

# Echo: The Editor's Wisdom with the Elegance of a Magazine

Joshua Hailpern & Bernardo Huberman

HP Labs

1501 Page Mill Rd, Palo Alto, CA 94304

{joshua.hailpern,bernardo.huberman}@hp.com

## ABSTRACT

The explosive growth of user generated content, along with the continuous increase in the amount of traditional sources of content, has made it extremely hard for users to digest the relevant pieces of information that they need to pay attention to in order to make sense of their needs. Thus, solutions are needed to help both professionals (e.g. lawyers, analysts, economists) and ordinary users navigate this flood of information. We present a novel interaction model and system called Echo which uses machine learning techniques to traverse a corpus of documents and distill crucial opinions from the collective intelligence of the crowd. Based on this analysis, Echo creates an intuitive and elegant interface, as though constructed by an editor, that allows users to quickly find salient documents and opinions, all powered by the wisdom of the crowd. The Echo UI directs the user's attention to critical opinions using a natural magazine style metaphor, with visual call outs and other typographic changes. Therefore, this paper presents two key contributions (an algorithm and interaction model) that allow a user to "read as normal," while focusing her attention on the important opinions within documents, and showing how these opinions relate to those of the crowd.

## Author Keywords

opinions; alignment; documents; collective intelligence; typography; call out; reading; economics of attention

## ACM Classification Keywords

H.5.m. Information Interfaces and Presentation (e.g. HCI): Miscellaneous

## INTRODUCTION

In the information age, few documents live in a vacuum, especially documents that express opinions or evaluations. Knowledge workers (e.g., political analysts, financial investors, economists, CEOs), legal professionals (e.g., clerks, lawyers, judges) and other consumers of topical information have to navigate hundreds, or perhaps thousands of opinion documents and meet the challenge of sorting out the relevant information that actually requires attention. The addition of

user-generated content to this mix has added tremendous diversity to this ecology of documents, which, while empowering at some level, also leads to a poverty of attention [38, 45]. This implies that potentially relevant content is missed when foraging for information among large corpuses of interlinked documents.

Solutions must therefore be created to help the recipients of this information "allocate their attention effectively among the overabundance of information sources that might consume it" [38]. This problem is known as the economics of attention, and directly relates to how users access key information in an ever-growing digital information space [22, 23, 1]. This problem has two main challenges: how to browse and find the most salient documents in a large corpus, and how to quickly find critical opinions within each document and situate them within the "big picture." Three critical limitations of the broad class of existing solutions pertaining to document and opinion browsing are that they do not situate each document's opinions in a broader context, they do not present key opinions in a natural manner, and they do not provide a corpus browsing UI that focuses on finding the most representative or out-of-line documents with respect to crucial opinions. Furthermore, many of the existing software solutions add to the information overload by presenting the user with more information (e.g. multiple documents side by side) in increasingly smaller areas of screen real-estate.

In response, we created a novel interaction model, algorithm and system called **Echo** that enables a user to easily traverse a corpus of documents and naturally read individual documents while maintaining a broader context. Echo utilizes a rich series of Machine Learning techniques that we refer to as the **Echo System**. The Echo System examines a large corpus of documents, extracts the content and gathers the collective opinions and intelligence of the crowd on key topics. In addition, we developed two key User Interfaces for Echo: the **Echo Locator**, which shows the user how each *document* fits in with the document database; and the **Echo Reader**, which allows the user to see how each *opinion in a document* fits into the opinions of the crowd. The Echo Locator allows users to peruse a corpus and find documents that have opinions (on a specific topic or overall) which are representative or out-of-line as compared to the crowd. The Echo Reader UI calls the user's attention to critical opinions using a natural magazine style metaphor, with visual call outs and other typographic changes. This technique allows a user to "read as normal," while focusing her attention on important opinions that have a very small or very large difference in opinion as compared to the crowd. Echo has been tested on multiple data sets to

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

ICIS'13, June 24–27, 2013, London, United Kingdom.

Copyright 2013 ACM 978-1-4503-2138-9/13/06...\$15.00.

ensure the breath of its flexibility, including political news, financial reports, and movie reviews.

The foremost contribution of this work is an algorithm and interaction technique (demonstrated in a functional system) that allows the user to focus on the most important opinions in any article, quickly analyze and synthesize a document corpus, and see how each article and opinion fits into the broader context in a natural way. To the authors' knowledge, Echo is the first UI system to present opinions by alignment or divergence from the sentiment distribution on that specific topic/word. We first discuss how Echo builds on the existing literature and is situated within the broader context of solutions. We then present Echo's features and implementation, followed by very positive results from a preliminary pilot study we conducted, and conclude with a discussion of future ongoing work.

## RELATED WORK

While much work in the machine learning community has focused on creating algorithms for document summaries[4, 29, 40], Echo is not a summarization tool. Rather, this paper focuses on user interaction design, and systems to support the navigation of opinions within a corpus. In this section, we discuss literature that explores *how* people read, and some existing UI solutions to help people navigate a large document or opinion space.

### Economics of Attention

While the original relationship between attention and information was described in 1971 [38], numerous research since then has continued to explore, expand and quantify his observations across many disciplines, including the digital domain [22]. Based on these findings, some research has been done to explore how to adapt information presentation based on these economics of attention. We therefore wish to build on their findings and further increase the speed and access of users to large document corpuses and the key opinions within them.

### Reading & Document Corpuses

The rich set of real-world activities that surround document corpuses can be organized into four groups of interactions: reading, annotating, collaborating, and authoring[11]. While issues of annotation, collaboration, and authoring are important problems, this work focuses specifically on the economics of attention [38] as it pertains to exploring the overwhelming collection of documents in a corpus and situating each document/opinion within a broader context.

We therefore must consider not just *what* information should be brought to the user's attention, but *how* to do so in the most natural manner. When users move within and between documents, they begin with a critical "planning phase" [32]. While planning, users quickly scan through a document to get an overall sense of structure and important facts, in order to connect key "bits" of information. When the medium of paper is used to present information, the physical layout of content on a page leads to incidental memory (e.g. remembering a key fact as being on a specific page, and specific column)[32]. These contextual cues [9] actively support

this planning phase, which can help users serendipitously encounter information through browsing[30]. In print media, editors actively take advantage of typographic changes and other visual callouts to call readers' attention to what they deem as key content. We can leverage visual callouts can help users navigate by bringing key opinions to their attention while working within a reader's natural planning phase.

