Elastic Documents: Coupling Text and Tables through Contextual Visualizations for Enhanced Document Reading

Sriram Karthik Badam, Zhicheng Liu, and Niklas Elmqvist, *Senior Member, IEEE*

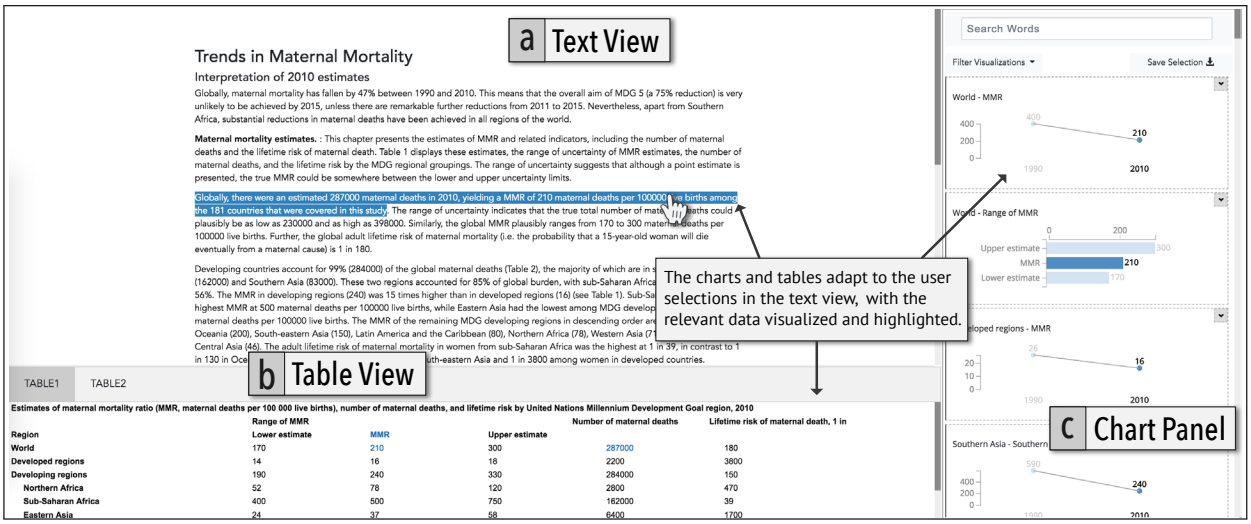


Fig. 1: Our Elastic Documents approach extracts text and tables from data-rich documents into separate views and connects them using visualizations generated from the tables. The visualizations adapt to the user's interaction including selection of text (highlighted in blue) and filtering the table attributes (on top right). The visualizations relevant to the user's focus are extracted by keyword matching; in this case, notice that the first two visualizations capture the data context of the highlighted sentence.

Abstract—Today's data-rich documents are often complex datasets in themselves, consisting of information in different formats such as text, figures, and data tables. These additional media augment the textual narrative in the document. However, the static layout of a traditional for-print document often impedes deep understanding of its content because of the need to navigate to access content scattered throughout the text. In this paper, we seek to facilitate enhanced comprehension of such documents through a contextual visualization technique that couples text content with data tables contained in the document. We parse the text content and data tables, cross-link the components using a keyword-based matching algorithm, and generate on-demand visualizations based on the reader's current focus within a document. We evaluate this technique in a user study comparing our approach to a traditional reading experience. Results from our study show that (1) participants comprehend the content better with tighter coupling of text and data, (2) the contextual visualizations enable participants to develop better summaries that capture the main data-rich insights within the document, and (3) overall, our method enables participants to develop a more detailed understanding of the document content.

Index Terms—Document reading, contextual visualizations, visual aids, comprehension, summarization

1 Introduction

Electronic documents are one of the most pervasive digital media in use today. Many of them are *data-rich* in that they are essentially complex datasets in themselves, containing not just text but also diverse forms of information such as tables, photographs, and illustrations (e.g., reports on hurricane damage [45] and cancer statistics [49, 50]). However, most traditional document formats—even data-rich ones—are still designed for print, which means that they are essentially static in nature and are not optimized for electronic reading. Examples of print formats include not only Adobe's Portable Document Format (PDF), but also word processors such as Microsoft Word and Google Docs, as well as desktop publishing tools such as Adobe InDesign [2]. Even online formats, HTML in particular, still follow many of the conventions of print media when used as a data-rich document format.

Current displays for data-rich documents not only often fail to provide a satisfactory reading experience, they also do not take full advantage of the dynamic nature of digital devices. Print formats often even constrain the layout, flow, and typography to a static and page-driven design. While modern e-book readers typically allow for changing font and font size as well as support hyperlinks and allow for reflowing paragraphs, they still lock the reader into a static and mostly linear reading sequence. In practice, reading a data-rich document on a digital device requires flipping back and forth to track linked content, such as associated figures and tables. This activity is often more cumbersome to perform on a digital device than using a physical document.

To improve the electronic document reading experience, we need to go beyond the classic notion that a document is a linear sequence of multimedia content with its layout and visual design fixed into a single presentation flow. Instead, a document can be viewed as a collection of multimedia content—essentially a heterogeneous dataset—to be dy-

- *Sriram Karthik Badam and Niklas Elmqvist are with the University of Maryland in College Park, MD, USA. E-mail: {sbadam, elm}@umd.edu.*
- *Zhicheng Liu is with Adobe Research in Seattle, WA, USA. E-mail: leoli@adobe.com.*

Manuscript received 31 Mar. 2018; accepted 1 Aug. 2018.
Date of publication 16 Aug. 2018; date of current version 21 Oct. 2018.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TVCG.2018.2865119

namically filtered, rearranged, and presented in any order depending on the readers' goals as well as the capacity of the display itself. Rendering such heterogeneous data on screen amounts to visualization.

Based on this idea, we present *Elastic Documents*, an adaptive approach for enhanced document reading. Text constitutes the main form of information in most data-rich documents, with data tables commonly used in reports and papers from corporations, government agencies, and non-profit organizations. Therefore, in this paper, we focus on two types of content—text and tables—within data-rich documents and investigate how to tightly couple them in an elastic document (Figure 1). Our approach establishes this tight coupling by parsing the document text, structure, and embedded data tables, and using the explicit or implicit relations from words in the text to the data in the tables to dynamically generate contextual visualizations based on the user's focus and interest. By doing so, the reader can interpret the visualizations to understand the context from the data tables for text in focus, and go beyond the traditional linear structure of the document.

To understand the effectiveness of this approach, we evaluated a prototype implementation of Elastic Documents in a laboratory study with 14 participants in comparison to a conventional document viewer. On each interface, the participant summarized data-rich documents containing text and tables and then answered specific questions that connect the text and tables. We found that the participants created summaries based on broader data from the document when using Elastic Documents compared to the baseline. They also provided precise answers when the visualizations were present. Furthermore, we found differences in the reading patterns compared to the conventional document reader, with participants showing more focused reading for Elastic Documents that closely followed the narrative of the document by perceiving the visualizations rather than just consolidating text and tables. Overall, we contribute the following through this paper:

1. An approach for data-rich document viewing by connecting the text and data tables through contextual visualizations, using a three-step pipeline involving table content extraction and visualization based on user interaction.
2. A proof-of-concept implementation of the proposed approach.
3. Results from a validation of the proposed approach showcasing better performance and usage patterns compared to a baseline.

2 Related Work

Here we summarize three related research themes that inform our Elastic Documents approach for data-rich documents.

2.1 Adaptive Document Viewing

Multiple research efforts augment or transform static documents for specific functions and activities. Grossman et al. [28] demonstrate how to replace static images with animated figures in PDFs to benefit text comprehension. Document Cards [54] summarizes research papers by choosing important figures and presenting the gist of the document in a deck of cards for quick consumption of document content. Jacobs et al. [32] investigate automatic formatting and layout of document content to adapt different display sizes. Fluid documents [13] dynamically update the visual salience of primary content (e.g., body of text) in a document and supporting materials (e.g., annotations) depending on user focus. TextTearing [65] introduces a technique that expands inter-line white space in a document for digital ink annotation. Finally, LiquidText [56] provides a multi-touch environment with a fluid document representation for active reading tasks such as highlighting and annotating. While these efforts add flexibility to static digital documents, they focus primarily on adapting the layout and not on connecting multiple entities in the content, such as text and tables.

Within accessibility research, document adaptation is important for helping readers with reading disabilities or visual impairment. For instance, techniques have been introduced to help visually impaired users by automatically transforming textual documents into responsive layouts on a small screen that can be selected to be read-out-loud by a speech synthesizer [47, 53]. Screen magnifiers [7] have also been effective to adapt documents to the needs of people with vision impairment. While the Elastic Documents approach is not designed for this domain, these research efforts showcase the current advances and methods for parsing and adapting documents to the user.

2.2 Generating Visualizations from Text and Tables

Visualizations have long been used for distant reading of documents [40] by transforming the text content into an abstract view to present global features. Techniques such as tag clouds [61], social relationship graphs [35], and Phrase Nets [59] support distant reading. For close reading [9], it is important to retain the text structure and content. Hence visual aids in close reading rely on augmentation through colors, fonts, and visual marks [12]. Jänicke et al. [33] surveyed such distant and close reading techniques for digital humanities.

