

Odin: Contextual Document Opinions On The Go

Joshua Hailpern

HP Labs

1501 Pagemill Road

Palo Alto, CA 94304

joshua.hailpern@hp.com

Bernardo A. Huberman

HP Labs

1501 Pagemill Road

Palo Alto, CA 94304

bernardo.huberman@hp.com

ABSTRACT

Information overload is a systemic problem for knowledge workers in enterprise. For a long time, information was scarce and therefore valuable. While, the explosion of digital information has made information plentiful, time to read and process that content is now scarce. This problem is only exacerbated by our increased mobility, and the expectation to be “on top” of the continuous barrage of documents while on the go. Knowledge workers in enterprise need solutions that are designed with quick methods for finding *what* to read in a large collection of documents (e.g. financial reports, legal documents, news), and ways of *presenting* it within small visual real estate. Unlike reviews, document collections are long, more varied, and context is extremely important. In response, we present Odin, a mobile web-based window onto a user’s document corpus. Rather than performing corpus summarization, Odin users can quickly find opinions and documents that are Aligned or Divergent from the corpus’ consensus, or those that are the most Relevant given the overall corpus’ of opinions. Odin presents this information through a simple and intuitive mobile interface. To the authors’ knowledge, this is the first UI/system (and support algorithm) to allow mobile users to place documents and their opinions in context through alignment rather than raw word count or sentiment. Positive results from two evaluations are also presented.

Author Keywords

Consensus; Mobile; Interface; Interaction; Economics of Attention; Alignment; Divergence; Opinions

ACM Classification Keywords

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

INTRODUCTION

While the web enables information access on a scale that would have been inconceivable a few years ago, it also creates the problem of information overload; access to a plethora of relevant content in a timely manner, without distractions from information which is marginal in value or relevance. The problem becomes even more acute when we acknowledge

that knowledge workers (e.g. lawyers, financial investors, economists, CEOs) are increasingly accessing their large corpus on their mobile devices. Mobile enterprise workers must therefore be able to access large collections of documents on mobile devices’ with small visual real estate and limited time/attention to process said content.

Consider: The executive that has a matter of minutes before a meeting, may have time to read only *one* document, and maybe not even its full length; The lawyer waiting fifteen or twenty minutes for his partners to arrive, can perhaps read two or three long documents; Meanwhile the financial analyst making an hour long train commute to work, may have more leisure to fully browse a corpus, though her time and UI must still be optimized as much as possible. Though each of these temporal and interaction constraints are different, the underlying goals of the user is constant: on a mobile device, quickly traverse a corpus of documents, find the best number documents that meet a time constraint, and naturally find key points in each document based upon a broader context.

This challenge is noticeably different from the solutions proposed by the existing literature. Real world knowledge workers must mitigate corpuses of documents that contain, on average, 1,262 documents [18]. Unlike reviews (e.g. product, restaurant, etc), documents are long, highly varied, many not be as focused, and context of a given opinion is extremely important. Further, mobile devices have extremely limited screen real-estate (which is not ideal for large document consumption) and mobile users have extremely limited time.

In response, we created a novel interaction model and system called **Odin**. By leveraging and exploiting our understanding of the economics of attention [38, 23, 1], we created the first system to allow mobile device users (tested within but applications beyond enterprise) to browse and find the most salient documents in a large corpus, by placing those documents and options in context of the full corpus using opinion alignment (rather than summarization, word count, or raw sentiment scores). Thus, users can find opinions and documents based on their relationship to the corpus’ consensus on a given topic. In this way, Odin is like an implicit “wisdom of the crowd.” The Odin UI and workflow is tailored for mobile devices and users that have limited time to consume documents. While there has been a large body of research on opinion mining, only one has investigated review mining on a smartphone (short Yelp reviews [22]) and none has examined document opinion mining on a mobile device nor using alignment of opinions (documents or reviews) to create mobile interfaces.

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. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

CHI 2014, April 26–May 1, 2014, Toronto, Ontario, Canada.
Copyright © 2014 ACM 978-1-4503-2473-1/14/04..\$15.00.

The foremost contribution of this work is a system and UI technique that allows the user to quickly find the document(s) needed given time and screen constraints, focus on the most important opinions in any document, while still being able to see how each article and opinion fits in its context (via a novel algorithm). Odin itself employs three key designs to provide this contribution and meet users' varying needs. First, to help users find a document that best meets their needs, Odin ranks each document on three different scales of importance: how **Relevant**, **Aligned** and **Divergent** it is from other documents in the corpus. Second, to help mitigate long document length, small screen sizes and limited time to consume each document, Odin provides an automatically generated **Executive Summary** of each document, highlighting key sentences and opinions for easy document digestion. Third, to address corpus size and complexity, the Odin UI has quick one-tap interactions to allow a user to find the most Relevant, Aligned or Divergent document without browsing, allowing users to get in-and-out of a corpus as quickly as possible.

In the remainder of this paper, we discuss how Odin builds on the existing literature and is situated within the broader set of solutions. We then present Odin's features and implementation, followed by positive results from two evaluations we conducted, and conclude with future and ongoing work.

RELATED WORK

Issues pertaining to information overload are not new, and were first codified in Simon's description of the relationship between attention and information [38]. More recently, information in the digital age has been shown to be a function of user access [1] and presentation [23]. Thus, we can look at these findings on economies of attention as a strong motivating and well grounded justification for new interfaces, especially those that focus on ease of information access to large document corpuses and presentation of content and opinions within a given document. Further, the need for document corpus management has been well established within the enterprise context [18], with corpuses averaging over 12,00 documents in size. This further motivates the need for solutions within a large, real-world context.

Likewise, there has been extensive work in automatic document summarization [3, 29, 40]. However, it is important to note that Odin is not a document summarization system or algorithm. Odin focuses on user interaction design, and systems to support the navigation of *key opinions* within a corpus on mobile devices by finding and placing opinions and documents in the context of a corpus.

Reading Documents

Research on document consumption/reading has identified four key activities: reading, annotating, collaborating, and authoring[9]. While all four are important, Odin focuses on the first activity *reading*, as it pertains to exploring the overwhelming collection of documents in a corpus on a mobile device. During this first activity, the reader enters a "planning phase" [31] when moving within and between documents. This is a critical step, in which, the readers focus on finding important facts so as to connect key "bits" of information.

And as documents' length and complexity increase, difficulty increase as well (e.g. consuming a restaurant review is easier than consuming a 20 page financial report).