In online media, this information recall is paralleled with textual relationship to navigation UI and pictures [16, 32]. However, reading online documents brings up new limitations. While paper readers commonly compare and contrast content by physically arranging documents or flipping back and forth, presenting documents side-by-side on a computer is problematic in that the reduced screen real-estate makes it more difficult to traverse content [32]. Further, reading documents in parallel (rather than jumping between them) may lead to issues of practice inference in working memory [24]. We can therefore see that there is a need to create a method for allowing a user to read a single digital document without losing context. Furthermore this solution should follow a known style metaphor (e.g. magazine callouts) because using a metaphor on top of new technology is a powerful way of conceptualizing interaction, especially for novice users [16].

### Reading Documents in Context

Providing context for a document in a broader space is a challenging technological problem. For example Souneil Park [34, 35] has attempted to explore bias in news events by showing pairs of documents on contrasting ends of a political spectrum side-by-side. While this technique helps with directly comparing two *known* contrasting opinions (via two documents), it makes no affordances for bringing forward critical information, since all information is at the same visual level). Nor does it help place the key opinions within a larger context or present the full "large" corpus of opinions, as the technique just shows the user *one* contrasting opinion. Further, by showing contrasting documents side-by-side, the literature suggests that this UI decision makes it more difficult to traverse content[32].

Other tools such as AKTiveMedia [13], Fishnet [7], and others [37, 39, 8]) use vibrant color highlighting within a document to draw users' attention to keywords based on queries issued. While this does draw a user's attention to key information, it is visually jarring and can be distracting, adding to issues to attention rather than resolving them. However, given the generally highly promising results of these systems, we seek to build upon their work by combining their successful use of highlighting with the metaphor of magazines.

Abstracting documents to a corpus level reveals a class of tools that help users organize a document space. Popcorn [14] and many other tools [20] employ a wide range of visualizations to help users traverse a collection based on hierarchical semantic relationships (e.g. tree maps, network graphs). However, many of these visualization styles can be confusing [20, 12]. For example, Tag Clouds are actively used in many systems (e.g. [26, 46, 18, 28, 27]) and are intended to provide a high level view of a document space. However, evidence suggests that while they may be aesthetically pleasing, from

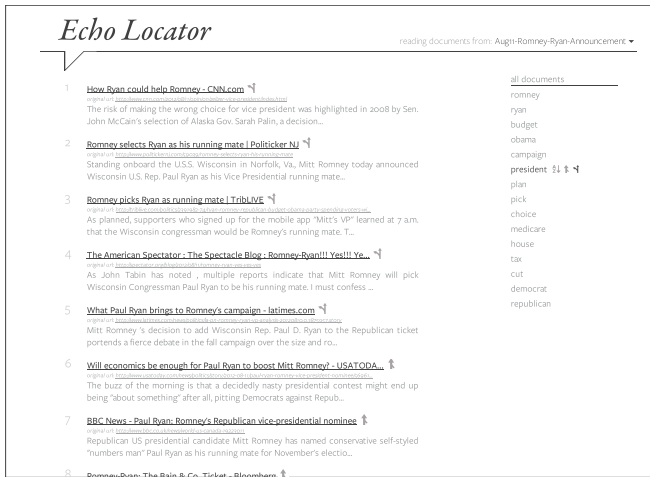


Figure 1: Screenshot of User Interacting with Echo Locator  
Screenshot shows documents that have opinions on the subject “President” sorted by how Out-Of-Line their opinions on the president are. Corpus can be switched by clicking the corpus name in the top right corner of the UI.

a content consumption perspective they are unusable, misrepresentative, and inferior to simple sorted lists [21, 26, 36, 19]. An alternative Tag Cloud visual summarizations are systems that utilize textual summaries/synopsis[29, 25] or key quote lists (e.g. Yelp or Amazon.com) to provide smoother, more natural means of conveying key critical information and assisting users in finding salient information. While text summaries are useful for generally understanding a corpus at a high level, they do not help users trace those summaries back to specific documents/opinions; nor do they allow users to traverse a document space itself. Further, none of these approaches allow interaction based on opinion alignment or divergence from those of the crowd<sup>1</sup> It is in the space of these unfulfilled needs that we situate our work on Echo.

## IMPLICATIONS FOR DESIGN

From on the above literature in reading, memory and HCI, we can distill a set of four design guidelines:

- G1 Contextual Cues:** Leverage contextual cues such that users can recall information easier [9, 30]
- G2 Planning Phase:** Help readers quickly find important facts with little cognitive effort [32] from a range of opinions from the perspective most applicable to a given user’s needs
- G3 Callouts:** Leverage visual callouts [16, 32] to help users find key information in the document text
- G4 Alignment:** allow users to find opinions and documents based on relation to crowd’s opinion (unlike existing solutions which focus on summary only)[33]

## SCOPE & MOTIVATION

<sup>1</sup>Rather, opinion sentiment itself are treated as the end deliverable (e.g. stating that the “beds” were rated positively in hotel reviews).

Echo is not a document or corpus summarization tool, nor is the underlying algorithm performing summarization. Rather, Echo is the first algorithm and a UI for opinion alignment. Echo places individual opinions in document context and crowd context, allowing users to quickly find those opinions. Unlike existing approaches that display documents side-by-side, or focus on summarizing a document or corpus at a birds-eye level (visually and algorithmically), Echo takes an orthogonal approach. We allow a user to read any document in their corpus as they normally would, with crucial opinions in *each* document brought to attention akin to a magazine’s metaphor. Given the broad set of potential applications of Echo, the system can consume and render any type of document (e.g. blogs, online news, forums, reports, court documents) or format (e.g. HTML, PDF, DOC).

In this paper, we will refer to these visual distortions that are used to highlight key opinions as **Callouts**. Traditionally, an editor would make a series of subjective decisions on what to call out, and how to bring each word or phrase to the reader’s attention. This is an expensive process, time consuming, and the decisions are entirely subjective. Echo automates this process by gathering the collective intelligence of the crowd (via a corpus of documents), distilling what “the crowd” is talking about, and noting what opinions the crowd have about the main topics. Echo then visually distorts the text of a document, bringing to the user’s attention the key opinions in *each* document. In this paper, we will refer to key opinions that are representative of the crowd as **Representative** opinions, and well as opinions that go against the grain as **Out-Of-Line** opinions. A weighting algorithm determines the degree to which each opinion is visually brought to the forefront or left in the periphery, based on how important each topic is, and how much an opinion is in-line or out-of-line with the crowd.

