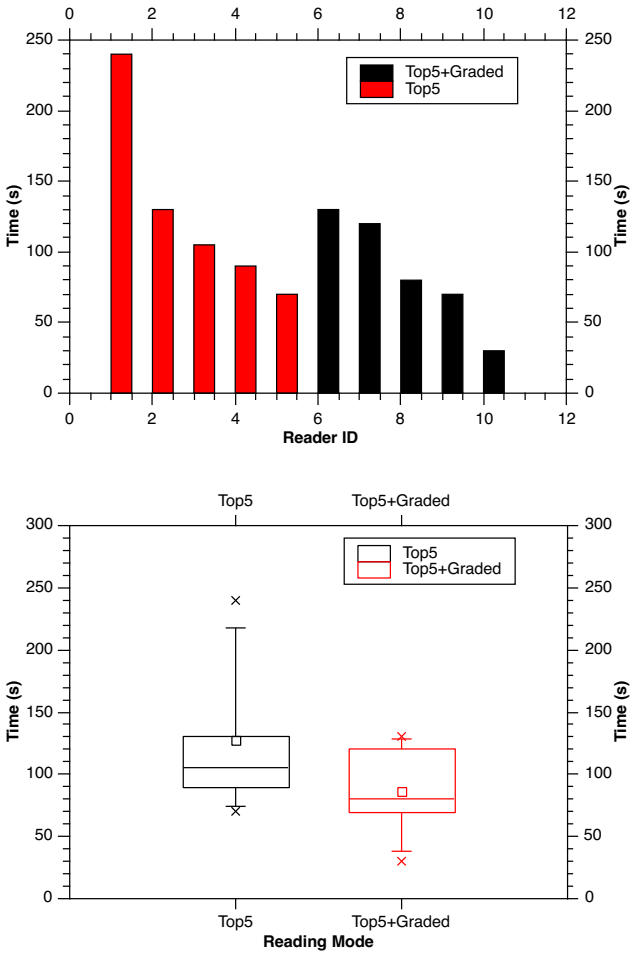

Figure 4: Analysis of reading times for article 1 (Experiment 1) as a bar chart and box plot. Top5 refers to the control group and serves as the baseline.

5.2 Experiment 2: Comprehension Tests

Experiment 1 focused on self-assessed satisfaction. Given the variability in skimming behaviour between different readers, we conducted an additional experiment to gain a second perspective, relying on more objective reading comprehension tests for the assessment. In Experiment 2-A, we consider the time variable (efficiency) for two different interfaces, given the same level of correctness (effectiveness) for comparable articles and questions. In Experiment 2-B, we instead focus on the correctness obtained with two different interfaces, given the same time limit and comparable articles and questions.

5.2.1 *Experiment 2-A: Response Time*

Materials. We used practice texts for the well-known standardized TOEFL (Test of English as a Foreign Language).[4]

[4]The tests, which we have made available online at `http://www.larayang.com/hitext/`, were taken from the gradu-

	Strongly Agree	Agree	Not Sure	Disagree	Strongly Disagree
Q1	1.00	0.00	0.00	0.00	0.00
Q2	0.90	0.10	0.00	0.00	0.00
Q3	0.90	0.10	0.00	0.00	0.00

Table 2: Qualitative feedback from participants.

These are assumed to be of comparable difficulty. Text 1 (Plain text) has 649 words, while Text 2 (TOP5) has 974 words, and Text 3 (TOP5+Graded) has 818 words. Each text already came with 10 reading comprehension questions (which were pre-existing, i.e. not written or selected by us and established prior to and entirely independent of the hightlighted sentences delivered by our system). We used the final test question (Question #10 of every TOEFL test) provided for each text with the original texts, as these final TOEFL reading comprehension questions evaluate the reader's global understanding of the text rather than seeking a specific piece of information that could be very local and can be selected by the reader via scanning, i.e. without a proper understanding of the overall message. Note that these are multiple choice assessments so as to facilitate evaluating the correctness consistently across different users.

Participants, Procedure, and Measure. Since in this evaluation we have texts and reading comprehension questions of comparable difficulty, we were here able to opt for a within-participants design. We sequentially let new participants (more than the 10) read these three texts with a reading comprehension question one by one under a counterbalanced order. Participants read texts first before seeing the question to avoid confounding effects. Since we are here considering the time variable given the correctness, only participants that answered all three questions correctly were considered. Thus additional participants kept taking the test (13 in total) until we had 10 participants with correct answers, and could end the data collection. The plots and figures refer to the 10 considered subjects (4 out of 10 female, 6 male).

We measured the time that the participants took until they made a choice, considering for our results only those who selected the correct response, indicating that they have properly understood the key ideas.