Data visualizations have also been used to enhance and complement news articles. Contextifier [31] automates the generation of annotated visualizations of stock prices for companies mentioned in an article. NewsViews [22] extends this approach to geovisualization and works with a broader set of news articles and data sources obtained through crawling. PersaLog [3] contributes a domain-specific language for personalizing visualizations in news articles given a reader profile. However, these approaches all assume that datasets are separate from the text content, and the data schema is relatively simple.

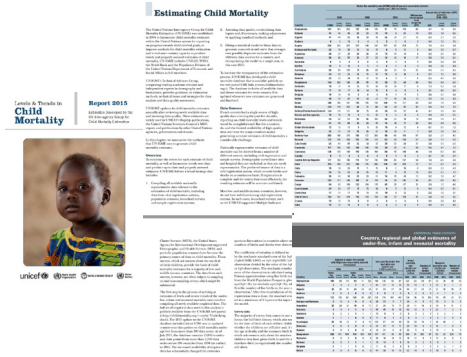
In our case, the relevant data tables are included in the documents, but the structures of the tables are more complex. We thus used approaches introduced by Chen and Cafarella [16, 17] to parse and extract data from data-frame tables [15]. Previous approaches of automatic visualization generation often rely on column types and names to decide encoding choices [38]. While these efforts enable us to work with existing data spreadsheets and tables, the question of how to extract tables from documents (in, say, PDF format) still needs to be answered for Elastic Documents. PDF-TREX [43], introduced by Oro and Ruffolo, is a heuristic approach for parsing PDF documents to recognize tables. Extraction of complex hierarchical tables from PDF documents has also been achieved through such heuristic approaches (cf. pdf2table [64]). Now, commercial tools such as Adobe Acrobat [1] support parsing and extraction of content from PDF documents.

Visualizations can enhance text and table reading. Table Lens [46] represents an early approach to improve table reading through in-situ visualizations in an interactive layout. Bertin's matrices [5, 44] exemplify tabular visualizations by visually encoding cell values within tables and supporting grouping operations. Word-scale visualizations [10, 24] enhance document reading with graphical encodings that span the size of a word embedded in the text content. However, there are open questions in the design of word-scale visualizations [26] and in generating them for data-rich documents. To connect visualizations to the text being read, Kong et al. [36] utilized crowdsourcing to extract relations and highlighted them in a document viewer.

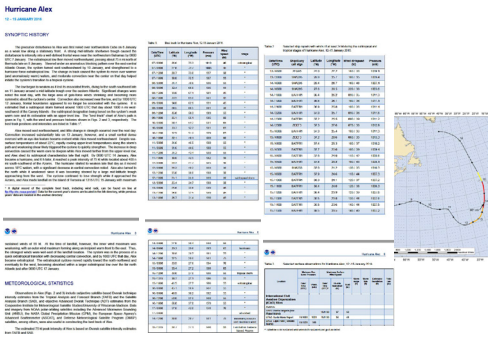
In contrast to Chen and Cafarella [16], we go beyond table headers and use cell values for visualization and highlighting. Also, our approach focuses on separating the visualizations and using them to connect the text and tables based on the user's focus, and hence, we have not yet considered the word-scale approach.

2.3 Effects of Visual Aids in Document Reading

Many studies evaluate and explain the role of visual aids in text comprehension. For instance, illustrations are found to help mentally represent procedures described in text [23]. Illustrations also help with the construction of a mental model that facilitates inferences in text comprehension [29]. Zellweger et al. [66] evaluated fluid documents [13] through an eye-tracking based study and found no differences in eye movement patterns when *glosses*, previews of hypertext content, are introduced. Duke and Pearson recommend that effective practices to improve reading comprehension include providing visual representations of text [21]. Recently, an evaluation of word-scale visualizations [25] has not confirmed any effect of these representations in question answering tasks, but participants may rely on them instead of textual sentences in case of ambiguity. While there is a developed understanding of the role of visualization in problem solving [37] and exploratory analysis [57], there still is a need to evaluate the affordances of visualization on comprehension and recall in document reading.



(a) Child Mortality report from UNICEF



(b) Hurricane Alex report from NHC

Fig. 2: Example data-rich documents from public sources [41, 58]. Both documents were released in the past three years and include long tables spanning many pages along with a data-rich textual narrative.

3 Design Considerations: Data-Rich Documents

Here we review the concept of data-rich documents as well as the design requirements we collected for close reading of such documents.

3.1 Data Model

A data-rich document is a form of document that conveys information using text as well as rich media such as tables, illustrations, and images. While text forms the primary content describing the main message of the story, it is augmented with other content types to support the narration with factual information (Figure 2). Data-rich documents come from a variety of sources. For science, they form the means to communicate, record, and replicate scientific progress in academic journals and conferences by presenting results in text augmented by data tables, images, and illustrations of results, procedures, and outcomes. For open data, data-rich reports are issued by private and public agencies to communicate findings, establish accountability, and detail procedures. For instance, the National Hurricane Center (NHC)[1] releases situation reports following hurricanes and tropical storms in the United States containing information from scientific sources about the timeline, the collected statistical measures, casualty and damage information, and accuracy of the forecasting models. In business, financial reports, replete with numerous data tables, are released by companies to track their profit and expenses.

While there exist many data-rich documents online,[2] we will focus on two documents—a report on Hurricane Alex from NHC [41] and a quarterly report from Apple [4]—as the running examples to describe our Elastic Documents approach. Both documents are canonical data-rich documents in that they incorporate data tables and illustrations to complement the textual content with factual data. However, reading these documents on a traditional document viewer is complicated because the tables and figures are large and sometimes span several pages, and references between the text and other content may require scrolling back and forth in the document. Our goal with Elastic Documents is to simplify the reading experience of data-rich documents by adapting the document content viewed to the user.

3.2 Tasks and Design Requirements

A major challenge with data-rich documents is reading the tables that tend to be long, complex, and often hard to connect to the text (shown in Figure 2). Therefore, we focus on the **data tables** as they are most amenable to automatically parse and connect to the textual content.

There are a wide range of reading tasks [9, 29, 33] including distant reading, browsing, searching, active reading, and close reading. We focus on close reading [9] and quick comprehension where users read and may even reread the document to extract data in a given amount of time to understand the main insights. This is applicable to data-rich documents as they are often used by journalists as an information source or by the public as a general reference. To answer these challenges, we considered the following requirements:

- R1 **Augment documents with visual aids.** Most data-rich documents present structured data in tables to augment textual content. Referring to raw data tables while reading the textual narrative is challenging in terms of understanding the data and its connection to the text. Therefore, visualizations can be used to ease comprehension of data in data-rich documents, owing to their benefits in data sensemaking [11, 37, 51] and close reading [33].
- R2 **Simplify complex and long table structures.** Tables in data-rich documents are often complex since they record the factual data related to the topics in the text. They do not always follow a relational table format, and hierarchy within columns and rows in the tables [14, 16] is common (Figure 3). Moreover, they may contain sparse structures (Figure 3(c)). From a user's perspective, these complexities should be hidden to provide effective representations of the tables.
- R3 **Connect content spread across text and tables.** Data-rich documents also spread the text and data tables across pages (Figure 2). To go beyond a linear layout that requires scrolling back-and-forth, Elastic Documents should bridge the content across text and tables (cf. adaptive document layouts for fitting text and graphics to a display [32]). For this purpose, the data from the tables should be presented in context to the user while reading the text to enhance document reading.
- R4 **Adapt to user's interest.** Data-rich documents cover multiple data attributes relevant to the purpose of the document. In doing so, parts of the document have specific footprint within the data tables. For instance, the report on Hurricane Alex [41] (Figure 2(b)) begins with outlining the timeline of the hurricane followed by description of collected statistics, causalities, and forecasts. Therefore it is important to help the reader see relevant content and data attributes from the data tables when reading particular sections of text (e.g., timeline of Hurricane Alex). Understanding user's focus and interest while reading the document can help in augmenting the content with relevant visual aids.

4 Contextual Visualization of Data Tables in Elastic Documents

In this paper, we focus on connecting the text and table content by generating visualizations that can help the user better comprehend the text and the underlying data while reading the document. These visual aids should be easy to interpret so as to not deviate the reader from the primary task of comprehending the text within the documents. To meet the above design requirements, we describe our approach through a three-stage document processing pipeline. We also discuss the design rationale in building an interactive interface for Elastic Documents.

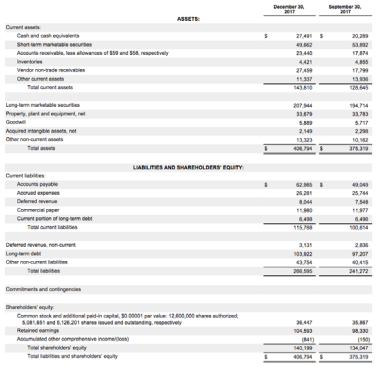
4.1 Document Processing

The data-rich document processing pipeline consists of three stages: (1) parsing document tables to extract data attributes from the table headers (based on Chen and Cafarella [16]), (2) generating visualizations, and (3) extracting and matching phrases from the text, as well as the tables, to make connections between them.