## ECHO

Echo is comprised of three key components: the **Echo Reader**, the UI that a user sees when they read a given document in a corpus; the **Echo Locator**, which is the UI that allows a user to browse the corpus of documents (and open them in the Echo Reader) based on both topic and opinion representativeness; and the **Echo System**, which is the back-end algorithm and system that performs the data mining analysis of the document corpus. The UI is HTML based, and is rendered in a web browser. In the following section, we detail the Echo Reader and the Echo Locator UI interactions as perceived by the user. The section following the UI discussion covers the specifics of *how* the underlying Echo System makes its calculations and models. Echo has been tested on political and financial reports, as well as the more mundane domain of movie reviews.

## DESIGN & INTERACTION

We designed the first-load form of a document in Echo to require no explicit user interaction while still providing a majority of the benefits of the Echo System’s analysis. Yet with simple and non-intrusive mouse-hovers, documents in Echo can become an interactive and exploratory experience.



Figure 2: Three Screenshots of a User Interacting with a Document in the Echo Reader  
From corpus of political articles about the August 11th 2012 Romney nomination of Paul Ryan.  
These are high resolution screenshots, and can be zoomed in when viewed digitally.

We illustrate the functionality and features of the Echo Reader and Echo Locator through a scenario that follows Fran, a political analyst. To illustrate the breadth of the features in the Echo Reader and Echo Locator, the following scenario is overly extensive to cover multiple approaches and uses of Echo. In practice, users can apply any subset of the following techniques, at any time, in any order or any type of data source.

*Fran is a political analyst for the WeSaySo TV News Network. Fran is working on a story about Republican presidential candidate Mitt Romney's selection of Paul Ryan as his running mate in the 2012 US election. To get a handle on what the news media is saying about this announcement, she loads a document corpus of political news articles related to the announcement into Echo. Fran first needs to get a general sense of, "what are people saying?"*

### Corpus Browsing with Echo Locator

Fran begins by loading her document corpus into Echo, and opening up the Echo Locator user interface (Figure 1). Fran really wants to see those documents that have the most representative opinions so she can get a high level view of the document space. The Echo Locator interface will allow Fran to browse her entire corpus of over 300 documents quickly and easily by mitigating information overload.

On page-load, Fran can see all documents in the corpus sorted alphabetically. For each document, its title, original URL (source of the content), and a preview of the body copy is shown. Along the right side of the user interface are the

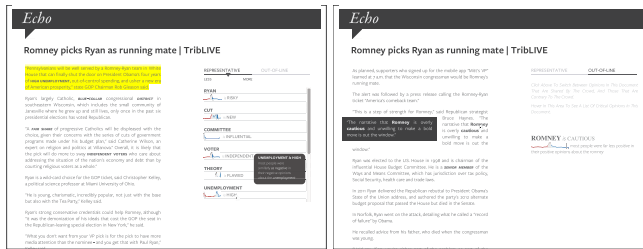
words "all documents," followed by a list of all the subjects of discussion sorted by how important the subject is within the corpus. This list is scrollable upon hover. Should she wish to filter the document space, Fran can click any subject (or "all documents") to show only documents that have opinions on the clicked subject. The currently selected filter (or "all documents") is in dark gray. Whichever subject (or "all documents") that is selected, three sort controls appear to its right. By clicking on any of these icons, the current list of documents is filtered alphabetically (default)  $\updownarrow$ , by how Representative they are on the given subject ( $\uparrow$ ) or how Out-Of-Line they are on the current subject ( $\nabla$ ). The opinion based sorts are in defending order, so that the most Representative/Out-Of-Line document is at the top<sup>2</sup>.

Because Fran wishes to gather the best "high level" view on this corpus, she leaves the filter as "all documents," and re-orders the list based on how Representative they are by clicking the  $\uparrow$  icon (G4). Skimming the list, Fran can then opens a documents at the top of the list that catch her eye.

### Page-load and Crucial Opinion Call-Outs

Upon page-load in Echo Reader, the document appears like any standard web-based article that has been laid out well by an Editor (Figure 2A). Some of the document's most Representative opinions are called out for Fran's attention (G3).

<sup>2</sup> The Echo Locator places a small icon to the right of each title to indicate for the given filter, if the document is Representative ( $\uparrow$ ) or Out-Of-Line ( $\nabla$ ) on the specific subject.



(a) Interactions with Opinion List (b) Out-Of-Line Opinions

Figure 3: Additional User Interactions with Echo Reader  
High resolution screenshots can be zoomed in when viewed digitally.

The degree of visual callout is proportional to how important the subject is, and how Representative the opinion is.

Thus, from the moment a page loads, Fran does not need to interact with the document and can still reap a large portion of Echo's benefits by scanning and reading the text as she normally would (G2). Immediately, Fran quickly scrolls through the document and quickly finds the most representative opinion in the document, which is brought to her attention in a black box (G3). Less noteworthy opinions are brought to her attention in a smaller degree visually, through the use of small caps, bold, italic, or even darker shades of gray based on the hierarchy of visual and typographic changes by Bringhurst[10]. The less the body copy is called out, the less Representative the opinion is. This allows Fran to quickly check the pulse of the community within the context of the document she is reading, for any particular opinion she encounters. By borrowing well known style metaphors from magazine layouts, the Echo Reader directly facilitates Fran's planning phase (G2), incidental and contextual memory (G1).

### Community Context Hover Events

After scanning the document, Fran is intrigued by the comment that, "Some will see [Paul] Ryan as politically risky..." which was called out in a black box. To find out exactly why this is opinion is so Representative, Fran moves her cursor on top of the phrase in the text. Upon hovering, a sparkline[42] and english sentence explanation situating this opinion in the broader context appears on the right side of the page (Figure 2B). The sparkline allows Fran to see the distribution of opinion sentiments on the subject of Ryan across all documents in the corpus (G4): the negative part of the distribution appear in red, the positive in blue, and neutral in gray:



The portion of the distribution which contains this particular opinion on Ryan (which is just slightly more positive than neutral) is filled in with dark blue. This allows Fran to visually place opinions within a given document in the broader context of the corpus/crowd.