Results and Discussion. The results are given in Table 3 and Figure 5. Plain articles required the longest time for participants to choose a correct answer, with an average time of 263 seconds. Articles shown with Top-5 highlighting required less time, with an average response time of 200s, while articles with Top-5 + graded highlighting required the least time, with an average time of 110s. The small standard deviation for Plain Text confirms that the reading ability levels for these participants is similar.

5.2.2 *Experiment 2-B: Time-Restricted Case*

In the previous response time evaluation, we considered the time variable given equal conditions with respect to the correctness of the supplied answers. The ability of subjects to correctly answer questions was instead evaluated separately in the following experiment, where we assess the rate

ateshotline.com website, since the official ETS page provides just a single example for the reading test.

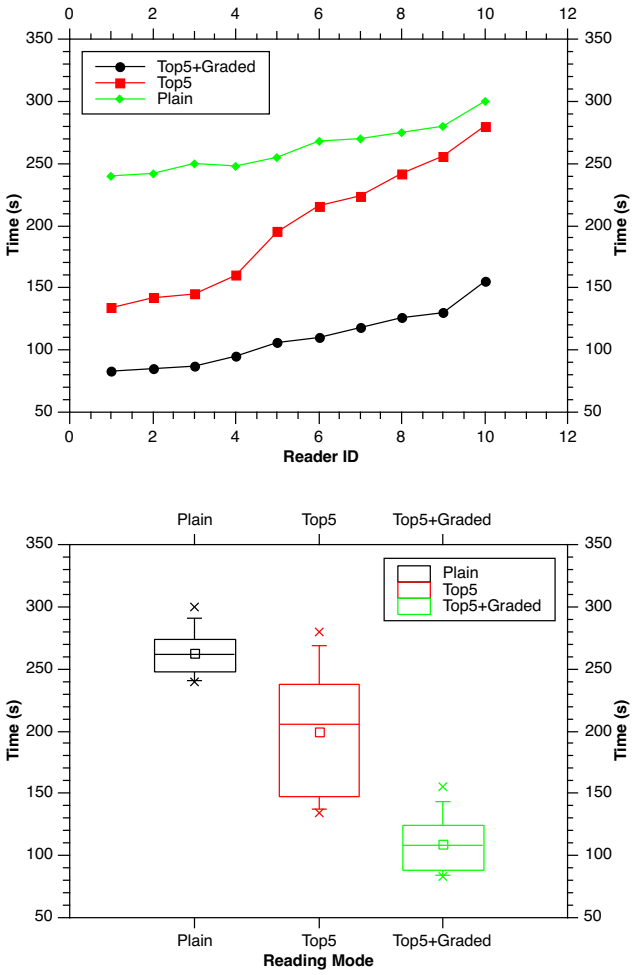

Figure 5: Response time for correctly answering reading comprehension questions (Experiment 2-A)

of correct answers given equal conditions with respect to the time taken.

Participants and Materials. A different set of 10 participants (5 female) took part in this evaluation. We used the same materials as in Experiment 2-A.

Procedure. The readers were asked to first read the texts and then answer the question within 2 minutes. Again, they were not allowed to see the answers while reading the text, in order to avoid confounding variables. The time constraint was announced in advance, and they were aware of the elapsed time during the reading phase. They were asked to respond to the final overall reading comprehension question, irrespective of whether the time had sufficed for them to understand the text or not.

Results and Discussion. We computed the resulting ratios of correct results for the three settings. For plain text, only 40% of participants answered the question correctly. For Top5 highlighting, 70% answered correctly, while for Top5 + Graded highlighting, in fact 100% made the right choice.

	Plain	Top5	Top5+Graded
Average Time (SD, SE)	262.8 (19.16, 6.06)	199.4 (52.17, 16.50)	109.5 (23.25, 7.35)
T-test (paired)	w/ Top5 t = 5.928, p-value = 0.0002213	w/ Top5+Graded t = 9.5539, p-value = 5.225e-06	w/ Plain t = 89.832, p-value = 1.331e-14

Table 3: Efficiency analysis (Experiment 2-A), given as means (std. deviation, std. error) in seconds.

5.3 Experiment 3: Salience Scoring

For additional analysis, we also compared our method with existing algorithms used for generating short summaries. For this comparison, we need extractive summarization algorithms such that the sentences in the summaries are sentences from the original text. We considered algorithms that are unsupervised, and as such do not require new training data for each domain or genre of text. Perhaps the most well-known such approach is the LexRank/TextRank strategy of generating summaries by measuring salience in terms of the PageRank algorithm for graph centrality in a sentence similarity graph [9]. Other well-known algorithms include the frequency-based SumBasic method [20] and the seminal IBM approach by Luhn et al. [17].

Materials. We use one of the texts from our pool of texts from Experiment 1. Specifically, "A New Clue Suggests Biden May Run" (The New Yorker, October 8, 2015) with 543 words.