[1]NHC reports: `https://www.nhc.noaa.gov/data/tcr/`
[2]Sample data-rich documents: `http://ter.ps/datadocs`

Table 1: Relationships within row and column header cells in the tables along with the method for identifying them. Note that these are directly based on the four relations from Chen and Cafarella [16], but split further into six for clarity. Examples of these relations are shown in Figure 4.

	Relation	Description	Identification
SS	Stylistic similarity	Elements in table with similar style aspects—spacing, fonts, weights, colors etc.—are treated as being related and within the same level of the hierarchy.	Match predefined styles in row and column headers to identify related attributes.
AD	Adjacent dependency	Adjacent attributes spread across rows and columns can capture a parent-child relationship in the hierarchy.	Check content of adjacent cells in headers to find parent-child relations.
LD	Layout design	Attributes can be laid out in a particular orientation with parents above/below the children or left/right. Elements in the same level are oriented similarly.	Group items oriented along the same row or column, and identify their parents which are placed above or below these items or other side.
OD	Overview/detail	Tables can contain aggregate information such as total and average embedded along with the actual data with row headers reflecting this aspect.	Identify labels capturing terms related to aggregates such as "total" and "average" to understand overview/detail.
MS	Semantic similarity	Related attributes can be spread across tables in the document with similar data types or descriptions.	Match column and row labels, and the cells across tables in a document to identify to similar ones.
MD	Metadata	Relationships might exist between headers based on the metadata corresponding to the tables.	If metadata exists, say in captions, use the keywords and content to connect related attributes.

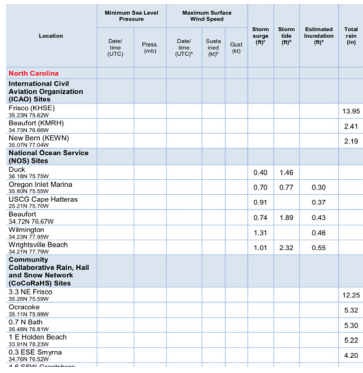


	December 30, 2017	September 30, 2017
ASSETS:		
Current assets:		
Cash and cash equivalents	$ 27,491	$ 20,289
Short-term marketable securities	49,662	53,892
Accounts receivable, less allowances of $99 and $58, respectively	23,440	17,874
Inventories	4,421	4,855
Vendor non-trade receivables	27,499	17,799
Other current assets	11,337	13,936
Total current assets	143,912	128,645
Long-term marketable securities	207,944	194,714
Property, plant and equipment, net	33,879	33,783
Goodwill	5,889	5,717
Acquired intangible assets, net	2,149	2,298
Other non-current assets	13,323	10,162
Total assets	$ 406,794	$ 375,319
LIABILITIES AND SHAREHOLDERS' EQUITY:		
Current liabilities:		
Accounts payable	$ 62,985	$ 49,049
Accrued expenses	26,281	25,744
Deferred revenue	8,044	7,548
Commercial paper	11,960	11,977
Current portion of long-term debt	6,496	6,496
Total current liabilities	115,766	100,814
Deferred revenue, non-current	3,101	2,930
Long-term debt	103,922	97,207
Other non-current liabilities	43,754	40,419
Total liabilities	266,595	241,272
Commitments and contingencies		
Shareholders' equity:		
Common stock and additional paid-in capital, $0.00001 par value: 12,600,000 shares authorized; 5,081,851 and 5,126,201 shares issued and outstanding, respectively	36,447	35,867
Retained earnings	104,593	98,330
Accumulated other comprehensive income/(loss)	(841)	(150)
Total shareholders' equity	140,199	134,047
Total liabilities and shareholders' equity	$ 406,794	$ 375,319

(a) Assets of Apple: Hierarchical row headers with total & individual cells.

Africa			Asia			Europe		
Countries and areas	Under-five mortality rate (U5MR)	U5MR rank	Countries and areas	Under-five mortality rate (U5MR)	U5MR rank	Countries and areas	Under-five mortality rate (U5MR)	U5MR rank
Angola	157	1	Afghanistan	91	16	Republic of Moldova	16	104
Chad	139	2	Pakistan	81	22	Albania	14	112
Somalia	137	3	Lao People's Democratic Republic	67	31	Romania+	11	130
Central African Republic	130	4	Timor-Leste+	53	41	Bulgaria	10	133
Sierra Leone	120	5	Turkmenistan	51	42	Russian Federation	10	133
Mali	115	6	Myanmar	50	44	Ukraine	9	139
Nigeria	109	7	India	48	48	Latvia	8	142
Benin	100	8	Tajikistan	45	52	Serbia+	7	148
Democratic Republic of the Congo	98	9	Yemen+	42	56	Slovakia	7	148
Niger+	96	10	Uzbekistan	39	59	Hungary+	6	153
Equatorial Guinea	94	11	Bangladesh+	38	61	Malta	6	153
Guinea	94	11	Nepal+	36	63	The former Yugoslav Republic of Macedonia+	6	153
Côte d'Ivoire	93	13	Bhutan+	33	67	Belarus+	5	159
Guinea-Bissau	93	13	Azerbaijan	32	68	Bosnia and Herzegovina+	5	159
South Sudan	93	13	Iraq	32	68	Greece	5	159
Lesotho	90	17	Cambodia+	29	71	Lithuania+	5	159
Burkina Faso	89	18	Philippines	28	73	Montenegro+	5	159
Cameroon	88	19	Indonesia+	27	77	Poland+	5	159
Mauritania	85	20	Democratic People's Republic of Korea	25	80	Austria	4	166
Burundi	82	21	Mongolia*	22	84	Belgium	4	166
Mozambique+	79	23	Viet Nam	22	84	Croatia+	4	166
Togo	78	24	Kyrgyzstan+	21	89	Denmark	4	166
Comoros	74	25	State of Palestine	21	89	France	4	166
Zimbabwe	71	26	Jordan	18	96	Germany	4	166

(b) Under-five mortality rate: Flat row structure & hierarchical column headers.



Location	Minimum Sea Level Pressure		Maximum Surface Wind Speed			Storm surge (ft)†	Storm tide (ft)†	Estimated Inundation (ft)†	Total rain (in)
	Date/ time (UTC)	Press. (mb)	Date/ time (UTC)*	Susta ined (kt)*	Gust (kt)				
North Carolina									
International Civil Aviation Organization (ICAO) Sites									
Frisco (KHSE) 35.23N 75.62W									13.95
Beaufort (KMRH) 34.73N 76.66W									2.41
New Bern (KEWN) 35.07N 77.04W									2.19
National Ocean Service (NOS) Sites									
Duck 36.18N 75.75W						0.40	1.46		
Oregon Inlet Marina 35.80N 75.55W						0.70	0.77	0.30	
USCG Cape Hatteras 25.21N 75.70W						0.91		0.37	
Beaufort 34.72N 76.67W						0.74	1.89	0.43	
Wilmington 34.23N 77.95W						1.31		0.46	
Wrightsville Beach 34.21N 77.79W						1.01	2.32	0.55	
Community Collaborative Rain, Hail and Snow Network (CoCoRaHS) Sites									
3.3 NE Frisco 35.26N 75.59W									12.25
Ocracoke 35.11N 75.99W									5.32
0.7 N Bath 35.48N 76.51W									5.30
1 E Holden Beach 33.91N 78.23W									5.22
0.3 ESE Smyrna 34.76N 76.52W									4.20

(c) Hurricane Alex statistics: Sparse layout with hierarchical headers.

Fig. 3: Examples of complex table structures in data-rich documents from Apple inc., UNICEF, and NHC.

4.1.1 Parsing Table Content

Data-rich documents often contain multiple tables with relevant data attributes that support the textual content. As shown in Figure 3, these tables are structured in many ways: ranging from flat relational table formats (rows and columns) to hierarchical row and column segments (e.g., rows within rows). For instance, Apple's quarterly report contains many tables tracking the income, assets, liabilities, etc., for the current and previous quarter along with some cells capturing total values (e.g., total income) for remaining cells in the table. This forms a data table containing hierarchies in row and column headers. Similarly, hierarchies exist in the tables within the reports from NHC and UNICEF (Figure 3). These rich information structures in the tables are first parsed to identify entities and values within the document tables.

Parsing of non-relational table structures has been an active research effort [14, 16]. Tools exist for transforming the data tables into long and wide formats [30, 34]. Our method requires segmentation of table content to understand the relationships within the row and column headers that define a hierarchy; therefore, we rely on the above research efforts. We characterize six such relationships within the table content based on the list from Chen and Cafarella [16] (Table 1).

These relationships are identified while parsing the document tables to extract hierarchical attributes (parent-child structures) within the headers. Flat relational tables are a special case where the hierarchy contains a single level of attributes. Figure 4 shows the input, the workings, and the output of the parsing method applied to a table from Apple's Quarterly report. This report contains (1) stylistic similarities for cells within the same level of the hierarchy, (2) layout design for parent-child relations, and (3) overview/detail design with "total" and individual rows. The algorithm works by iterating over pairs of adjacent cells to tag them with possibilities and then iterating over the entire table to add cells to a hierarchical structure. The output contains hierarchical data structures for both rows and columns. This understanding of the table structure drives the generation of visual aids to support Elastic Documents, detailed in the next section.