Yet on mobile devices, this initial "planning phase" is made harder by the limited screen size. Cui suggests that because mobile devices do not allow users to see multiple documents at the same time (or long documents where less is visible at one time), people must keep all salient information in working memory [12], which is quite finite. We therefore must conclude that there is a glaring need for solutions that directly support the planning phase, as well as minimizing cognitive overload in finding the "right" document and opinions because a user's working memory is in active use.

Technological solutions are well situated to help users wade through a large document, and find key sentences on which to focus. AKTiveMedia [11], Fishnet [6], and others [36, 39, 7]) use vibrant color highlighting within a document to draw users' attention to keywords based on queries issued. These techniques have proven successful in a desktop metaphor, and we build upon their success by using color to highlight sentences in a long document. In a similar vein, Ahmed's recent work on accessible skimming [2] attempts to merge document summarization and screen-readers to help visually impaired users find key points within a document. Given this success for interactive reading tools, we believe that there is much evidence for this need, and room for improvement.

Context, Visualization, and Opinions

Beyond supporting the process of reading itself, some research has examined document "context." Souneil Park [34] explored 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 does so by showing contrasting documents physically side-by-side which, according to existing literature, does not port to a mobile context [12, 31]. Thus this approach lacks scalability and portability, which Odin directly addresses.

At a corpus level, tools can help users organize a document space. Popcorn [13] 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, 10]. For example, while Tag Clouds (e.g. [26, 44, 16, 28, 27]), may be aesthetically pleasing from a content consumption perspective they are unusable, misrepresentative, and inferior to simple sorted lists [21, 26, 35, 19]. Further, while these and many other solutions [29, 25] help organize content by topic, they do not provide assistance to locate documents based on the relationship between a document's opinions and those in the corpus (are they aligned or divergent). It is in the space of unfulfilled needs that we situate our work on Odin.

Opinion Mining and Mobile Devices

Many research systems have focused on opinion mining [33]. However, the vast majority of the HCI approaches [44, 22]

focus on reviews (e.g. Yelp). There is a large difference, computationally and from a user's perspective, between reviews and full length documents. As a result, many of the algorithmic and design considerations cannot be used directly. Within the document opinion mining literature, only one system, Echo, has investigated something akin to Alignment, Divergence or Relevance (as used in Odin) as a substrate for the UI design [18].

However unlike desktop interfaces, mobile device UI must be optimized both for the smaller screen-size and "on-the-go" use case of users. Thus systems like Echo [18] simply cannot function in a mobile environment (due to visual and interaction complexity and level of detail presented to the user). Until 2012, "none of that work has investigated opinion mining... for a smart phone" [22]. RevMiner[22] provided the first mobile UI system that helps users understand reviews on restaurants. Their UI is optimized for the small screens of mobile devices by utilizing a multi-facet view on the review data, and we can learn many lessons from their work. However, there are two distinct limitations of RevMiner. First, the opinions displayed to users are simply positive or negative. There is no sense of context for said opinions (are they aligned or divergent from the consensus of the corpus). Second, there is a substantial difference between reading short, to the point, reviews and reading long complex documents. Time required, content complexity, and document length greatly impact readability (especially on mobile devices) [12].

Other mobile UI research has focused on mapping multicolumn web content into mobile-friendly versions [43, 24, 45] or tree-like structures [32], or pagination. However, while these approaches do improve readability over the original text, empirical studies have shown that users greatly prefer scrolling text/lists[37, 17] and that users have the fastest information extraction when navigation is broad, with the fewest levels before reaching the desired content [17]. Based on this work, solutions must be created that get users to documents and content quickly, without pagination, while still providing context. The theory behind Odin is that these visual changes and text/document selection can be facilitated through opinion Alignment.

SCOPE & MOTIVATION

Odin is a *document* (not reviews) corpus reading solution for users on the go. Unlike previous approaches, Odin allows mobile users to have extremely easy access to the output of complex NLP and IR analysis on a corpus of documents. Within this scope, Odin is the first system that allows a mobile user to place and find documents and opinions via context within the corpus' distribution of opinions (unlike above solutions that rely upon only word count, topic modeling, or raw sentiment scores for reviews). The additional device constraints (e.g., small screen real-estate) the context of use (e.g. users with greatly decreased time availability to browse and consume) and type of input (documents vs. reviews) greatly change the functioning of our resulting solution, Odin.

In this paper, we will refer to sentences whose opinions are inline with those of the corpus as **Aligned**, and sentences with opinions that go against the grain of the corpus as **Divergent**.

A weighting algorithm determines the rank, or presentation order, of each document and sentence based on how important its topics are, and how much the opinions are Aligned with or Divergent from the corpus. We also created a **Relevance** ranking which can be used to find the sentences or documents that are the most opinionated (regardless of if the opinions are Aligned or Divergent from the corpus). Relevant documents can therefore be thought of as documents that provide a broad perspective.

The authors wish to stress that this approach and it's underlying algorithm are *not* summarization tools. Odin focuses on using opinion alignment to improve document selection, and opinion highlighting, to make for a more meaningful and wide selection of crucial opinions. Odin automates this process by gathering the corpus' consensus (via a corpus of documents), distilling what is being talking about, and examining how the authors refers to those topics. In addition, the authors wish to stress that there is a large, substantive difference between reviews and mining full documents (which impact UI, interactions, and algorithms needed).

DESIGN REQUIREMENTS

Odin builds upon the existing literature above from the research community on reading, economics of attention, and opinion mining. From this background, Odin seeks to address five key design requirements:

- R1 Support Planning Phase of Reading:** Help readers quickly find important facts with little cognitive effort [31] from a range of opinions from the perspective most applicable to a given user's needs
- R2 Support Finding Key Documents:** Reduce cognitive load from managing multiple documents on mobile devices [12] from a range of opinions from the perspective most applicable to a given user's needs
- R3 Mitigate Document Length:** use visual techniques to help users find key sentences in documents regardless of document length [11, 6, 36, 39, 7]
- R4 Document Selection by Alignment:** allow users to find documents based on relation to corpus' opinion (unlike existing solutions which focus on summaries, word count, or raw sentiment scores only)
- R5 Context Based Interaction:** rather than a high-level abstract UI (unlike summarization or sentiment extraction tools), allow users to see all text within an individual document's context [32]

ODIN: DESIGN & INTERACTION

Odin is comprised of a web-based UI, that allows a user interact with documents and the broader corpus on *any* mobile device¹, and a back-end algorithm and system, that performs the data mining analysis of the document corpus. Any type of "document" corpus can be loaded, analyzed, and accessed: PDFs (e.g. legal documents or financial analysts reports),

¹ Screenshots that appear in this paper were taken from an iPhone5, though the system has been tested on Android as well.