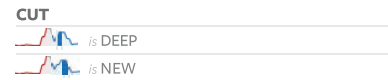
### Opinion Sliders

Upon load, the Echo Reader calls a subset of all Representative opinions to the reader's attention. Because Fran wishes to get the broadest overview of the crowd's opinions (G4),

she wants to see more of the Representative opinions in this document. She moves her cursor over the right hand side of the screen, and a slider under the word Representative fades in (Figure 2C). Fran can use this slider to adjust (more or less) the amount of Representative opinion callouts in the text. Increasing this slider causes new callouts to emerge which, though still Representative, are on somewhat less important subjects, or situated slightly further away from the mean community opinion on the topic.

### Opinion Overview List

After spot-exploring a handful of opinion callouts in the document, Fran wants to see an overview of *this* document's Representative opinions. She once again moves her cursor over the right hand side of the screen. During a slightly longer hover event (as compared to the shorter hover event that produced the opinion sliders), a scrollable list of Representative opinions then appears (Figure 2C). The list consists of a series of subjects, with the modifying opinions appearing below the subject (G2). If Fran now manipulates the slider under the word Representative, not only would more/less opinions callouts appear in the text, but more/less opinions would populate this list. When the document has more than one opinion on a subject, they both appear below the subject word:



Each opinion appears in italic, and has a opinion distribution sparkline next to it. Fran can now find noteworthy opinions she may have missed. Because each opinion in the list corresponds to a specific word/sentence in the text, when Fran hovers over an opinion in the list, the Echo Reader auto scrolls to where that opinion occurred in the text, highlights the sentence with the opinion (in yellow like a highlighter) and provides an brief english sentence explanation situating this opinion in the broader context<sup>3</sup> (Figure 3a). Highlighting or pointing to text from a list has been empirically shown to be a powerful interaction to improve document browsing[16]. Thus with this overview and interaction, Fran can quickly survey the Representative opinions in this document, quickly gauge their sentiment, see the document context, and place the opinions within the spectrum of the crowd's.

### Out-Of-Line Opinions

Now that Fran has a good grasp on this document's Representative opinions, she wants to see where this document differs from the crowd (G4). Next to the word Representative on the right hand side of the Echo Reader, is a grayed-out word Out-Of-Line. By clicking on Out-Of-Line, the Echo Reader switches from bringing Representative opinions to the reader's attention, to calling out Out-Of-Line opinions. Fran now can visually see the contrasting opinions in the document (Figure 3b) through visual callouts. Just as with the Representative opinions, she can use community context hover events in the body copy (Figure 3b), explore with a summary list, and move the Out-Of-Line slider to adjust the amount of callout. By keeping Representative and Out-Of-Line opinions

<sup>3</sup>This is the same text that appear in the community context hover events, but without the sparkline.

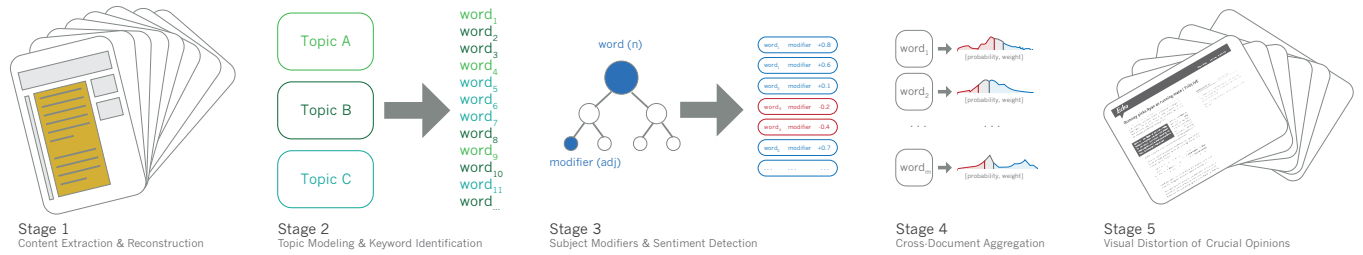


Figure 4: Echo Five-Stage Algorithm Overview

separate through two modes, the Echo Reader avoids confusion as to what type of opinion callout (or what element in the list) is currently being shown.

### Continued Exploration

When Fran finishes reading her document, she wishes to continue exploring the corpus by returning to the Echo Locator (Figure 1). After reading the first document, Fran has decided she needs more information about how people talk about the office of the president. She quickly filters her list of documents by clicking on the word “president,” and gets an alphabetically sorted list of documents that have opinions of the topic “president.” To signify the filter is active, “all documents” becomes light gray, and “president” darkens. Fran can now re-sort her list, find documents that are highly Out-Of-Line or Representative of the crowd’s opinion on the president, and begin reading those documents in the Echo Reader.

### IMPLEMENTATION

The Echo interface is built in HTML, CSS and JavaScript. Document corpora undergo a Java-based five-stage algorithm, the (Echo System), in order to generate the Echo UI (Figure 3): 1) Content Extraction & Reconstruction; 2) Topic Modeling & Keyword Identification; 3) Subject Modifiers & Sentiment Detection; 4) Cross-Document Aggregation; and, 5) Visual Distortion of Crucial Opinions. Figure 4 presents a overview of the five-stage process.

It should be noted that the Echo System assumes that a corpus of documents (PDFs or links to webpages) on the same subject have been collected.<sup>4</sup>

#### Stage 1. Content Extraction & Reconstruction

*Finding The Text in Each Document, and Building Indexable Content*

First, we must crawl and clean every document in a corpus to identify the body copy<sup>5</sup>. Body copy is needed to perform the ML analysis and generate the re-rendered document in Echo.

PDF body copy extraction[3] is relatively easy, as most PDF documents do not contain copious amounts of non-body copy

<sup>4</sup>Grouping and clustering articles by subject matters is an active area of research for those in the IR and ML communities, and is outside the scope of this work. The Echo System implementation does have a crawler for recent Google News stories and a PDF uploader so as to facilitate the core contributions of Echo itself.

<sup>5</sup>Body copy is a term from the layout and design community used to describe the main text of a document, as compared to logos, images, title, advertisements, comments etc.

material. Unlike PDFs, web content is not as “clean.” The raw HTML for any given webpage has navigation elements, ads, comments, or other non body copy content. Further complicating body copy extraction is that every webpage uses different CSS markup for its content. While templates, hand-coded or auto-generated, are often used to identify body copy [6], the number of templates needed would grow quite large. Consider the diversity of the set of all sources from a Google News query. In response, we utilized the CETR algorithm [43] which extracts body-copy from webpages using HTML tag ratios and multidimensional clustering. Thus, we can extract body-copy from any source, whether it has seen a given layout before or not.