4.1.2 Generating Visualizations

Parsing of the data tables extracts structured data and relationships within the table, which can be used to adapt the document content. Following requirement R1, our primary design choice is to utilize visual aids to understand the data tables while reading the document. Visualizations can be generated from the data tables by traversing the parsed hierarchies within the table headers to find combinations of rows and columns. However, the number of combinations can be large based on the density of the table—for sparse tables, certain combinations may not contain any data in the table (e.g., Figure 3(c)). Furthermore, since our goal is to use the visual aids to augment the document text, the representations should be easily interpretable without significant deviation from the reading task and contain sufficient labels to read the actual data values along with the trends. For instance, hierar-

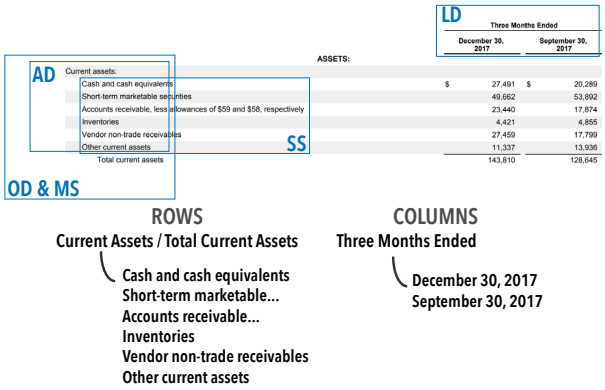


Fig. 4: Outline of the parsing algorithm applied to a segment of a table from Apple's quarterly report (Figure 3(a)). Nodes in the same levels have similar styles (SS); the cells lacking SS in the row headers are adjacent (AD) showcasing a parent-child relation. Overview-detail relation (OD) exists due to "total" keyword and semantic similarity (MS) defines that first and last rows are at the same level. The output of parsing is the hierarchical row and column trees as shown here. Note that this is just a small segment of a larger table.

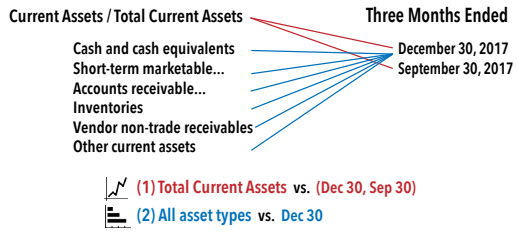


Fig. 5: Two example unit combinations between the row and column hierarchies from Figure 4. The charts created based on the combination are highlighted with an icon. A total of 9 such combinations are possible for this small segment of a larger table in Figure 3(a).

chical representations such as Treemaps [48] or Sunburst [52] would not be ideal as they are hard to interpret while document reading.

For this purpose, we focus on generating visual aids by traversing the hierarchies in the table headers to find combinations of rows and columns that are meaningful and quick to understand (R2). We rely on combinations that generate familiar and simple representations such as line charts and bar charts. We call them, *unit* combinations. A unit combination is defined as a combination of multiple rows with a single column or multiple columns with a single row (essentially, a list of values). These combinations are meaningful as they can share the same unit and they can be visualized by mapping to a scale. Figure 5 shows example unit combinations for the hierarchies in Figure 4.

Unit combinations are represented as a line chart or a bar chart depending on the data content. The values in the table cells for the unit combinations as well as the headers are used to develop the data scales for visual mapping. The row/column headers are checked for time-series attributes by identifying time-related strings—months, years, hours, etc. By doing so, combinations with time-series attributes are represented as line charts, while the rest are presented using horizontal bar charts. Both charts are augmented with labels and axes to show the values along with the visual mapping. Note that the choice of the two chart types is meant to be a starting point to prototype the Elastic Documents approach. Alternate representations, more suitable to the content, are feasible depending on the document. For instance, unit combinations can be long if the table has a flat structure, in which case packed list representations can save space [27, 63] or sampling techniques can be used to show details on demand. The unit combinations are generated automatically by gathering children of each node in the hierarchy and iterating through them as shown in Algorithms 1 and 2.

Input: root node of tree representing a hierarchy in a table header
Result: list of subtrees
list of subtrees = empty list;
queue for traversal = empty queue;
push root into queue for traversal;
while *queue for traversal contains at least a tree node* **do**
  s = first node in queue for traversal;
  L_i = list of children attributes of s;
  if L_i *is empty* **then**
    push the attribute name in tree node s into L_i;
  end
  push L_i into list of subtrees;
end

Algorithm 1: Extracting a list of subtrees from a hierarchy.

Input: list of row subtrees, list of column subtrees
Result: list of combinations
list of combinations = empty list;
for R_i *in list of row subtrees* **do**
  for C_j in *list of column subtrees* **do**
    if *data table is not empty at the combination of rows in* R_i *and columns in* C_j **then**
      if *either* R_i *or* C_j *has one element and not both* **then**
        push the data table segment in the combination of R_i and C_j into list of combinations;
      end
    end
  end
end

Algorithm 2: Finding unit combinations.

4.1.3 Linking Text and Visualizations

Simple visualizations—line and bar charts—are generated by extracting unit combinations of rows and columns (Figure 5). However, the number of such combinations and thus, visualizations will quickly increase with more tables in the documents. For instance, even for the small segment of a table in Figure 4, nine charts can be generated. Going through all many such charts is infeasible when the focus is on reading the document. Therefore, the visualizations need to be adapted to the user's interest within the document (R4). This also ensures that the visualizations enhance the data-rich documents by bridging the text and the tables (R3). In our approach, we match extracted words from the text with the attributes and values in a visualization and its parent information from the tables to develop a relevance score.

Word extraction: For a given text, the attribute names and values (typically, a few words and numbers) from the tables should be matched to the context extracted from the document text to find relevant ones to be visualized and highlighted to the reader. We utilize textual preprocessing techniques such as stop word removal, stemming, and lemmatization to extract a set of words that convey the text context. For instance, consider this sentence, "Globally, there were an estimated 287000 maternal deaths in 2010, yielding a MMR of 210 maternal deaths per 1000000 live births among the 181 countries that were covered in this study." This sentence is picked by the reader in Figure 1. Our extraction algorithm identifies the terms: "globally", "maternal", "deaths", "2010", "287000", "yield", "210", "100000", "live", "births', "181","countri", "cover","study." Such base forms of words are matched to the row and column headers and cell values.

Scoring: The set of words from text is matched with the data behind the visualizations including the attribute names, their parents in the hierarchy, and their values. Individual matches are weighted and summed to compute a relevance score. We identified the weights by testing the approach ourselves and in the pilot testing before the evaluation. We used a binary weighing scheme: matches where the words from the text contain both the attribute name and corresponding value for a data item in a visualization weighted as $w_i = 1$, while partial

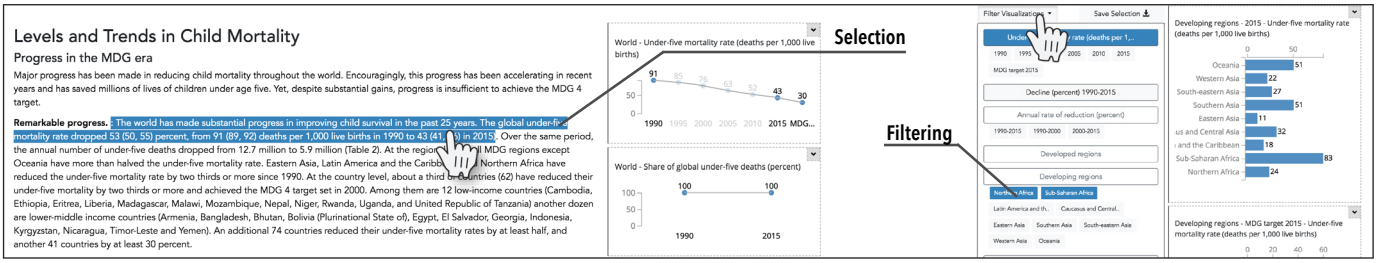


Fig. 6: Two user interactions in the Elastic Documents approach. The reader can select a sentence to see relevant visualizations with keywords highlighted or filter an options menu to see visualizations containing the filtered attributes.

matches between word in text and attribute name or a parent attribute in a table weighted less ($w_i = 0.1$; for the documents in the user study). These matches are exemplified in Figure 1(c) with the first chart in the chart panel showcasing a complete match—both value "210" and year "2010" matching words from the selected text—and partial matches based only on the year appearing down the list. We adapt the visual aids to user's focus in the text by sorting the visualizations based on these relevance scores. Finally, these matches are also highlighted in the data tables to help the reader see the context in the data tables.

4.2 Interface and Interaction Design

The document processing methods promise generation of contextual visualizations that bridge the text being read and the data tables present in the data-rich document. As a proof-of-concept, we developed a simple interface for the Elastic Documents approach as shown in Figure 1 building on aforementioned design rationale. The interface separates the content types from the document and places them in three views—a text view, a chart panel for generated visualizations, and a table view that can be minimized. The text view occupies the majority of the screen space to support reading the document content, while the table and chart panels augment the text by presenting the structured data. The visualizations shown on the chart panel are based on the user selection within the text and the filters or search items set by the user on the top right of the interface. We define two interaction mechanisms within the interface to connect the document content. (Figure 6).