Participants and Procedure. For this evaluation, we had two independent readers individually annotate each sentence in the text. For each sentence, the readers were asked to assess to what degree they deemed the sentence important for obtaining a basic understanding of the contents of the article. For the responses, we used the 7-point Likert-style importance scale given by Vagias [26].[5]

For LexRank, SumBasic, and Luhn, we gradually increased the length of the expected summaries, so that in every step newly added sentences were assessed as less salient than previously existing ones. Thus, we ultimately rank the salience of every sentence. For HiText, we obtain the rank directly from its salience scores. To assess the correlation of such ranks with the human assessments, we rely on Spearman's rank correlation coefficient with proper tie handling.

Results and Discussion. The resulting Spearman correlation scores results are given in Table 4. At 0.91, the inter-annotator agreement between the two human annotators was remarkably high. Our deep learning-based scores correlated quite well with the human judgements, outperforming LexRank. Surprisingly, SumBasic and the Luhn method performed staggeringly poorly.

We suspect that these methods fall short because they rely on a form of word probability weights as a proxy for semantic similarity. Word-based comparisons may work well for large document clusters, but such a strategy often fails when just operating with short text units such as sentences, as these may use different words to describe related concepts or matters of affairs. This ranges from synonyms such as *car* and *automobile* to entirely different sentence phrasing. Deep learning-based representations diminish the effects of such phenomena by learning to map different sentences to similar representations, even when the specific words and phrases differ. In future work, we intend to explore improved salience scores that excel even further at this [31, 27].

[5](1) Not at all important, (2) Low importance, (3) Slightly important, (4) Neutral, (5) Moderately important, (6) Very important, (7) Extremely important

	Annotator 1	Annotator 2
Annotator 2	0.91	1.00
Our method	0.75	0.83
LexRank	0.53	0.52
SumBasic	-0.06	-0.05
Luhn	-0.15	-0.06

Table 4: Spearman correlations with human salience assessments.

Another minor issue was that both the SumBasic and the Luhn implementation misinterpreted some full stops as sentence boundaries (for incorrectly split sentences, our evaluation chooses randomly among the ranks computed for the sentence fragments).

5.4 General Discussion

Our study focus on news articles, but can be adapted to books etc. by regarding individual chapters or a restricted window of context as the current document when computing salience scores.

One limitation of the current instantiation of HiText is its reliance on a pointing device for meta-guiding. HiText can also be used on mobile devices with floating touch technology, i.e., the ability to detect a finger or pen hovering over the screen, in this case directly mirroring the finger or pen-based meta-guiding of people reading traditional print media. On mobile devices still lacking such technology, support for touch or stylus inputs would entail minor differences in the behaviour of the user interface. In particular, a paragraph will remain highlighted until a new paragraph is selected, rather than for the duration of the pointer remaining within the paragraph. Thus, further study is necessary to assess this alternative in greater detail.

6. CONCLUSION

Reading has become one of the most common forms of interaction between humans and machines. Yet, students and other readers are often ill-equipped to deal with the modern challenge of information overload. We have presented HiText as a simple but effective new method for guiding the reader towards salient content units in text, enabling faster and better reading comprehension. This opens up the possibility of widespread adoption in software as well as the potential to foster important further research in this little-studied but increasingly critical area.

7. ACKNOWLEDGMENTS

This work was supported in part by the National Basic Research Program of China Grants 2011CBA00300 and 2011CBA00301, as well as National Natural Science Foundation of China Grants 61033001 and 61361136003.

8. REFERENCES

[1] F. Ahmed, Y. Borodin, Y. Puzis, and I. V. Ramakrishnan. Why read if you can skim: Towards enabling faster screen reading. In *Proc. W4A*, pages 39:1–39:10, 2012.

[2] P. L. Carrell. Facilitating esl reading by teaching text structure. *TESOL Quarterly*, 19(4):727–752, 1985.

[3] J. Chen, N. Tandon, C. D. Hariman, and G. de Melo. WebBrain: Joint neural learning of large-scale commonsense knowledge. In *Proc. ISWC*, 2016.

[4] E. H. Chi, L. Hong, M. Gumbrecht, and S. K. Card. ScentHighlights: Highlighting conceptually-related sentences during reading. In *Proc. IUI*, 2005.

[5] J. Chung, Ç. Gülçehre, K. Cho, and Y. Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *CoRR*, abs/1412.3555, 2014.

[6] D. Das and A. F. T. Martins. A survey on automatic text summarization. Technical report, Carnegie Mellon University, 2007.

[7] G. de Melo. Inducing conceptual embedding spaces from Wikipedia. In *Proc. WWW 2017 (Cognitive Computing Track)*. ACM, 2017.