Figure 1: Splash Screen & Executive Summary Screenshots

These screenshots also illustrate the three color schemes corresponding to the three ranking options available to the user: Relevant, Aligned, and Divergent

DOCX (e.g. student essays to be graded), HTML (e.g. product discussion blogs, online news, etc)² or streaming temporal documents (e.g. financial news on a company over time). Odin has been tested on a wide variety of data-sets from political articles to financial reports, to the more mundane domain of world/us news.

To illustrate the features and multiple uses of Odin, we present a series of three overly extensive scenarios. In practice, users may apply any subset of the following techniques at any time or in any order. Each scenario involve Anthony, a financial analyst, having progressively less time to react:

Anthony is a junior financial analyst for Kirby, Lee & Co. On October 19th, 2012 Anthony heard about Google's accidental earnings report release³, and the negative Google's financial outlook. Anthony must quickly get a grasp on the flood of financial reports and get into a meeting with his boss to discuss how they should handle their company's investments. A corpus of financial reports was loaded into Odin.

Scenario 1: a Matter of Minutes

As news comes across Anthony's desk about Google's announcement, he gets a worried email from his boss requesting an immediate meeting. Anthony grabs his mobile phone, and heads from his office to Boss' (a short distance away).

Anthony opens up Odin with a dataset of financial articles and is presented with the Odin splash screen (Figure 1a). While Odin has hundreds of documents, Anthony only has time to read one. Whichever document Anthony reads must provide him with the most information possible (R2). On the Odin splash screen, Anthony taps the large green button labeled "Most Relevant." This loads the Executive Summary (Figure 1b) for the most Relevant document (algorithm detailed later), which is entitled *Google shares suspended after accidental*

email wipes We can conceptually think of the "Most Relevant" document, as the document with the broadest spectrum of opinions; those opinions that are on central topics, that are either very Aligned or Divergent with the corpus' conscious.

While the quick tap helped Anthony find a good document, it is still a very long piece of text – too long to read quickly, especially on a mobile device (even with pagination). Much like an executive assistant helps a CEO, Odin displays an Executive Summary with key sentences from that document, sorted in ranked order based on how Relevant they are (R1 & R4). Thus, Anthony can quickly and easily find the most interesting information in the document, covering a broad set of opinions regardless of the length of the original document or where the opinions occurred in the source text (R1 & R3).

Scenario 2: Meeting in a Few

We begin our scenario again, this time Anthony's boss asks for an all hands meeting in ten minutes. Anthony grabs his phone, and heads to the meeting room (he does not want to be late). While he waits for the meeting to begin, he brings up Odin to get up to speed. Given a slightly more relaxed time constraint, Anthony wants to explicitly see what the corpus' conscious is about the Google announcement. In addition, he wants to know what outlying opinions are being voiced so he can be prepared to discuss them.

Rather than click on Relevant, as in the above scenario, Anthony directs he attention to the two other large quick select buttons on the splash screen: "Most Aligned" in gold and "Most Divergent" in red (R4). Similar to the "Most Relevant" button, these other two quick select options will load the "most" Aligned or Divergent document into the Executive Summary view (Figure 1c and 1d respectively), showing the key sentences/opinions in ranked order. Within any of the three Executive Summary layouts, Anthony can tap a given sentence and Odin loads the full document text, highlights

²These are not contributions of this work, and as a result, we do not include screen-shots or a description of the upload infrastructure.

³This scenario in no way reflects opinions on the value of Google stock. This is a test data-sets used to construct a narrative.

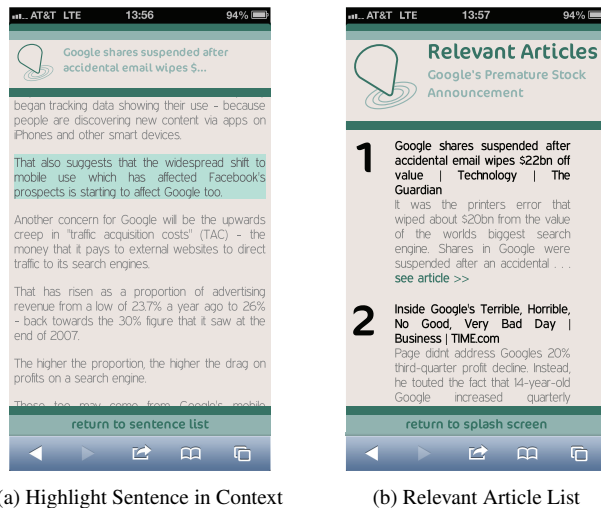


Figure 2: Additional Odin UI Views

The above examples are from the Relevant sort view, and therefore are in green. Other sorts would use differing color schemes as seen in Figure 1d

the tapped sentence, and auto scrolls the full text to the location of the highlight (e.g. Figure 2a) showing that comment in context (R5).

The Aligned Executive Summary (Figure 1c) shows Anthony that the general consensus on Google is that the company has not monetized their mobile platform well. In the Divergent view Figure 1d, the power of Odin's algorithm truly shines; the most Divergent opinion in the most Divergent document is a quote, "Our business had a strong quarter". This is a strikingly different point of view that goes against the grain. Anthony then can tap the sentence, and see the full full context of that sentence (R5), and can easily see that this is a quote from the chief business officer, who is attempting to spin lukewarm results in the best way possible. While this is so different from the consensus, Anthony now can see Google's perspective, why Google thinks they should be viewed positively, and Anthony also now realizes that these views are not shared by the financial analysts.

Scenario 3: On the Train-Ride to Work

The final scenario begins early in the morning (before work), when Anthony gets a message from his boss for a all hands meeting when everyone gets into the office. Anthony has roughly a thirty minute window of time, during his train commute, to get caught up with the financial reports to be discussed. Because the train is not the place to pull out a laptop, he turns on his phone and uses Odin.

While Anthony can use the quick tap buttons, as above, he can utilize his more flexible time to expand his understanding of the issues, and the opinions of the financial analysts⁴. Running along the bottom of the Odin splash screen are three smaller colored buttons (Figure 1a). Each button when clicked, loads the full corpus of documents (R2) sorted by

⁴By tapping the words *full text* along the bottom of the Executive Summary, will load the full text of the document, like Figure 2a, without the highlight.

how Relevant, Aligned, or Divergent the document is (Figure 2b) with the color corresponding to the sort order. Now Anthony can tap any document, and see the Executive Summary for *that* specific document. These Executive Summary screens function just as the ones in the above scenarios.