1. **Specify current focus in text** implicitly based on scrolling actions or explicitly by highlighting segments of the text.
2. **Specify current focus in data attributes** to convey interest in specific attributes within the document tables.

Through the former, the user's focus in the text is processed to connect to the data within tables. Relevant visualizations are then shown on the top of the chart panel, along with highlights within the visualizations and the tables (see Figure 6). Through the latter, user's interest in certain data attributes is identified as filters. Filtering on a dimension or a group of dimensions sorts corresponding visualizations to be on the top. Thus, the document is adapted to the user's interest (R4).

Design alternatives. The interface is split into three views to show text, tables, and the contextual visualizations separately. Alternative designs exist including, (1) showing the visualizations and data on demand through tooltips, (2) placing the charts in-situ within the tables, similar to Table Lens [46], or (3) exploring a word-scale approach with charts embedded in the text. While these designs are also interesting, our current interface presents all three content types to the reader and makes the connections between text and tables more explicit, at the cost of extra screen space spent on the chart panel. The alternate designs can fit better in specific scenarios—e.g, in the presence of a limited display space. In terms of interaction design, the interactions described above are focused on presenting relevant data while reading the document text. Apart from them, our prototype also supports pinning a chart to always show it on top of the chart panel and saving user's focus and filters to retrieve them later. Beyond the ones in Elastic Documents, standardized interactions are needed to fully support close reading such as for note taking and annotation of the documents.

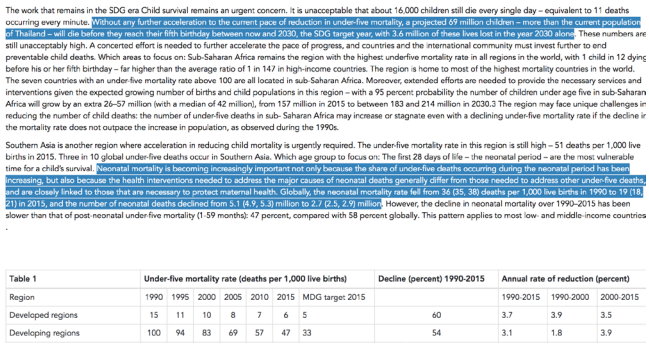


Table 1

	Under-five mortality rate (deaths per 1,000 live births)							Decline (percent) 1990-2015	Annual rate of reduction (percent)		
Region	1990	1995	2000	2005	2010	2015	MDG target 2015		1990-2015	1990-2000	2000-2015
Developed regions	15	11	10	8	7	6	5	60	3.7	3.9	3.5
Developing regions	100	94	83	69	57	47	33	54	3.1	1.8	3.9

Fig. 7: The conventional document viewer interface used in the study. This interface retains the original layout of the document with text and tables at fixed positions and without visualizations. Only "search and find" interaction is available along with the ability to highlight.

4.3 Implementation

Our implementation of Elastic Documents is a proof-of-concept that connects the text and data tables with contextual visualizations. Our tool currently works with the example data-rich documents listed in this paper and therefore, it was used for our user study. It is built with HTML/CSS/JS technologies and D3 framework [8] for visualization. It uses Python as a back-end to extract hierarchies from tables and find combinations and serves them using a Flask server. The text processing is performed using Python's NLTK library [6].

5 User Study

Elastic Documents has the potential to present the data context through simple charts when reading the text. Such an approach promises to help the reader comprehend the document by connecting multiple content types. To understand the effectiveness of this overall approach, we chose to compare against a baseline—-a conventional PDF viewer—for standard reading tasks to maintain the ecological validity.

5.1 Interface Conditions

The study comprised of two conditions: Elastic Documents and Conventional Viewer interfaces. The Elastic Documents interface (Figures 1, 6) allows participants to: (1) view visual representations of the tables within documents, (2) interact with the text by selecting sentences by click or any text content by dragging the mouse, to see relevant visualizations from tables, (3) filter visualizations based on the table attributes using a dedicated drop-down menu, and (4) save selections and pin visualizations for future reference. On the other hand, the conventional viewer (Figure 7) shows the text and tables in a flat layout, with tables presented in their original fixed locations. It allows the user to search for keywords using web browser's search option and highlight parts of the document, similar to a typical document viewer.

5.2 Datasets

Targeting a within-subject design, we chose two different documents for the two interface conditions to eliminate learning effects. We

picked two data-rich documents from public sources on child mortality (CMR) [58] and maternal mortality (MMR) [62]. These documents have similar topics and comparable levels of complexity. Both data-rich documents showcase trends in mortality rates in developing and developed countries in the world and outline goals for the future. As such, they capture the mortality rates and deaths for regions of the world over time within the tables. For the scope of this study, we focused on parts of these documents that outline the levels and trends in mortality data (8-9 paragraphs) along with two tables.

5.3 Participants

We recruited 14 participants (age 22-45; 7 female; 7 male). Participation was voluntary and participants received $10 for their work. Participants were visualization literate with experience in charting with tools such as Excel and Tableau; 4 of them used visualizations for data analysis (for their course or general work). All but one participant (P14) frequently work with documents—reading research papers and articles, or writing reports. Participants worked with both interfaces.

5.4 Tasks

As described earlier, data-rich documents can act as records of factual data, which makes them an information source for journalists and policymakers. Therefore, we identified two primary tasks:

- **Summarization:** Participants worked on one summarization task for each condition. They are asked to summarize the document in 6 sentences. They were instructed to capture the important points and data discussed in the document. Participants were allowed to use any of the aforementioned interactions with the interfaces for this purpose.
- **Comprehension:** Following the summary task, participants answered four questions about the data in the documents within 5 minutes. To answer these questions, participants need to be aware of content of the text and tables to give a comprehensive answer. Some example questions used for these tasks include:
  - Which regions or countries meet their MDG target for 2015 for under-five mortality rate?
  - Which regions or countries reduced their maternal mortality rates by 50% between 1990 and 2010?
  - Which regions or countries have a high lifetime risk of maternal death (units: 1 in)?

  Participants were allowed to use the interactive filtering options to find visualizations of interest and access previously-saved selections during this task.

5.5 Procedure

Each user study session started with signing a consent form and completing a demographic survey. The experimenter first trained participants in the assigned interface by demonstrating the visualizations and interactions for reading the document. The participants were then allowed to train on their own, until they were comfortable with the interface. They then read the assigned document on the interface to write the 6-sentence summary. Following the summary task, they completed the four comprehension tasks. Then, they completed a survey on the perceived usability and utility of the interface. They then moved on to the second condition with the other document and repeated the procedure. The sequence of interfaces and documents was counterbalanced across participants. Sessions lasted one hour each.

5.6 Data Collected

For both tasks, the time was fixed: 20 min for summarization and 5 minutes for the four comprehension questions on each interface. The following data was collected:

- Accuracy in answering the comprehension questions.
- The summary of the documents.
- Interactions with the text while reading the documents.
- Think-out-loud feedback.
- Subjective feedback.

Each participant wrote down the answers in a provided document. The subjective feedback was collected through a questionnaire. Finally, the interactions performed on the interfaces were recorded, along with the audio, to understand the usage patterns.

5.7 Pilot Studies

The comprehension questions, as well as the time assigned for the tasks, were decided through two pilot studies conducted within our research group. After the first pilot, which lasted two hours without the time limits on the summarization task, we decided to fix it to 20 minutes. For comprehension tasks, a 5-minute time limit was chosen to ensure that the participants do not completely reread the documents.

5.8 Data Analysis

While the accuracy of answers can be verified for comprehension, there are no established metrics to evaluate the summaries. Instead of a pure qualitative analysis, we use two derived measures from the summary that exemplify the role of data in these documents and represent document reading tasks. A summary should cover the main topics in the document. Therefore, we chose breadth of summary to observe the differences between the two conditions. The main topics were manually identified by the study investigators and verified during the pilot studies. Furthermore, the text in the two data-rich documents convey a sample of the data and the user understanding should therefore be developed from both text and tables to better understand the main insights from the document. For this reason, we used number of statements in the summary quoting data from tables, developed from visualizations or by viewing tables, as the second measure. We call these statements, *data-rich* statements.

Considering recent concerns with null-hypotheses testing [18, 20], APA recommendations [60], and an example set by the visualization community [19, 25], we focused on estimation techniques to report on effect sizes and confidence intervals, instead of p-value statistics. All point estimates and 95% confidence intervals (CI) are based on 1000 percentile bootstrap replicates.

6 Results

Here, we report the results followed by our qualitative observations.

6.1 Summary: Breadth of Topics

Our primary derived measure was the number of topics from the original document within the summaries. For both documents, these topics include "*comparison of trends over time,*" "*mortality rates of developing and developed regions,*" "*improvements for countries,*" "*projections for the future,*" "*challenges for countries,*" and "*correlation to other indicators.*" By tagging these topics, we compared Elastic Documents with the conventional viewer (Figure 8(left)). The effect of the interface is strong as the size of the non-overlapping region between the two interfaces is more than 80% for both documents (showcasing a large effect [55]). Overall, this indicates that Elastic Documents (Mean 4.65; CI [4.07, 5.21]) increased the topics discussed compared to the conventional viewer (Mean 3.37; CI [2.86, 3.86]).