[8] G. B. Duggan and S. J. Payne. Skim reading by satisficing: Evidence from eye tracking. In *Proc. CHI 2011*, pages 1141–1150. ACM, 2011.

[9] G. Erkan and D. R. Radev. Lexrank: Graph-based lexical centrality as salience in text summarization. *J. Artif. Int. Res.*, 22(1):457–479, Dec. 2004.

[10] D. J. Gillick. *The Elements of Automatic Summarization*. PhD thesis, EECS Department, University of California, Berkeley, May 2011.

[11] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Comput.*, 9(9):1735–1780, Nov. 1997.

[12] N. Kalchbrenner and P. Blunsom. Recurrent continuous translation models. In *Proc. EMNLP*, pages 1700–1709. ACL, 2013.

[13] D. Kingery and R. Furuta. Skimming electronic newspaper headlines: a study of typeface, point size, screen resolution, and monitor size. *Inf. Process. Manage.*, 33(5):685–696, 1997.

[14] R. Kiros, Y. Zhu, R. Salakhutdinov, R. S. Zemel, A. Torralba, R. Urtasun, and S. Fidler. Skip-thought vectors. *CoRR*, abs/1506.06726, 2015.

[15] X. Long, C. Gan, and G. de Melo. Video captioning with multi-faceted attention. *CoRR*, abs/1612.00234, 2016.

[16] E. Loza Mencía, G. de Melo, and J. Nam. Medical concept embeddings via labeled background corpora. In *Proc. LREC*, Paris, France, 2016.

[17] H. P. Luhn. The automatic creation of literature abstracts. *IBM Journal of Research and Development*, 2(2):159–165, 1958.

[18] M.-T. Luong, H. Pham, and C. D. Manning. Effective approaches to attention-based neural machine translation. In *Proc. EMNLP*, pages 1412–1421, 2015.

[19] A. Nenkova and K. McKeown. A survey of text summarization techniques. In *Mining Text Data*. 2012.

[20] A. Nenkova and L. Vanderwende. The impact of frequency on summarization. *Microsoft Research, Redmond, Washington, Tech. Rep. MSR-TR-2005-101*, 2005.

[21] V. d. Paiva, D. Oliveira, S. Higuchi, A. Rademaker, and G. de Melo. Exploratory information extraction from a historical dictionary. In *Proc. Workshop on Digital Humanities and e-Science at the 10th IEEE International Conference on e-Science*, 2014.

[22] E. Pitler and A. Nenkova. Revisiting readability: A unified framework for predicting text quality. In *Proc. EMNLP*, pages 186–195, 2008.

[23] D. N. Rapp and P. Broek. Dynamic text comprehension: An integrative view of reading. *Current Directions in Psychological Science*, 14(5):276–279, 2005.

[24] I. Sutskever, O. Vinyals, and Q. V. V. Le. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems 27*, pages 3104–3112. 2014.

[25] M. Theobald, J. Siddharth, and A. Paepcke. Spotsigs: Robust and efficient near duplicate detection in large web collections. In *Proc. SIGIR 2008*, 2008.

[26] W. M. Vagias. Likert-type scale response anchors. *Clemson International Institute for Tourism & Research Development*, 2006.

[27] O. Čulo and G. de Melo. Source-Path-Goal: Investigating the cross-linguistic potential of frame-semantic text analysis. *it - Information Technology*, 54, 2012.

[28] M. Walsh. The 'textual shift' : examining the reading process with print, visual and multimodal texts. *Australian Journal of Language and Literacy*, 29(1):24–37, 2006.

[29] Y. Wang, Z. Ren, M. Theobald, M. Dylla, and G. de Melo. Summary generation for temporal extractions. In *Proc. DEXA*, 2016.

[30] A. J. Wecker, J. Lanir, O. Mokryn, E. Minkov, and T. Kuflik. Semantize: Visualizing the sentiment of individual document. In *Proc. AVI 2014*, pages 385–386. ACM, 2014.

[31] Q. Yang, R. J. Passonneau, and G. de Melo. PEAK: Pyramid evaluation via automated knowledge extraction. In *Proc. AAAI*. AAAI Press, 2016.

[32] J. S. Yi. QnDReview: Read 100 CHI papers in 7 hours. In *CHI '14 Extended Abstracts*, pages 805–814. ACM, 2014.

[33] K. X. Yong, K. Rajdev, T. J. Shakespeare, A. P. Leff, and S. J. Crutch. Facilitating text reading in posterior cortical atrophy. *Neurology*, 85(4):339–348, 2015.

[34] Y. Zhu, R. Kiros, R. S. Zemel, R. Salakhutdinov, R. Urtasun, A. Torralba, and S. Fidler. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. *CoRR*, abs/1506.06724, 2015.