ALGORITHM & IMPLEMENTATION

The Odin interface is built in HTML, CSS and JavaScript allowing for corpus and document access on any mobile device (regardless of operating system). The server-side corpus analysis and upload system is constructed using Java. This section will discuss the seven-stage server-side algorithm that analyzes a corpus, and generates content for the Odin UI. This algorithm provides the analysis to directly support R4, which facilitates the UI components that satisfy R1 and R2. Odin can load documents in by hand, by being tied to a data base, streaming (e.g. RSS), or crawling the web (e.g. google news collections as corpuses).

Odin's algorithm has two key distinctions from existing research. First, Odin must calculate and situate the alignment of opinions (R4) within the implicit distribution of opinions within a corpus. Second, Odin provides users documents and opinions within a full context (R5) rather than simply displaying the output of a topic model or sentiment analysis. This requires a heuristic that can calculate the alignment at the document and sentence level, not just at the word (output of topic model) or modifier (sentiment analysis) level. This novel cross subject comparison/ranking with a scalar sentiment variation is a complex aspect of our algorithmic approach which builds upon [18].

Stage 1. Extract Content

Regardless of document format (e.g. PDF, HTML, DOCX, etc.), the raw paragraphs, sentences, and words must be extracted in order to perform any ML analysis. We borrow the term **Body Copy** from the layout and design community, to refer to the main text of a document, as compared to logos, images, title, advertisements, comments etc. Body Copy from text, PDF and DOCX files can be extracted with existing Java libraries (e.g. Apache PDFBox). However, HTML (uploaded by a user or crawled from a site like Google News) is more complex due to ads, navigation, comments, variation of page layout, etc. We therefore implemented the CETR algorithm [42] which extracts body-copy from webpages without use of a template through multidimensional clustering.

Stage 2. Identify Discussion Subjects

In order to determine *what* the corpus is talking about and what the most *important* subjects are, we leveraged LDA topic modeling[30] to extract topics and associated keywords across all documents. However, we first cleaned all body copy by filtering out all words that were not nouns using POS tagging⁵. Text was lemmatized using the Stanford NLP toolkit[41], and then fed into LDA.

To explain our use of LDA, we will define A to be the set of all topics ($a_0 \dots a_v$), B to be the set of all documents in our

⁵While an alternative could be Named Entity Recognition[15], entity disambiguation is an open challenge, and we opted to utilize a more conservative approach.

corpus ($b_0 \dots b_m$), and C to be the set of all unique words ($c_0 \dots c_n$) across all documents. Any given document (b_x) is made up of a subset of words from C . Likewise, each topic (a_w) is made up of a subset of words from C , each word having a probability $P(c_y|a_w)$ that if a word is randomly chosen from a_w , c_y is that particular word. Thus we can think of a_w and b_x as being probability distributions $\sum_{y=0}^n P(c_y|a_w) = 1$ and $\sum_{w=0}^v P(a_w|b_x) = 1$ respectively. In layman's terms, every word in a document is contributed by one of the topics.

Thus, LDA gives us the rank ($P(c_y|a_w)$) of each word (c_y) given a particular topic (a_w). However, these word probabilities cannot be compared across topics as they are conditional. We must therefore flip this conditional argument using Bayes' Rule, marginalizing out the topics for each word:

$$P(c_y) = \sum_{w=0}^v P(c_y|a_w)P(a_w) \quad (1)$$

This calculates a rank for each word that is topic independent, comparable across our document corpus, giving us a weight for how important or central each subject is in the corpus⁶.

Stage 3. Detect Opinion Sentiment

Using the subjects identified, we attempt to find any opinions rendered on these important subjects. We consider all sentences that have one of the Stage 2 subjects in them⁷. Using a statistical parser [14] we can create a dependency tree (similar to sentence diagramming in grade school). From these parse trees we programmatically uncover the modifiers (e.g., adjectives) that were applied to any subject (and any negations).

For each modifier uncovered with the statistical parser, we lookup⁸ a sentiment score with Senti-Wordnet[4], a hand-compiled list covering a large selection of common word modifiers. Senti-Wordnet thus associates each modifier on two 0.0-1.0 sentiment scales, Positive and Negative⁹. Thus, for each of the important subjects in Stage 2 (c_y), we uncover the set of opinions $D(c_y) = d_1(c_y) \dots d_u(c_y)$ across every sentence of every document where the subject occurs: .

Stage 4. Calculate Corpus' Opinions

Unlike previous literature on opinion extraction, Odin presents opinions by alignment rather than simply positive or negative. Thus we must next place each unique opinion within the context of the "distribution" of per-subject opinions ($D(c_y)$). Stage 4 calculates that context. From $D(c_y)$, we can calculate a mean positive $\mu_{pos}(c_y)$ and negative sentiment $\mu_{neg}(c_y)$, which we can refer to as the corpus' opinion on a given subject. Given that each per-subject distribution is different, and we will need to compare opinions on different subjects against one another (e.g. "which opinion is more Divergent"), we must normalize. To this end, we calculate each per-subject standard deviation ($\sigma_{pos}(c_y)$ and

$\sigma_{neg}(c_y)$) from the mean. As a result, all comparisons will be a measure of standard deviations away rather than absolute distance. We therefore can define opinion distance $E(d_z(c_y)) = < E_{pos}(d_z(c_y)), E_{neg}(d_z(c_y)) >$:

$$\begin{aligned} E_{pos}(d_z(c_y)) &= \frac{|d_{pos,z}(c_y) - \mu_{pos}(c_y)|}{\sigma_{pos}(c_y)} \\ E_{neg}(d_z(c_y)) &= \frac{|d_{neg,z}(c_y) - \mu_{neg}(c_y)|}{\sigma_{neg}(c_y)} \end{aligned} \quad (2)$$

Stage 5. Instance Alignment and Divergence

Given the above calculated context, we can now determine how Aligned or Divergent a given opinion is. Whereas $D(c_y)$ is the set of all opinions on a given word, we define the set of all opinions across all words in all documents as $F = \bigcup_{q=0}^p (D(c_y))$. We therefore define an individual opinion on a specific instantiation of a word as $f_q = d_z(c_y)$. And therefore, f_q can be substituted for any $d_z(c_y)$ allowing us to easily calculate an Alignment (**G**) and Divergence (**H**) score:

$$\begin{aligned} E_S(f_q) &= \text{MAX}(E_{pos}(f_q), E_{neg}(f_q)) \\ G_{opinion}(f_q) &= G_{opinion}(d_z(c_y)) \propto P(c_y) * E_S(f_q) \\ H_{opinion}(f_q) &= H_{opinion}(d_z(c_y)) \propto \frac{P(c_y)}{E_S(f_q)} \end{aligned} \quad (3)$$

These scoring functions were based on two principals: First, as Alignment should grow as distance from the corpus' opinion shrinks (hence in the denominator) while Divergence should shrink (hence as a multiplier); Second, both Alignment and Divergence grow as importance of the subject increases. In addition, we can consider an individual opinion of a specific instance of a word as being "Aligned" if $G > H$ (and "Divergent" if $H > G$).