6.2 Summary: Number of Data-Rich Statements

We tag a statement in the summary as a data-rich statement if it satisfies one of the following conditions: (1) the statement contains precise numerical facts, (2) the participant pinned a chart and used it to write the statement, (3) the participant consolidated multiple charts to augment the text, or (4) used the data from the table to write it. Figure 8 (middle) showcases the effect sizes for Elastic Documents vs. conventional viewer overall as well as for each of the two documents. Overall, summaries from Elastic Documents (Mean 3.73; CI [3.29, 4.21]) contained more data-rich statements than the conventional interface (Mean 1.29; CI [0.79, 1.86]). This effect is also strong for each individual document, but appears to be at different levels. Example data-rich statements made by participants include:

"*[The] majority of the maternal deaths can be found in 3 developing regions, namely: Sub-saharan Africa, Southern Asia, and Southeastern Asia. Of these 3, Southern Asia and Eastern Asia, had the*

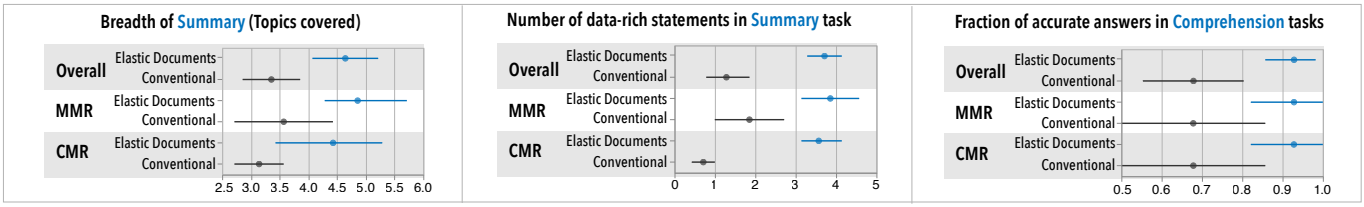


Fig. 8: Point estimates and 95% confidence intervals for the derived measures from the user study.

highest % change in MMR between 1990 and 2010 at 69% and 64% respectively." (P1)

"*Sub-saharan Africa remains to be a region that needs to improve more, since it also has one of the lower % change in MMR between 1990 and 2010, at 41%.*" (P3)

"*This acceleration [in the improvements in Under-five Mortality rates] is mostly due to developing regions, as the rates over the same two periods [1990-2000, 2000-2015] in developed regions actually slowed, from 3.9% per year to 3.5% per year.*"(P8)

6.3 Comprehension: Accuracy of Answers

Following the summary tasks, participants answered questions regarding the data within the document. Again, there is evidence that Elastic Documents (Mean 0.93; CI [0.86, 0.98]) outperformed the alternative (Mean 0.68; CI [0.55, 0.80]) overall (Figure 8(right)); however, the evidence is slightly weaker for individual documents, and the variance in accuracy is higher for the conventional viewer.

6.4 Qualitative Observations

The text within the chosen data-rich documents provides both high-level discussions on trends and projections in mortality rates as well as details related to specific countries in the developing regions. For the high-level content, the tables provide detailed data to support the claims. For detailed discussions, the tables provide aggregate estimates at regional levels. Here we describe the reading styles in working around these aspects in the presence and absence of visualizations, and describe our observations about the role of visualization.

6.4.1 Reading Styles

Elastic Documents is meant for close reading, where the reader has the option read and reread the document to develop an understanding. This reading pattern is very conducive to visualization as the reader can perceive and interpret simple visualizations when going through the document, in contrast to speed reading or browsing a document.

On the conventional interface. Majority of the participants adopted either a skimming reading style or a back-and-forth style: 8 participants read the document in a cursory style to quickly go through the entire content to extract important points, and 4 participants went back and forth between text and tables based on the references in the text to gain a holistic understanding. For the skimming style, some participants never looked at the tables (P6, P7) or missed some attributes in the table (P9). They were then compelled to read parts of the documents again when writing the summary or answering comprehension tasks. For the back-and-forth style, the participants focused on a few attributes and referred to the tables repeatedly so that they do not lose their understanding of the textual content.

The participants complained that about the layout and content of the document in the conventional viewer: "*The document is not very easy to read. There's so much information and data, with only two tables far away from the paragraphs*" (P8), and "*It was hard to move the page up and down to see the table and going back to the point that I was reading*" (P2). Based on their comments, we believe that some participants in this condition felt a missing aspect when reading these documents: when read in a skimming style, they lost track of details and found it hard to understand the statements in text, and when read closely, they felt it was too cognitively challenging.

To deal with this problem, two participants tried a third reading style (P5, P14; who began their experiment with Elastic Documents). They started reading by directly going through the two tables. Both participants found the task easier than others, but starting with tables and connecting them to text is not sensible especially when the tables become larger (cf. Hurricane Harvey report [42]).

On the Elastic Documents interface. All participants started by observing the visualizations while reading the document. 9 participants incorporated text highlighting into their reading style: (1) participants P1, P2, P4, and P9 selected every sentence while reading and observed the top of the chart panel, and (2) the other participants selected only sentences that have some data or insight. P2 said, "*The interface automatically shows relevant information when I click a sentence, so it sometimes showed me useful information to understand the text.*" For a few participants, selecting sentences one after another seemed to be second nature they acquired in past digital reading experiences. The selection was carried out to track their progress with no deliberation.

During summarization, after developing a preliminary idea of the main topics of the document, five participants (P5, P8, P9, P11, P12) went beyond text selection, by filtering on the table attributes to curate the visualizations to their interests. We characterize this as a *hybrid* reading pattern, where the reader progressively transforms the reading into a more exploratory method. Participants used the line charts to observe trends for the attribute (e.g., maternal deaths) mentioned in the text during other years, and used the bar charts to observe extrema in the statistics, along with reading exact values. Only two participants accessed the actual tables when reading the data-rich documents to gain an overview of all the attributes.

6.4.2 Role of Visualizations in Document Reading

The visualizations played different roles based on the sentence being read and the user's intent. Participants primarily found that the charts augmented the text by showing both the corresponding data and the contextual data. For instance, when reading about Sub-Saharan Africa, the charts included not only data about that region but also information about other regions. They realized this region had the highest mortality rate, which helped them remember the sentence. Sometimes, participants would go through one or two charts before even reading the sentences to observe the data context. After all, visualizations leverage the full capabilities of the high-bandwidth sense of human vision. This complementary nature of visualizations may have helped in document reading to understand the context and develop a more comprehensive summary for the data-rich documents.

In cases when visualizations were not complementary (maybe due to the lack of good keywords for matching), some participants used filters to highlight specific visualizations to learn the information. Filters were especially useful when writing the summary as they enabled participants to quickly access the specific data values and to quote from the charts they recalled. Visualization along with interaction in this case provided a means to quickly search for information. Since filtering breaks the flow of reading in some cases, the participants wanted the interface to adapt to their interactions. P8 suggested that his previously filtered attributes can be weighed higher for better matches.

When the visualizations were indeed relevant to a selected sentence, the participants tended to pay more attention to the sentence and more likely would pin the charts to refer to them when writing the summary.

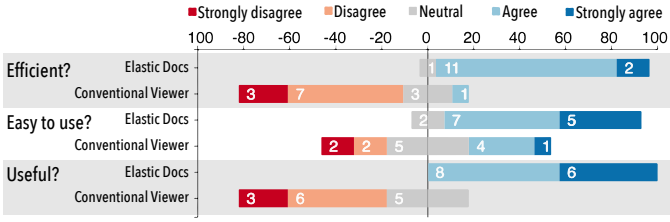


Fig. 9: Participants felt that Elastic Documents was more efficient and useful for reading data-rich documents than the conventional viewer.

6.5 Subjective Ratings

After each session, the participants rated the interfaces on three metrics: efficiency, ease of use, and usefulness for reading data-rich documents, on a Likert scale ranging from 1 (e.g., strongly disagree) to 5 (e.g., strongly agree). Figure 9 shows the subjective ratings provided by the participants. Elastic Documents outperformed conventional interface in the efficiency and usefulness scale, with almost all participants agreeing that the presence of visualizations aids in reading data-rich documents. For ease of use, they felt that the familiarity of the conventional viewer made it also easy to use (P14). When asked to choose one, all participants preferred Elastic Documents.

Finally, while we have not evaluated recall, participants also felt visualizations added to the memorability of the data in the document. P9 said, "*Visuals created by elastic documents helped me memorize the facts, also it is easy to search for visualizations and pin the important visuals.*" However, this needs to be validated further.

7 Discussion

The positive results from the user study as well as the qualitative observations confirm the benefits of Elastic Documents. Here we discuss our results, including their implications as well as their limitations.