Once the body copy of a given document is extracted, we reconstruct each document fully marking-up the content, with each sentence and word<sup>6</sup> within a distinct HTML `<span>`. Thus, any element of text can be easily found using JavaScript lookups, so that their visual weight can be modified.

#### Stage 2. Topic Modeling & Keyword Identification

*Finding What the Corpus is Talking About*

Next we attempt to identify *what* the articles within a given corpus are talking about. We limit our word selection to Nouns, as they are generally considered the “subject” of a sentence, by applying POS tagging[2] as a filter<sup>7</sup>. With filtered documents, we apply Latent Dirichlet Allocation (LDA) topic modeling[31] to extract topics and associated keywords across all documents.

LDA topic modeling considers each document as being the outcome of a mix of topics which each generate some portion of the document’s content. In order explain our use of LDA, consider  $T$  to be the set of all topics ( $t_0 \dots t_k$ ),  $D$  to be the set of all documents in our corpus ( $d_0 \dots d_l$ ), and  $W$  to be the set of all unique words ( $w_0 \dots w_m$ ) across all documents. Every document ( $d_i$ ) consists of a subset of words from  $W$ . Every topic ( $t_j$ ) consists of a subset of words from  $W$ , each with a probability  $P(w_q|t_j)$  that if a word is randomly chosen from  $t_j$ , it is that particular word ( $w_q$ ). In this way,  $t_j$  is a probability distribution over  $W$ :

$$\sum_{q=0}^m P(w_q|t_j) = 1 \quad (1)$$

<sup>6</sup>Sentence and Word detection is done through use parser from [41].

<sup>7</sup>While an alternative to filtering by Noun could be Named Entity Recognition[17], entity disambiguation is an open challenge, and we opted to utilize a more conservative approach.

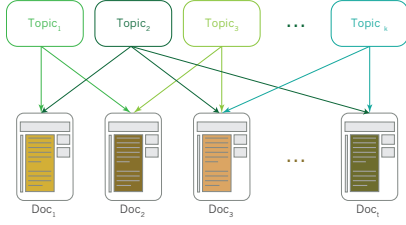


Figure 5: Latent Dirichlet Allocation (LDA)

and a document ( $d_i$ ) consists of a probability distribution over all topics  $T$ :

$$\sum_{j=0}^k P(t_j|d_i) = 1 \quad (2)$$

In layman's terms, every word in a document is contributed by one of the topics.

Our goal is to rank order all words in  $W$  based on how "important" they are across all topics and documents. LDA gives us the rank ( $P(w_q|t_j)$ ) of each word ( $w_q$ ) given a particular topic ( $t_j$ ). However, these word probabilities cannot be compared across topics as they are conditional. We therefore marginalize out the topics for each word:

$$P(w_q) = \sum_{j=0}^k P(w_q|t_j)P(t_j) \quad (3)$$

This calculates for each word a rank that is topic independent, and therefore valid across our document corpus. This ranking provides the most important subjects of discussion in the corpus.

### Stage 3. Subject Modifiers & Sentiment Detection

*Finding the Opinions on Key Subjects*

Next we attempt to identify the opinions on each of the important subjects of discussion. We apply a statistical parser[15] to each sentence in each document, generating a full parse tree, a dependence parse tree, and the part of speech for each word. These sentence parse trees are akin to sentence diagramming done in grade school. From these parse trees we can programmatically uncover which modifiers (e.g. adjectives) were applied to any sentence subject.

For each modifier uncovered, we calculate a sentiment score by looking up its value in Senti-WordNet[5]. Senti-WordNet is a hand-compiled list covering a large selection of common word modifiers. Each word is evaluated on two 0.0-1.0 scales, Positive and Negative. We maintain both positive and negative values because if combined on a single scale, on average, most results would trend towards neutral. Modifier lookup does not have word sense disambiguation (context). However, it does allow us to explicitly associate a modifier with a subject in a sentence. Thus we are able to calculate the opinion on a subject within a sentence, rather than the holistic sentiment of the entire sentence.

During the sentiment analysis stage, we also identify negations in the parse tree, so we can invert the opinion of the modified subject. For example, in the phrase "the stock is

not a good buy," the subject "stock" is modified by "good." However there is a negation - "not" - requiring us to swap the opinion's positive and negative scores.

Thus, for each of the important subjects in Stage 2 ( $w_q$ ), we uncover the set of opinions  $O(w_q)=[o_1(w_q) \dots o_n(w_q)]$  within each sentence where the subject occurs.

### Stage 4. Cross-Document Aggregation

*Finding the Opinion Trends Across the Corpus*

Next, we aggregate all opinions on each subject over all documents. First we find all instances of each subject by computing the base form of each word[41] and americanizing all spelling. Thus, for a given corpus, we have a weighted list of subjects (from stage 2) and for each subject a distribution of opinions (from stage 3).

From these per-subject distributions ( $O(w_q)$ ), we can calculate a mean positive  $\mu_{pos}(w_q)$  and negative sentiment  $\mu_{neg}(w_q)$ , as well as the standard deviation  $\sigma_{pos}(w_q)$  and  $\sigma_{neg}(w_q)$ . We refer to this mean as the crowd's opinion on a given subject. For each subject-opinion instance in each document  $o_v(w_q)$ , we calculate that instance's distance:

$$F_{pos}(o_v(w_q)) = \frac{|o_{pos,v}(w_q) - \mu_{pos}(w_q)|}{\sigma_{pos}(w_q)}$$

$$F_{neg}(o_v(w_q)) = \frac{|o_{neg,v}(w_q) - \mu_{neg}(w_q)|}{\sigma_{neg}(w_q)} \quad (4)$$

$$F(o_v(w_q)) = \min(F_{pos}(o_v(w_q)), F_{neg}(o_v(w_q)))$$

We therefore consider a specific subject-opinion to be "representative" of the crowd if the opinion's distance is less than or equal to one (within one SD). If the opinion distance is greater than one, we consider the subject-opinion to be "out of line" from the crowd.