Stage 6. Sentence Alignment, Divergence & Relevance

The inclusion of subject importance ($P(c_y)$) in our scoring functions in Stage 5 has an added benefit – both $G_{opinion}$ and $H_{opinion}$ become subject independent, and can be compared or combined at a sentence (or document) level with the $G_{opinion}$ and $H_{opinion}$ of another subject (which may have a different weight in the model from Stage 2). We can therefore define the set of all subject opinions ($\{f_q\}$) within a given sentence as $I = \{f_q\}$. Subsequently, the Alignment or Divergence of the sentence is a sum over all opinions' Alignment or Divergence scores:

$$\begin{aligned} G_{sentence}(I) &= \sum_{r=0}^{|I|} G_{opinion}(i_r) \\ H_{sentence}(I) &= \sum_{r=0}^{|I|} H_{opinion}(i_r) \end{aligned} \quad (4)$$

As described earlier, we defined Relevance as sentences or documents that have a broad set of opinions. To calculate a Relevance score (**J**), we combine both the Alignment and Divergence scores: $J_{sentence}(I) = G_{sentence}(I) + H_{sentence}(I)$. Thus, the more opinions rendered (Alignment, Divergence, or both) on the more important subjects, the higher the appropriate score will be, and are sorted accordingly based on their raw scores.

⁶To reduce complexity, and noise from topic modeling, we only consider the top 500 words.

⁷By filtering sentences, this improves run/computation time.

⁸Modifier lookup does not have word sense disambiguation (context). While there are many other sentiment calculation alternatives, Senti-Wordnet is a well regarded and robust system that provides fast and relatively accurate sentiment scoring.

⁹We maintain both positive and negative values because if combined on a single scale, most results trend towards neutral. Any negations detected will "swap" the positive and negative scores. Neutral opinions (both scores being 0) are dismissed.

Stage 7. Document Alignment, Divergence & Relevance

Calculating Alignment, Divergence or Relevance scores for documents follows a similar approach to that in Stage 6. We define the set of all opinions in a document as $\mathbf{K} = \{I\}$, with scoring functions using sum, rather than mean, so as to give more weight to documents that have many opinions, rather than one strong opinion:

$$\begin{aligned} G_{doc}(I) &= \sum_{o=0}^{|K|} G_{sentence}(k_o) \\ H_{doc}(I) &= \sum_{o=0}^{|K|} H_{sentence}(k_o) \\ J_{doc}(I) &= G_{doc}(K) + H_{doc}(K) \end{aligned} \quad (5)$$

Documents, unlike sentences, have the potential to contain more opinions simply because they have more words (not because the document is more opinionated)¹⁰. Consider a corpus where all documents contain at least 1% Divergent comments. A longer document, which is largely Aligned, may appear to also be very Divergent, simply because of its length (as compared to a short document which is 95% Divergent, but so short that the sum is smaller than the longer document). We therefore cannot rely upon the raw Alignment, Divergence scores. Rather we must look at documents that are highly Aligned or Divergent, which can be calculated as the delta between the two raw scores: $\delta_{Aligned}(K) = G_{doc}(K) - H_{doc}(K)$ and $\delta_{Divergent}(K) = H_{doc}(K) - G_{doc}(K)$. Thus, documents whose Alignment score is much larger than Divergent score will appear more Aligned, and documents whose Divergent score is much larger than Alignment score will appear more Divergent.

PILOT USER STUDY

An pilot study was conducted to understand how well our system, algorithm and UI could support corpus access for a mobile users and how Odin could be improved.

10 users participated (8 male, 2 female), and were shown a demonstration of Odin, its features and capabilities with multiple corpuses on finance, technology, and news¹¹. Though a straw-man, we also showed users a page styled like Google News “more sources” (listing multiple documents on the same news story), and demonstrated how different news/article websites (e.g. CNN, ABC, etc.) rendered articles for mobile devices. Participants were given 5-10 minutes to explore Odin. During the informal 20-30 minute sessions, participants were encouraged to think aloud. At the end of the session, participants completed a brief questionnaire about their experience. Though our sample size is too small to conduct formal statistics (due to low confidence intervals) we do report quantitative responses to illustrate user reactions and comparisons to other approaches (Table 1).

¹⁰There are many IR techniques to normalize for a document’s length itself. However, longer documents may simply have more sentences worth reading, and help a user quickly find key opinions, which is Odin’s goal. We therefore did not account for document length itself.

¹¹One of the corpuses users were shown, was the same document corpus as highlighted in the figures in this paper.

Finding Documents	Odin is...		Equal	Other is...	
	Best (1)	Better (2)	(3)	Better (4)	Best (5)
Google News	20%	60%	10%	10%	0%
Date Sorted List	40%	50%	10%	0%	0%
Popularity Sorted List	40%	50%	10%	0%	0%

Reading Documents					
Google News	20%	50%	20%	10%	0%
Date Sorted List	20%	40%	30%	10%	0%
Popularity Sorted List	30%	30%	20%	20%	0%

Table 1: Which Method Do You Prefer For ...

Participants were asked to respond on a 5 point scale

All were recruited from a large US corporation, had a mean age of 41.1. All of our participants reported using large collections of documents daily or a few times a week. Participants had varying educational backgrounds, both degree level (1 BS, 3 MS/MA, 6 Doctorate) and degree area (5 CS, 2 Statistics, 1 Math, 1 Business, 1 Engineering).