7.1 Limitations and Future Work

Chart types and text processing. Our Elastic Documents technique for connecting the text and tables focuses on utilizing visualizations to simplify the complex data tables based on the user's focus. To showcase this approach, we chose to utilize two chart types, as well as word extraction and matching algorithms that can apply this approach to the sample documents. To extend this approach to wider use, further research is required to isolate other visual representations—geographical maps, multi-category charts (e.g., stacked bars), and set visualizations—that can aid in reading data-rich documents. Furthermore, more advanced algorithms for keyword extraction and matching—e.g., using word embedding [39]—need to be identified to make the connection between text and tables stronger. Given the benefits of the Elastic Documents approach, this will be our next step in extending the technique.

Current implementation. We focused on presenting a proof-of-concept of our approach to connect text and tables within the document using contextual visualizations. As such, it is currently applied to sample documents available online in PDF format. While doing so, we also relied on an internal tool developed at Adobe that tags elements in a PDF document and converts them to a Document Object Model, which was amenable to the table parsing and extraction methods introduced in this paper. Our approach is not limited to PDF documents, however, and can be applied to any document format. To handle other formats, customized document parsers are necessary.

Parsing performance. By developing on previous research [16], we did not evaluate the low-level performance of the table parsing methods. According to Chen and Cafarella [15], the lower bound of the precision of the parsing algorithm is around 81% and the lower bound for recall is around 73% for similar spreadsheets. Errors in table parsing can significantly impact the subsequent visualization generation and linking stages. For example, misidentification of the header structure may result in incorrect table segments and visualization; the match scores may also be inaccurate, which may produce nonsensical order of visualizations. We alleviate these problems partially by supporting search and filtering of the generated visualizations. As a next step, we are investigating techniques that increase the transparency of parsing and allow users to interactively rectify potential errors.

Study design. Our evaluation compared two conditions: Elastic Documents vs. a baseline. Our goal in doing so was to compare the Elastic Documents approach to the current interface for data-rich documents. We chose this holistic evaluation method, involving an entirely new document viewer, to ensure ecological validity. It is certainly possible to evaluate intermediate versions of Elastic Documents that isolate the effects of visualization, layout manipulation, and contextual presentation of data. Another interesting idea is to improve our approach by drawing inspiration from the analog paper medium. However, these directions are outside the scope of this paper and left for future studies.

7.2 Implications

PDF documents could be seen as a dead end for open data, since they are not as suitable for analytics as other serialized formats such as JSON or CSV. However, many organizations—both public and private—publish data-rich documents in the PDF format as it provides a self-contained medium for data-driven storytelling or data archival. Adaptive document viewers have the potential to alleviate the challenges in reading such documents. They can view the data-rich documents in a user interface that fits the reader's specific goal and task. For instance, during cursory reading, many visualizations can be shown to increase the bandwidth, while in active reading, detailed text can be provided along with the visualizations to support annotation.

Beyond document viewers, Elastic Documents has implications to document creation. New document creation tools can be introduced to help creators add more complete metadata and contextual information to data-rich documents. This can enable better accuracy in adaptive document viewing by eliminating error-prone parsing and inference. One way to realize this idea is to support tagging connected information across content types—not just text and tables but also illustrations and even dynamic content such as animations and videos [28]—semi-automatically through user interaction. For instance, when writing a textual narrative, the creation tool can suggest references to data from tables or parts of figures and insert them based on some user input.

8 Conclusion

In this paper, we introduce the Elastic Documents approach for data-rich documents. These documents contain multiple content types including text, figures, and complex tables spreading many pages. As a part of this approach, we extract the tables and text from these documents and parse the tables to generate visualizations. These visualizations containing structured data are matched with the user's focus in the text to present the relevant ones. By doing so, the output format of a document is made adaptive to the user's specific interest.

We evaluated our approach against a conventional document viewer in a laboratory study. Participants covered more topics in their summaries and covered more data-rich statements when using Elastic Documents. The qualitative observations showcase unique reading patterns in the presence of visualizations. While our current implementation is just a prototype, we are currently working towards a more general viewer inspired by Elastic Documents that can connect multiple content types together—rich text, tables, illustrations, and pictures—and simplify the reading of data-rich documents.

Acknowledgments

We thank the anonymous reviewers, Catherine Plaisant, and Clemens N. Klokmose for their valuable feedback that substantially improved this manuscript. Majority of the ideation and technical development for this research work was done during the first author's summer internship at Adobe Research in Seattle. This work was partially supported by the U.S. National Science Foundation award IIS-1539534. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the funding agency.

REFERENCES

[1] Adobe Acrobat. https://acrobat.adobe.com/us/en/.

[2] Adobe InDesign. www.adobe.com/products/indesign.html.

[3] E. Adar, C. Gearig, A. Balasubramanian, and J. Hullman. PersaLog: Personalization of news article content. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*, pp. 3188–3200. ACM, New York, NY, USA, 2017. doi: 10.1145/3025453.3025631

[4] Apple. SEC filings. http://investor.apple.com/sec.cfm?DocType=Quarterly.

[5] J. Bertin. *Sémiologie graphique: Les diagrammes-Les réseaux-Les cartes*. Gauthier-VillarsMouton & Cie, 1973.

[6] S. Bird, E. Klein, and E. Loper. *Natural Language Processing with Python*. O'Reilly Media, Inc., 1st ed., 2009.

[7] P. Blenkhorn, G. Evans, A. King, S. H. Kurniawan, and A. Sutcliffe. Screen magnifiers: Evolution and evaluation. *IEEE Computer Graphics and Applications*, 23(5):54–61, 2003. doi: 10.1109/MCG.2003.1231178

[8] M. Bostock, V. Ogievetsky, and J. Heer. D^3: Data-driven documents. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2301–2309, 2011. doi: 10.1109/TVCG.2011.185

[9] N. Boyles. Closing in on close reading. *On Developing Readers: Readings from Educational Leadership, EL Essentials*, pp. 89–99, 2012.

[10] U. Brandes, B. Nick, B. Rockstroh, and A. Steffen. Gestaltlines. *Computer Graphics Forum*, 32(3pt2):171–180, 2013. doi: 10.1111/cgf.12104

[11] S. K. Card, J. D. Mackinlay, and B. Shneiderman. *Readings in Information Visualization: Using Vision to Think*. Morgan Kaufmann, 1999.

[12] S. Chandrasegaran, S. K. Badam, L. Kisselburgh, K. Ramani, and N. Elmqvist. Integrating visual analytics support for grounded theory practice in qualitative text analysis. *Computer Graphics Forum*, 36(3):201–212, 2017. doi: 10.1111/cgf.13180

[13] B.-W. Chang, J. D. Mackinlay, P. T. Zellweger, and T. Igarashi. A negotiation architecture for fluid documents. In *Proceedings of the ACM Symposium on User Interface Software and Technology*, pp. 123–132. ACM, New York, NY, USA, 1998. doi: 10.1145/288392.288585

[14] K. S.-P. Chang and B. A. Myers. Using and exploring hierarchical data in spreadsheets. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*, pp. 2497–2507. ACM, New York, NY, USA, 2016. doi: 10.1145/2858036.2858430

[15] Z. Chen and M. Cafarella. Automatic web spreadsheet data extraction. In *Proceedings of the ACM Workshop on Semantic Search Over the Web*, pp. 1:1–1:8. ACM, New York, NY, USA, 2013. doi: 10.1145/2509908.2509909

[16] Z. Chen and M. Cafarella. Integrating spreadsheet data via accurate and low-effort extraction. In *Proceedings of the ACM Conference on Knowledge Discovery and Data Mining*, pp. 1126–1135. ACM, New York, NY, USA, 2014. doi: 10.1145/2623330.2623617

[17] Z. Chen, M. Cafarella, J. Chen, D. Prevo, and J. Zhuang. Senbazuru: A prototype spreadsheet database management system. *Proceedings of the VLDB Endowment*, 6(12):1202–1205, Aug. 2013. doi: 10.14778/2536274.2536276

[18] G. Cumming. The new statistics: Why and how. *Psychological Science*, 25(1):7–29, 2014. doi: 10.1177/0956797613504966

[19] E. Dimara, A. Bezerianos, and P. Dragicevic. The attraction effect in information visualization. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):471–480, 2017. doi: 10.1109/TVCG.2016.2598594

[20] P. Dragicevic, F. Chevalier, and S. Huot. Running an HCI experiment in multiple parallel universes. In *Extended Abstracts of the ACM Conference on Human Factors in Computing Systems*, pp. 607–618. ACM, 2014.