### Stage 5: Visual Distortion of Crucial Opinions

*Showing the User Crucial Opinions*

Finally, for each document, we must determine what subject-opinions should be called out, and how much to visually bring them to the user's attention. From Stages 1-4, every subject-opinion  $o_v(w_q)$  has an importance value  $P(w_q)$  and an opinion distance  $F(o_v(w_q))$ . To find the most crucial "representative" and "out of line" subject-opinions, we use the following ranking equations:

$$F_L(o_v(w_q)) = \max(F_{pos}(o_v(w_q)), F_{neg}(o_v(w_q)))$$

$$R_{out}(o_v(w_q)) \propto P(w_q) * F_L(o_v(w_q))$$

$$R_{rep}(o_v(w_q)) \propto \frac{P(w_q)}{F_L(o_v(w_q))} \quad (5)$$

Based on the rankings in Equation 5, within each document, we place each subject-opinion instance in a ranked-tier (separately for both representative and out of line subject-opinions): 1) Normalize the ranking values of all subject-opinions in the document; 2) Sort the normalized ranking values in descending order; 3) Based on cumulative sum, distribute the subject-opinion instances into 10 tiers, such that each tier must have at least as many words as the tier index (at least one word in tier one, two words in tier two, etc).

While the user specifies *which* tier of subject-opinions is called out via the UI sliders, the rendering of said call out is determined by the relative importance of typeface/weights combinations [10]. Robert Bringhurst outlines the hierarchy of visual and typographic changes that can be made to text to call salient information to the user's attention at various degrees. For tiers 2-10, we adhere closely to [10], visually distorting the subject and modifiers in the crucial opinions. Tier 1 (being the most crucial "representative" or "out of line" opinion) is additionally called out by the entire sentence containing the subject and modifier, being replicated in a black box with white type. This extreme, though sparse, callout parallels the large callouts commonly found in magazine articles. Any element of text in the body copy that needs to be visually distorted or undistorted, can be easily found using the tags created in Stage 1, during the document recreation.

This same ranking algorithm (5) is used for the Echo Locator UI generation. To sort documents by how representative or out-of-line they are, we simply sum all rankings on a given subject within a document to get a document level rank. For cross subject rankings, we sum across all subject-opinions in a document. By using sum, rather than mean, we give more weight to documents that have many representative or out-of-line opinions, rather than just one strongly opinion.

## INITIAL USER REACTIONS

In order to explore the benefits of and the reaction to Echo, we conducted a preliminary pilot study to evaluate the reaction to and value of this approach. We gauged the user reaction, and examined the initial interaction with the Echo UI related to aspects of information overload, and discovery so as to support the main contribution of this work, our system and algorithm.

To ecologically ground our pilot study, we recruited 12 knowledge workers (eight female and three male, ages 33 to 56) from a large US corporation (3 from the legal department, 4 administrative assistants and 5 from marketing and outreach). All of our participants reported using large collections of documents daily or a few-times a week, with self reported average corpus size of 1,262 documents. None of the participants had prior knowledge of, or experience with, Echo.

To provide further grounding, our study task followed O'Hara [32] and Winograd[44] who suggest that text summarization requires a deep understanding of text, relates to the skills required to comprehend a document space, and has a "strong connection with many of the kinds of reading tasks... knowledge workers [carry] out" [32]. Further, to demonstrate the breadth of the Echo, we demonstrated its use with 3 data sets from 3 different domains during this experiment.

## Dependent and Independent Measures

This experiment followed a within subject design that manipulated one independent variable, the UI for reading documents, which consisted of two levels 1) The Echo Reader UI (referred to as the *Echo Condition*) and 2) A plain text UI with the same general layout and typeface as the Echo

Reader without any call-outs or interactive components (referred to as the *Traditional Condition*)<sup>8</sup>. The primary dependent variable measure was the time taken to read a small collection of (5) documents with each UI. Given the small size of our dataset, we utilized a Two-Sample Wilcoxon Rank-Sum (Mann-Whitney) test, a more conservative metric than the Student's T-Test as it makes no assumptions about the data distribution. In addition, we also recorded the user's screen and administered questionnaires to understand users' interactions, satisfaction and preferences.

## Set-up & Methods

Participants were first presented with the Echo Locator UI and the Google News UI as two methods for finding a document in a corpus. They were then asked to complete a brief questionnaire focused on how they navigate document collections, and how each of the just demonstrated UIs could benefit their own work. The document collection was a corpus of financial analyst articles about the August 8th HP announcement that the company was writing down the value of its enterprise-services business by \$8 billion, and changing people in upper management. While gaining feedback on Echo Locator was valuable, the primary purpose of this phase was to situate the task of reading documents in a large corpus.

Next, participants were shown the Echo Reader UI using the same dataset. Following the opportunity to explore the UI or ask questions, participants were asked to read two sets of five documents,<sup>9</sup> each on a different topic. One document set was a collection of movie reviews about the 2009 film *Star Trek*, and the other was a collection of political news articles about the pick of Paul Ryan as Mitt Romney's running mate in the 2012 US election.

For each participant, one set of five documents was in the *Echo Condition* and the other was in the *Traditional Condition*. The assignment of topic and presentation order was counterbalanced across participants to mitigate ordering and topic effects. Thus, there were four possible combinations of topic and order, and each group had three participants.

Based on the readings of the two sets of documents, participants were then asked to write a 2-3 sentence summary of the opinions within each set. After completing the summaries, participants were asked to complete a questionnaire about their experience with both interfaces. The experiment was run using a full-screen Chrome Browser on a 24" monitor. On average, the study took 45 minutes per participant.

## Quantitative Results

For each participant, the total time to read each set of five documents was calculated in order to control for potential differences between each individual document. Document reading with the Echo Reader had a mean time of 466.08 seconds ( $\sigma=226.87$ ) while reading with the Traditional UI had a mean time of 289.75 seconds ( $\sigma=116.92$ ). Statistical

<sup>8</sup>The articles in both conditions used the same layout, typography, and sizing. Thus, users were comparing interactivity and highlights not UI design.