Results & Discussion

Participants were overwhelmingly positive in their feedback on Odin with respect to its ability to manage a large corpus:

It had a fluid look and feel that allowed for the rapid exposure to a number of articles with little thought about trying to “find” them. - P2

As a boss would say... ‘give me the five minute rundown,’ and you are asking for that, automatically. But now, you don’t have to read everything to get that. You don’t have to go ‘not relevant, not relevant, not relevant’... if you don’t have that hour, you can just go through it - which is great! - P7

and Odin’s Executive Summary UI, that quickly brings key opinions within each document to the forefront:

[Executive summary] with aligned and divergent key sentences gives me a better overview than the standard abstract or summary - P10

Very useful, that it infers the main ideas and opinions, to give a higher-level understanding of the context of the corpus as a whole before I start to look at documents on a more detailed level. - P8

We asked participants “If you could load any corpus into Odin or Google News’s List View for mobile reading, which would you choose?” 90% of participants chose Odin. Responses to other comparisons are included in Table 1.

Participants were also generally positive about the different features of Odin (Table 2). While the reaction was positive, we explicitly asked users what additional features could be useful to readers on the go. Surprisingly, each participant proposed a different idea:

- A quick tap button for documents that were read by many people viewing this corpus (social popularity)
- See articles that are aligned/divergent from the currently open document (relative alignment)
- Adapt the ranking based on individual user’s viewing preferences or histories across multiple corpuses
- Summary statistics (e.g. number of Aligned opinions on topic A)

We asked participants what they thought of suggestions we heard from prior subjects. Unexpectedly, participants generally did not feel the others’ suggestions would be valuable. Given that each participant justified their additional feature

	1	2	3	4	5
Different Executive Summaries (Relevance, Alignment, Divergence)	80%	10%	10%	0%	0%
Only Showing Key Sentences In Executive Summaries	50%	30%	10%	10%	0%
Seeing Each Sentence in Context	50%	40%	0%	10%	0%
Most Relevant Document	50%	40%	0%	10%	0%
Most Aligned Document	30%	40%	30%	0%	0%
Most Divergent Document	50%	20%	30%	0%	0%

Table 2: How Useful Is Each Feature of Odin

Participants were asked to respond on a 5 point scale,
1 was a "Key Feature" and 5 was "Provides no value"

within their own unique work or situation, we believe that this suggests that the features currently in Odin are universally critical, though additional features may be useful if turned on based on individual preference. This was echoed by P9:

In a world where you have inundation of information, and opinions you could spend 'how many?' hours a day trying to read much less put it together... I like it... and I think a lot of these [features] if you don't have them, you lose a lot. I would say most of [the features] are key features. Cause if you're not, then your missing too much. What you are looking for is a minimum ante to play the game... Odin solves that. - P9

P1 discussed an interesting perspective of summarization UI (e.g. tag clouds, or summaries) versus Odin's approach of Alignment and Divergence (R4).

This is probably something generalizable, usually you get the official opinion... you see the same thing over and over again, but if someone says something different, now that is interesting so it is important that you find it quickly - P1

He continued to articulate that traversing a corpus is more than just seeing key words, but seeing relative opinions.

All users also commented about the elegance and simplicity of the Odin UI. Often these comments were associated with praise that something could be so simple despite a powerful back-end and service (R1), especially for a mobile system:

It appears to allow for personalization of how the information is presented and allows it to meet the demands of the moment instead of dictating what you "should" be interested. - P2

It is important to note that even while Odin was built using the most straightforward approaches used in the ML and NLP domains, participants explicitly stated that the key sentences were helpful and appeared to be picked out well by the system:

It behaves like a "smart" system. For what I've seen, results are always appropriate. - P3

Many users even commented how strikingly different the documents were, and how they immediately saw the distinctions between the three ranking/categories. We strongly believe that this is due to the power of aggregating opinions based on the corpus' consensus, combining both subject importance in a topic model with sentiment detection and delta from the mean.

Overall, results from our study provide very promising. Encouraged by these results, a formal study was then conducted.

COMPARATIVE USER STUDY

Based on the feedback from our initial pilot user study, we conducted a second user evaluation of Odin¹². This study directly compared Odin to the only other research system that enables opinion mining on mobile devices, RevMiner[22]. We re-implemented the RevMiner system based on the description in the original paper¹³. We acknowledge that RevMiner was designed to synthesize Yelp Reviews, which are an extremely different type of data as compared to full length documents. While RevMiner may not be designed to function in the same use case as Odin, it is the only work that "has investigated opinion mining as a substrate for a smartphone interface,"[22]. In addition, by comparing Odin to RevMiner, users can contrast 'traditional' keyword-modifier pair/tag cloud style UI and the Executive Summary/Alignment in Odin. As in the first experiment, this study also presented users with a Google News style list UI as a straw man.

14 users participated (6 male, 8 female), and were able to experience each system (Odin, RevMiner, and the Google News styled UI) in turn¹⁴. For each system, users were given a demo, and then allowed to use it to explore a handful of corpuses. Users could select the corpuses they wished to read from a pre-loaded collection of 20 corpuses ranging from Financial Analysis, to Political Documents to more germane World and US news¹⁵. To motivate actual use of the three systems, we asked participants to try to summarize what the corpus was about after opening it (though this was not recorded).

At the conclusion of each session, participants were given a brief survey about the three systems, and specific aspects of their design. The survey contained a SUS usability metric[8], as well as more open-ended qualitative questions.

All were recruited from a large US corporation's Legal, Marketing a Consulting divisions (the target users of a system like Odin), had a mean age of 31.7. Participants reported using large collections of documents daily, and had a variety of educational attainment: 6 had MS/MA, 6 BS/BA, 1 Doctorate or Equivalent Degree.

Results & Discussion

Participant responses were all strongly in favor of Odin over the other two solutions both in perceived value-add within their own work (Table 3b) and usability (Table 3a)¹⁶, with almost all differences being statistically significant (Table 3c). Likewise, all participants selected Odin as the "best" system, the one they would choose to use in their day-to-day work. P4 described/justified her selection of Odin by stating:

RevMiner feels like a low-res version of Odin... It's like you pulled out all the useful features and obscured everything that matters.

¹²Based on user feedback, one minor change was made to Odin; The letters R, A, D we added to the "see more document" circles. Each circle corresponds to the corpus list sort order (Relevance, Alignment and Divergence respectively) by color and typography.

¹³The 'Categories' tab was omitted as their is does not have a logical parallel in a document corpus.

¹⁴Presentation order was randomized to control for order effects

¹⁵Corpuses contained from 100 to 300 documents each

¹⁶It is worth noting that while both Google UI and Odin have similar SUS scores (are equally usable), the benefits of Odin's additional synthesis improves its value add.