[21] N. K. Duke and P. D. Pearson. Effective practices for developing reading comprehension. *The Journal of Education*, 189(1/2):107–122, 2008. doi: 10.1177/002205740918900l-208

[22] T. Gao, J. R. Hullman, E. Adar, B. Hecht, and N. Diakopoulos. NewsViews: an automated pipeline for creating custom geovisualizations for news. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*, pp. 3005–3014. ACM, New York, NY, USA, 2014. doi: 10.1145/2556288.2557228

[23] A. M. Glenberg and W. E. Langston. Comprehension of illustrated text: Pictures help to build mental models. *Journal of Memory and Language*, 31(2):129–151, 1992. doi: 10.1016/0749-596X(92)90008-L

[24] P. Goffin, J. Boy, W. Willett, and P. Isenberg. An exploratory study of word-scale graphics in data-rich text documents. *IEEE Transactions on Visualization and Computer Graphics*, 23(10):2275–2287, 2017. doi: 10.1109/TVCG.2016.2618797

[25] P. Goffin, W. Willett, A. Bezerianos, and P. Isenberg. Exploring the effect of word-scale visualizations on reading behavior. In *Extended Abstracts on Human Factors in Computing Systems*, pp. 1827–1832. ACM, 2015. doi: 10.1145/2702613.2732778

[26] P. Goffin, W. Willett, J.-D. Fekete, and P. Isenberg. Exploring the placement and design of word-scale visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):2291–2300, 2014. doi: 10.1109/TVCG.2014.2346435

[27] X. Gregg. Introducing packed bars, a new chart form. https://community.jmp.com/t5/JMP-Blog/Introducing-packed-bars-a-new-chart-form/ba-p/39972, June 2017.

[28] T. Grossman, F. Chevalier, and R. H. Kazi. Your paper is dead!: Bringing life to research articles with animated figures. In *Extended Abstracts of the ACM Conference on Human Factors in Computing Systems*, pp. 461–475. ACM, New York, NY, USA, 2015. doi: 10.1145/2702613.2732501

[29] V. Gyselinck and H. Tardieu. The role of illustrations in text comprehension: what, when, for whom, and why? In *The Construction of Mental Representations During Reading*, pp. 195–218. Lawrence Erlbaum Associates Publishers, 1999.

[30] W. R. Harris and S. Gulwani. Spreadsheet table transformations from examples. *ACM SIGPLAN Notices*, 46(6):317–328, 2011. doi: 10.1145/1993316.1993536

[31] J. Hullman and N. Diakopoulos. Visualization rhetoric: Framing effects in narrative visualization. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2231–2240, 2011. doi: 10.1109/TVCG.2011.255

[32] C. Jacobs, W. Li, E. Schrier, D. Bargeron, and D. Salesin. Adaptive document layout. *Communications of the ACM*, 47(8):60–66, 2004. doi: 10.1145/1012037.1012063

[33] S. Jänicke, G. Franzini, M. F. Cheema, and G. Scheuermann. On close and distant reading in digital humanities: A survey and future challenges. In *Eurographics Conference on Visualization (EuroVis)-STARs. The Eurographics Association*, 2015. doi: 10.2312/eurovisstar.20151113

[34] S. Kandel, A. Paepcke, J. Hellerstein, and J. Heer. Wrangler: Interactive visual specification of data transformation scripts. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*, pp. 3363–3372. ACM, New York, NY, USA, 2011. doi: 10.1145/1978942.1979444

[35] L. Klein. Social network analysis and visualization in 'the papers of thomas jefferson'. *Proceedings of the Digital Humanities*, 4(9):12, 2012.

[36] N. Kong, M. A. Hearst, and M. Agrawala. Extracting references between text and charts via crowdsourcing. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*, pp. 31–40. ACM, New York, NY, USA, 2014. doi: 10.1145/2556288.2557241

[37] J. H. Larkin and H. A. Simon. Why a diagram is (sometimes) worth ten thousand words. *Cognitive Science*, 11(1):65–100, 1987. doi: 10.1111/j.1551-6708.1987.tb00863.x

[38] J. Mackinlay, P. Hanrahan, and C. Stolte. Show me: Automatic presentation for visual analysis. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1137–1144, 2007. doi: 10.1109/TVCG.2007.70594

[39] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[40] F. Moretti. *Distant reading*. Verso Books, 2013.

[41] National Hurricane Center. Tropical hurricane report: Hurricane Alex. https://www.nhc.noaa.gov/data/tcr/AL012016_Alex.pdf, 2016.

[42] National Hurricane Center. Tropical hurricane report: Hurricane Harvey. https://www.nhc.noaa.gov/data/tcr/AL092017_Harvey.pdf, 2017.

[43] E. Oro and M. Ruffolo. Trex: An approach for recognizing and extracting tables from pdf documents. In *Proceedings of the International Conference on Document Analysis and Recognition,*, pp. 906–910. IEEE, 2009. doi: 10.1109/ICDAR.2009.12

[44] C. Perin, P. Dragicevic, and J.-D. Fekete. Revisiting Bertin matrices: New interactions for crafting tabular visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):2082–2091, 2014. doi: 10.1109/TVCG.2014.2346279

[45] R. A. Pielke Jr, J. Gratz, C. W. Landsea, D. Collins, M. A. Saunders, and R. Musulin. Normalized hurricane damage in the United States: 1900–2005. *Natural Hazards Review*, 9(1):29–42, 2008.

[46] R. Rao and S. K. Card. The Table Lens: merging graphical and symbolic representations in an interactive focus+context visualization for tabular information. In *Proceedings of the ACM Conference on Human Factors*

in Computing Systems, pp. 318–322. ACM, 1994. doi: 10.1145/191666.191776

[47] S. A. Sabab and M. H. Ashmafee. Blind reader: An intelligent assistant for blind. In *Proceedings of the International Conference on Computer and Information Technology*, pp. 229–234. IEEE, 2016. doi: 10.1109/ICCITECHN.2016.7860200

[48] B. Shneiderman. Tree visualization with tree-maps: 2-D space-filling approach. *ACM Transactions on Graphics*, 11(1):92–99, 1992. doi: 10.1145/102377.115768

[49] R. L. Siegel, K. D. Miller, and A. Jemal. Cancer statistics, 2015. *CA: A Cancer Journal for Clinicians*, 65(1):5–29, 2015.

[50] R. L. Siegel, K. D. Miller, and A. Jemal. Cancer statistics, 2016. *CA: A Cancer Journal for Clinicians*, 66(1):7–30, 2016.

[51] R. Spence. *Information Visualization*. Springer, 2001.

[52] J. Stasko and E. Zhang. Focus+context display and navigation techniques for enhancing radial, space-filling hierarchy visualizations. In *Proceedings of the IEEE Symposium on Information Visualization*, pp. 57–65. IEEE, 2000. doi: 10.1109/INFVIS.2000.885091

[53] L. Stearns, R. Du, U. Oh, Y. Wang, L. Findlater, R. Chellappa, and J. E. Froehlich. The design and preliminary evaluation of a finger-mounted camera and feedback system to enable reading of printed text for the blind. In *Proceedings of the European Conference on Computer Vision*, pp. 615–631. Springer, 2014. doi: 10.1007/978-3-319-16199-0_43

[54] H. Strobelt, D. Oelke, C. Rohrdantz, A. Stoffel, D. A. Keim, and O. Deussen. Document cards: A top trumps visualization for documents. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):1145–1152, 2009. doi: 10.1109/TVCG.2009.139

[55] G. M. Sullivan and R. Feinn. Using effect size—or why the P value is not enough. *Journal of Graduate Medical Education*, 4(3):279–282, 2012. doi: 10.4300/JGME-D-12-00156.1

[56] C. S. Tashman and W. K. Edwards. LiquidText: a flexible, multitouch environment to support active reading. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*, pp. 3285–3294. ACM, New York, NY, USA, 2011. doi: 10.1145/1978942.1979430

[57] J. W. Tukey. *Exploratory Data Analysis*. Addison-Wesley, 1977.

[58] UNICEF. Levels and trends in child mortality, 2015. `https://data.unicef.org/resources/levels-and-trends-in-child-mortality-2015/`.

[59] F. Van Ham, M. Wattenberg, and F. B. Viégas. Mapping text with Phrase Nets. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):1169–1176, 2009. doi: 10.1109/TVCG.2009.165

[60] G. R. VandenBos. Publication manual of the American Psychological Association. `http://www.apastyle.org/manual/`, 2016.

[61] F. B. Viegas, M. Wattenberg, and J. Feinberg. Participatory visualization with Wordle. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):1137–1144, 2009. doi: 10.1109/TVCG.2009.171

[62] World Health Organization. Trends in maternal mortality: 1990 to 2010, 2012. `http://www.who.int/reproductivehealth/publications/monitoring/9789241503631/en/`.

[63] M. A. Yalçin, N. Elmqvist, and B. B. Bederson. Raising the bars: Evaluating treemaps vs. wrapped bars for dense visualization of sorted numeric data. In *Proceedings of the Graphics Interface Conference*, pp. 41–49. Canadian Human-Computer Communications Society / ACM, 2017. doi: 10.20380/GI2017.06

[64] B. Yildiz, K. Kaiser, and S. Miksch. pdf2table: A method to extract table information from PDF files. In *Proceedings of the Indian International Conference on Artificial Intelligence*, pp. 1773–1785, 2005.

[65] D. Yoon, N. Chen, and F. Guimbretière. TextTearing: opening white space for digital ink annotation. In *Proceedings of the ACM Symposium on User Interface Software and Technology*, pp. 107–112. ACM, New York, NY, USA, 2013. doi: 10.1145/2501988.2502036

[66] P. T. Zellweger, S. H. Regli, J. D. Mackinlay, and B.-W. Chang. The impact of fluid documents on reading and browsing: An observational study. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*, pp. 249–256. ACM, New York, NY, USA, 2000. doi: 10.1145/332040.332440