<sup>9</sup>Each set of 5 documents were drawn from a collection of over 100 documents fed into Echo.

|                                                             | Mean | (SD)   |
|-------------------------------------------------------------|------|--------|
| In Terms of Ease of Use, Which was Better? †                | 1.25 | (0.62) |
| In Terms of Usefulness, Which was Better? †                 | 1.08 | (0.29) |
| Which Would Give the Most Benefit for Your Own Work? †      | 1.17 | (0.58) |
| Which Was Best for Making Sense of a Document Collection? † | 1.25 | (0.62) |
| How Helpful were Callouts in Finding Key Information? ‡     | 4.42 | (1.16) |
| How Well Did the Callouts Pick Important Information? ‡     | 4.17 | (1.11) |

Table 1: Reader UI Comparisons

Rated From: † 1 = Echo, 3 = neutral, 5 = Traditional UI

Rated From: ‡ 1 = "Not at All," 5 = "Extremely"

significant differences were found (Wilcoxon Rank-Sum) between the two UIs ( $p=0.02$ ), though there appeared to be no order effects ( $p=0.27$ ) nor topic effects ( $p=0.21$ ).<sup>10</sup> Participants were asked to rate their satisfaction with the two UIs and the value of callouts (Table 1). In addition, when examining the videos, 10 of the users employed the Community Context Hover Events, 10 of the users employed the Opinion Overview List, and 9 users clicked to view the Out-Of-Line opinions. When using the Echo Reader, two of our participants' entire method of document reading was traversing the opinions using the Opinion Overview List, and letting the document auto-scroll. While it did appear that many users performed an initial scroll through of the document in Echo Reader looking for key callouts, we are hesitant to include a quantitative count, since we are unable to categorically state that this was the users' intent.

### Qualitative Results & Discussion

Participants were overwhelmingly positive in their quantitative and qualitative feedback regarding Echo:

*Wow, that was really really great... You know when you're in school, and you buy a used book thats been highlighted. You have to hope that its an A student [that highlighted it], not a D student.... its almost like you are cheating. - P2*

When participants were asked why they thought they took longer with the Echo Reader, responses tended to focus on not missing information, ease of finding information, and taking advantage of Echo's intelligence;

*[I]t was easier to note the essential information and I spent more time lingering on it and thinking about it than when I was trying to wade through lots of homogeneous material - P11*

*I wanted to make sure that I got the key point of the callouts and it led me to want to see the opinions associated with the callouts. - P5*

While our data showed that reading with Echo took longer, half of the participants did not notice this difference and were surprised that Echo took longer. P4 stated that she thought the Echo Reader was "*faster and [gave a] more accurate summary of keywords than skimming*." This suggests that the reading experience with Echo was not only more powerful, but users did not feel that it was tedious or were wasting their time on the unnecessary information in the documents. This would need to be confirmed in a longer-term study.

<sup>10</sup>Lack of significance does not mean lack of statistical difference.

In addition, we asked participants to come up with applications of the Echo Reader and Echo Locator interaction models. Most participants suggested applications in the legal domain (litigation, and IP), medical, finance, education, and news aggregators. When asked if there were any benefits to the traditional UI, participants generally focused on being "forced to read the whole thing," and drawing their own conclusions on document text without bias from the crowd.

It is important to note that while the individual algorithmic components in Echo make use of straightforward approaches from the ML and NLP domains, participants explicitly stated that the callouts were helpful and appeared to pick out key information well. We strongly believe that this is due to the power of aggregating opinions based on the wisdom of the crowd, combining both subject importance in a topic model with sentiment detection and delta from the mean.

### LIMITATIONS & FUTURE WORK

While the Echo System employed standard Machine Learning techniques, there are more powerful topic and sentiment tools that exist on the "bleeding edge." As we continue to develop Echo, we'll seek to improve the Echo System algorithm with more advanced techniques for sentiment analysis (taking sentence context into account), keyword extraction (with n-gram phrases and Named Entity Recognition) and topic modeling (combining supervised and unsupervised techniques). However, participants did report that the callouts were highly accurate, even with our current approaches.

Another direction of future work would be the inclusion of document traversal in the Echo Reader, such that users could select a called out keyword, and request to see a list of documents that further support, or go against that opinion. In addition, based on feedback from users in our Pilot Study, we wish to include in keyword search in Echo Locator, allowing users to filter based on topics as well as keywords. These additional features would expand both the usability of Echo, and the ability of users to traverse their datasets.

Given the small size of this pilot study, our initial investigation is not intended as a validation of Echo. rather to gauge initial Echo's potential utility. We would like to perform a long-term ecological valid study of Echo, integrating it within a corporate or legal environment allowing us to better understand how impact on job performance.

### CONCLUSION

In the information age, it is often hard to sort out the relevant information that one needs to pay attention to. With so many sources of documents at our digital fingertips, we enjoy enormous diversity of sources but at the cost of suffering from a poverty of attention. In this paper, we provide a novel automated system and interface to help users navigate and find relevant content when foraging for information. This system, Echo, leverages the "wisdom of the crowd," to create elegant magazine-style interfaces to documents. Through natural layout and typographic techniques, key information that can be displayed in any platform is brought to the attention of users. This is an important consideration, given that the problem with attention is often exacerbated by the small visual real

estate of small devices such as smart phones. Further, positive results from a pilot study offer support for the benefits offered by Echo's design. Thus, the foremost contribution of this work is a interaction model and algorithmic approach that allows users to place each document and opinion within the larger echo of voices.