		Mean (SD)	Median [IQR]			Mean (SD)	Median [IQR]			Value Add	SUS Score
Odin	Overall	86.79 (12.39)	87.5 [77.5, 97.5]	RevMiner	Overall	63.93 (11.07)	62.5 [58.75, 66.25]	Google	Overall	4.71 (0.49)	5 [4.5, 5]
	Main Screen	89.64 (10.55)	90 [83.75, 98.75]		Common	67.5 (13.54)	75 [61.25, 76.25]		Article List	2.14 (0.9)	2 [2, 2]
	Executive Summary	89.29 (12.14)	92.5 [82.5, 98.75]		Special	55 (6.12)	52.5 [52.5, 60]		External Links	3.43 (0.98)	3 [3, 4]
RevMiner	Overall	63.93 (11.07)	62.5 [58.75, 66.25]	Google	Overall	84.29 (9.43)	87.5 [76.25, 90]	Odin vs. RevMiner	Odin vs. RevMiner	< 0.001	0.01
	Common	67.5 (13.54)	75 [61.25, 76.25]		Article List	84.29 (19.19)	97.5 [67.5, 100]		Odin vs. Google	0.01	0.26
	Special	55 (6.12)	52.5 [52.5, 60]		External Links	66.79 (14.41)	60 [58.75, 73.75]		RevMiner vs. Google	0.002	0.01

(a) SUS Usability Scores

(b) System Value-Add in Work Context

(c) T-Test Comparison P-Values

Table 3: Comparative User Study Results

SUS scores range from 0 to 100. 90-100 = A, 80-89 = B, etc.[5] — Value-Add Questions asked on 5 point scale (1=Low, 5= High) — T-Test Compared Overall SUS Scores

This view was echoed by P6 (R2, R3 R4):

when i am looking for something i need to read, i have 5 or 10 minutes. i want to get the most knowledge in that amount of time. So when i open the app, i want the relevance and alignment thing.

These views and SUS scores overall, and for specific features, suggest that more abstract visual presentations (e.g. tag-clouds and word-modifier pairs) do not support the real-world tasks of knowledge workers who prefer the Executive Summary style of presentation based on Alignment. This is largely consistent with the literature discussed earlier. Within the extremely simple Google styled list UI (which had high SUS scores), participants preferred the Executive Summary and Alignment driven approach. P17 justified this by stating: “I like the Executive Summary, not just [seeing] the first few lines [of the article shown in the Google UI].” So while both are “usable” one has much more value (R3).

While there is a noticeable difference between Odin and RevMiner, we do not believe that these usability issues are generally true for the RevMiner system in its original use case. Rather, we believe that more traditional approaches to opinion mining (as used by RevMiner) do not map onto document corpuses on mobile devices. Along this line, P1 stated that “Revminer just wasn’t very useful, it’s not that it wasn’t hard to understand.” This difference, highlights the need for new algorithmic and UI approaches when handling different types of textual data (e.g. reviews vs. documents).

LIMITATIONS & FUTURE WORK

Given the informal nature of the first study, and the unstructured exploration (rather than explicit task) in the second, the findings should not be interpreted as a final validation of Odin. However, these results do provide an insight into feature utility, and the potential use of Odin in the real world. Presently, we are undertaking a long-term ecological deployment of Odin, integrating it within a corporate environment. This will allow us to better understand how Odin quantitatively impacts job performance.

While Odin leverages standard Machine Learning algorithms to perform its analysis, there are many additional topic and sentiments analysis libraries on the “bleeding edge.” As Odin development continues and expands, we aim to continue to integrate more advanced APIs and techniques or sentiment analysis (taking sentence context into account), keyword ex-

traction (with n-gram phrases and Named Entity Recognition) and topic modeling (combining supervised and unsupervised techniques). As stated above, however, participants’ initial reaction to Odin’s performance was quite good.

Lastly, another direction of future work would be the inclusion of social aspects of corpus reading as suggested by many of our participants. While Odin focuses on the written corpus itself to weight rankings, we wish to explore how we can integrate user’s view habits into the algorithm to “bubble up” popular articles and opinions. Further, we wish to explore taking source “credibility” into the algorithm’s weight. Whether the source is the enterprise hierarchy of the author, or the veracity of the journal/author from more public domains, we believe source can play a powerful role.

CONCLUSION

The digital age has resulted in increased informational awareness, which has caused a growing scarcity of human time and attention. This problem is exacerbated by the growing ubiquity of mobile devices as a major information consumption platform, in that they have smaller screen real-estate and mobile device users have far less time to consume content. In this paper we presented Odin, a fundamentally new approach to mitigating information overflow by creating an algorithm and mobile UI that allows a user to view a corpus from three distinct perspectives; Relevance, Alignment and Divergence. Thus, rather than simply seeing a summary of a corpus keywords, Odin lets a user quickly access key opinions and documents by their relation to the corpus’ conscious. Odin is fundamentally a different approach from the standard summarization, topic modeling, or sentiment analysis, and is deeply grounded in a breadth of literature from many disciplines. Two user studies provide strong support for the quality and utility of a system like Odin within real-world scenarios. Thus, the foremost contribution of this work is an interaction model and algorithmic approach that allows users to place each document and opinion within the larger corpus of opinions while “on the go.”

REFERENCES

1. E. Adar and B. A. Huberman. The economics of surfing. *Quarterly Journal of Electronic Commerce*, 2000.
2. F. Ahmed, Y. Borodin, A. Soviak, M. Islam, I. V. Ramakrishnan, and T. Hedgpeth. Accessible skimming: faster screen reading of web pages. In *UIST ’12*. ACM Request Permissions, Oct. 2012.

3. N. Archak, A. Ghose, and P. G. Ipeirotis. Show me the money!: deriving the pricing power of product features by mining consumer reviews. In *KDD '07*. ACM Request Permissions, Aug. 2007.
4. S. Baccianella, A. Esuli, and F. Sebastiani. 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.
5. A. Bangor, P. Kortum, and J. Miller. Determining what individual SUS scores mean: Adding an adjective rating scale. *Journal of usability studies*, 2009.
6. P. Baudisch, B. Lee, and L. Hanna. 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.
7. D. Billsus, D. M. Hilbert, and D. Maynes-Aminzade. Improving proactive information systems. In *the 10th international conference*, pages 159–166, New York, New York, USA, 2005. ACM Press.
8. J. Brooke. SUS-A quick and dirty usability scale. *Usability evaluation in industry*, 1996.
9. A. J. B. Brush, D. Barger, A. Gupta, and J. J. Cadiz. Robust Annotation Positioning in Digital Documents. In *the SIGCHI conference*, pages 285–292, New York, New York, USA, 2001. ACM Press.
10. G. Carenini, R. T. Ng, and A. Pauls. Interactive Multimedia Summaries of Evaluative Text. In *the 11th International Conference*, page 124, New York, New York, USA, 2006. ACM Press.
11. M. Chakravarthy and P. Ciravegna. Cross-media document annotation and enrichment. 2006.
12. Y. Cui and V. Roto. How people use the web on mobile devices. In *WWW '08*. ACM, Apr. 2008.
13. S. Davies, S. Allen, J. Raphaelson, E. Meng, J. Engleman, R. King, and C. Lewis. Popcorn: the personal knowledge base. In *the 6th ACM conference*, pages 150–159, New York, New York, USA, 2006. ACM Press.
14. M. C. De Marneffe and B. MacCartney. Generating typed dependency parses from phrase structure parses. In *Proceedings of LREC '06*, 2006.
15. J. R. Finkel, T. Grenager, and C. Manning. Incorporating non-local information into information extraction systems by gibbs sampling. *ACL'05*, 2005.
16. K. A. Ganesan, N. Sundaresan, and H. Deo. Mining tag clouds and emoticons behind community feedback. In *WWW '08*. ACM, Apr. 2008.
17. V. Giller, R. Melcher, J. Schrammel, R. Sefelin, and M. Tscheligi. *Usability Evaluations for Multi-device Application Development Three Example Studies*, volume 2795 of Verena Giller, Rudolf Melcher, Johann Schrammel, Reinhard Sefelin and Manfred Tscheligi. Lecture Notes in Computer Science, Berlin, Heidelberg, 2003.
18. J. Hailpern and B. A. Huberman. Echo: the editor's wisdom with the elegance of a magazine. In *EICS '13: Proceedings of the 5th ACM SIGCHI symposium on Engineering interactive computing systems*. ACM Request Permissions, June 2013.
19. M. J. Halvey and M. T. Keane. An Assessment of Tag Presentation Techniques. In *the 16th international conference*, page 1313, New York, New York, USA, 2007. ACM Press.
20. M. A. Hearst. User interfaces and visualization. *Modern information retrieval*, 1999.
21. M. A. Hearst and D. Rosner. Tag clouds: Data analysis tool or social signaller? ... on *System Sciences*, 2008.
22. J. Huang, O. Etzioni, L. Zettlemoyer, K. Clark, and C. Lee. RevMiner: an extractive interface for navigating reviews on a smartphone. In *UIST '12*. ACM Request Permissions, Oct. 2012.
23. B. A. Huberman and F. Wu. The economics of attention: Maximizing user value in information-rich environments. *Advances in Complex Systems*, 2008.
24. B. Kim and B. Aeschliman. Experimental assessment of a visual interface designed for web documents on mobile devices. *ICCIT 2011*, pages 231–236, 2011.
25. T. Kriplean, J. Morgan, D. Freelon, A. Borning, and L. Bennett. Supporting reflective public thought with considerit. In *CSCW '12*. ACM Request Permissions, Feb. 2012.
26. B. Y.-L. Kuo, T. Hentrich, B. M. Good, and M. D. Wilkinson. Tag clouds for summarizing web search results. In *the 16th international conference*. ACM, May 2007.
27. S. E. Lee, T. Chun, and S. S. Han. Using Qtag to Extract Dominant Public Opinion in Very Large-Scale Conversation. In *2009 International Conference on Computational Science and Engineering*, pages 753–758, IEEE, 2009.
28. S. E. Lee, D. K. Son, and S. S. Han. Qtag: tagging as a means of rating, opinion-expressing, sharing and visualizing. In *SIGDOC '07*. ACM Request Permissions, Oct. 2007.
29. B. Liu, M. Hu, and J. Cheng. Opinion observer: analyzing and comparing opinions on the Web. In *the 14th international conference*, page 342, New York, New York, USA, 2005. ACM Press.
30. A. K. McCallum. Mallet: A machine learning for language toolkit. 2002.
31. K. O'Hara and A. Sellen. A Comparison of Reading Paper and On-Line Documents. In *the SIGCHI conference*, pages 335–342, New York, New York, USA, 1997. ACM Press.
32. J. Otterbacher, D. Radev, and O. Kareem. News to go: hierarchical text summarization for mobile devices. In *SIGIR '06*, 2006.
33. B. Pang and L. Lee. Opinion Mining and Sentiment Analysis. *Foundations and Trends in Information Retrieval*, 2(1-2), Jan. 2008.
34. S. Park, M. Ko, J. Kim, and H. Choi. NewsCube2. 0: An Exploratory Design of a Social News Website for Media Bias Mitigation. In *Workshop on Social Recommender Systems*, Mar. 2011.
35. A. W. Rivadeneira, D. M. Gruen, M. J. Muller, and D. R. Millen. 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.
36. B. Sereno, S. B. Shum, and E. Motta. ClaimSpotter: an environment to support sensemaking with knowledge triples. In *the 10th international conference*. ACM Request Permissions, Jan. 2005.
37. S. Shrestha. Mobile Web Browsing: Usability Study. In *the 4th international conference on mobile technology, applications, and systems and the 1st international symposium*, page 187, New York, New York, USA, 2007. ACM Press.
38. H. A. Simon. Designing organizations for an information-rich world. *Computers, Communication, and the Public Interest*, (The Johns Hopkins Press), 1969.
39. B. Suh, A. Woodruff, R. Rosenholtz, and A. Glass. 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.
40. J.-T. Sun, X. Wang, D. Shen, H.-J. Zeng, and Z. Chen. CWS: a comparative web search system. In *WWW '06*. ACM, May 2006.
41. The Stanford Natural Language Processing Group. Stanford CoreNLP.
42. T. Weninger, W. H. Hsu, and J. Han. CETR: content extraction via tag ratios. *WWW 2010*, 2010.
43. X. Xie, G. Miao, R. Song, J.-R. Wen, and W.-Y. Ma. Efficient Browsing of Web Search Results on Mobile Devices Based on Block Importance Model. In *PERCOM '05*. IEEE Computer Society, Mar. 2005.
44. K. Yatani, M. Novati, A. Trusty, and K. N. Truong. Review Spotlight: A User Interface for Summarizing User-generated Reviews Using Adjective-Noun Word Pairs. In *the 2011 annual conference*, page 1541, New York, New York, USA, 2011. ACM Press.
45. X. Yin and W. S. Lee. Using link analysis to improve layout on mobile devices. In *WWW '04*. ACM, May 2004.