## REFERENCES

- Adar, E., and Huberman, B. A. The economics of surfing. *Quarterly Journal of Electronic Commerce* (2000).
- Aphache.org. Apache OpenNLP.
- Aphache.org. Apache PDFBox.
- Archak, N., Ghose, A., and Ipeirotis, P. G. Show me the money!: deriving the pricing power of product features by mining consumer reviews. In *KDD '07*, ACM Request Permissions (Aug. 2007).
- Baccianella, S., Esuli, A., and Sebastiani, F. Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In *Proceedings of the 7th Conference on Language Resources and Evaluation* (2010).
- Bar-Yossef, Z., and Rajagopalan, S. Template detection via data mining and its applications. In *the eleventh international conference*, ACM Press (New York, New York, USA, 2002), 580.
- Baudisch, P., Lee, B., and Hanna, L. Fishnet, a fisheye web browser with search term popouts: a comparative evaluation with overview and linear view. In *AVI '04: Proceedings of the working conference on Advanced visual interfaces*, ACM (May 2004).
- Billsus, D., Hilbert, D. M., and Maynes-Aminzade, D. Improving proactive information systems. In *the 10th international conference*, ACM Press (New York, New York, USA, 2005), 159–166.
- Bjork, R. A., and Richardson-Klavehn, A. *On the puzzling relationship between environmental context and human memory*. Lawrence Erlbaum Associates, Inc, 1989.
- Bringhurst, R. *The Elements of Typographic Style*, 3rd ed. Hartley and Marks Publishers, Oct. 2004.
- Brush, A. J. B., Barger, D., Gupta, A., and Cadiz, J. J. Robust Annotation Positioning in Digital Documents. In *the SIGCHI conference*, ACM Press (New York, New York, USA, 2001), 285–292.
- Carenini, G., Ng, R. T., and Pauls, A. Interactive Multimedia Summaries of Evaluative Text. In *the 11th International Conference*, ACM Press (New York, New York, USA, 2006), 124.
- Chakravarthy, M., and Ciravegna, P. Cross-media document annotation and enrichment.
- Davies, S., Allen, S., Raphaelson, J., Meng, E., Engleman, J., King, R., and Lewis, C. Popcorn: the personal knowledge base. In *the 6th ACM conference*, ACM Press (New York, New York, USA, 2006), 150–159.
- De Marneffe, M. C., and MacCartney, B. Generating typed dependency parses from phrase structure parses. In *Proceedings of LREC '06* (2006).
- Dillon, A., Richardson, J., and McKnight, C. Navigation in hypertext: a critical review of the concept. *Navigation* (1990).
- Finkel, J. R., Grenager, T., and Manning, C. Incorporating non-local information into information extraction systems by gibbs sampling. *ACL'05* (2005).
- Ganesan, K. A., Sundaresan, N., and Deo, H. Mining tag clouds and emoticons behind community feedback. In *WWW '08*, ACM (Apr. 2008).
- Halvey, M. J., and Keane, M. T. An Assessment of Tag Presentation Techniques. In *the 16th international conference*, ACM Press (New York, New York, USA, 2007), 1313.
- Hearst, M. A. User interfaces and visualization. *Modern information retrieval* (1999).
- Hearst, M. A., and Rosner, D. Tag clouds: Data analysis tool or social signaller? ... *on System Sciences* (2008).
- Huberman, B. A., Piroli, P., Pitkow, J., and Lukose, R. Strong Regularities in World Wide Web Surfing. *Science* 280, 5360 (Apr. 1998), 95–97.
- Huberman, B. A., and Wu, F. The economics of attention: Maximizing user value in information-rich environments. *Advances in Complex Systems* (2008).
- Keppel, G., and Underwood, B. J. Proactive inhibition in short-term retention of single items. *Journal of Verbal Learning and Verbal Behavior* (1962).
- Kriplean, T., Morgan, J., Freelon, D., Borning, A., and Bennett, L. Supporting reflective public thought with considerit. In *CSCW '12*, ACM Request Permissions (Feb. 2012).
- Kuo, B. Y.-L., Hentrich, T., Good, B. M., and Wilkinson, M. D. Tag clouds for summarizing web search results. In *the 16th international conference*, ACM (May 2007).
- Lee, S. E., Chun, T., and Han, S. S. Using Qtag to Extract Dominant Public Opinion in Very Large-Scale Conversation. In *2009 International Conference on Computational Science and Engineering*, IEEE (2009), 753–758.
- Lee, S. E., Son, D. K., and Han, S. S. Qtag: tagging as a means of rating, opinion-expressing, sharing and visualizing. In *SIGDOC '07*, ACM Request Permissions (Oct. 2007).
- Liu, B., Hu, M., and Cheng, J. Opinion observer: analyzing and comparing opinions on the Web. In *the 14th international conference*, ACM Press (New York, New York, USA, 2005), 342.
- Marshall, C. Saving and using encountered information: implications for electronic periodicals. In *CHI '05* (2005).
- McCallum, A. K. Mallet: A machine learning for language toolkit.
- O'Hara, K., and Sellen, A. A Comparison of Reading Paper and On-Line Documents. In *the SIGCHI conference*, ACM Press (New York, New York, USA, 1997), 335–342.
- Ottbacher, J., Radev, D., and Kareem, O. News to go: hierarchical text summarization for mobile devices. In *SIGIR '06* (2006).
- Park, S., Kang, S., Chung, S., and Song, J. NewsCube: delivering multiple aspects of news to mitigate media bias. In *CHI '09: Proceedings of the 27th international conference on Human factors in computing systems*, ACM Request Permissions (Apr. 2009).
- Park, S., Ko, M., Kim, J., Liu, Y., and Song, J. The Politics of Comments: Predicting Political Orientation of News Stories with Commenters' Sentiment Patterns. In *the ACM 2011 conference*, ACM Press (New York, New York, USA, 2011), 113.
- Rivadeneira, A. W., Gruen, D. M., Muller, M. J., and Millen, D. R. Getting our head in the clouds: toward evaluation studies of tagclouds. In *CHI '07: Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM Request Permissions (Apr. 2007).
- Sereno, B., Shum, S. B., and Motta, E. ClaimSpotter: an environment to support sensemaking with knowledge triples. In *the 10th international conference*, ACM Request Permissions (Jan. 2005).
- Simon, H. A. Designing organizations for an information-rich world. *Computers, Communication, and the Public Interest*, The Johns Hopkins Press (1969).
- Suh, B., Woodruff, A., Rosenholtz, R., and Glass, A. Popout prism: adding perceptual principles to overview+detail document interfaces. In *CHI '02: Proceedings of the SIGCHI conference on Human factors in computing systems: Changing our world, changing ourselves*, ACM Request Permissions (Apr. 2002).
- Sun, J.-T., Wang, X., Shen, D., Zeng, H.-J., and Chen, Z. CWS: a comparative web search system. In *WWW '06*, ACM (May 2006).
- The Stanford Natural Language Processing Group. Stanford CoreNLP.
- Tufte, E. R. *Beautiful evidence*, 1st edition ed. Graphics Pr, 2006.
- Weninger, T., Hsu, W. H., and Han, J. CETR: content extraction via tag ratios. *WWW 2010* (2010).
- Winograd, P. Strategic Difficulties in Summarizing Texts. *Reading Research Quarterly* 19 (1984), 404–425.
- Wu, F., and Huberman, B. A. Novelty and Collective Attention. *PNAS* 104, 45 (Nov. 2007), 17599–17601.
- Yatani, K., Novati, M., Trusty, A., and Truong, K. N. Review Spotlight: A User Interface for Summarizing User-generated Reviews Using Adjective-Noun Word Pairs. In *the 2011 annual conference*, ACM Press (New York, New York, USA, 2011), 1541.