ReCloud: Semantics-Based Word Cloud Visualization of User Reviews

Ji Wang[a,*] Jian Zhao[b,†] Sheng Guo[a,‡] Chris North[a,§] Naren Ramakrishnan[a,¶]

[a]Department of Computer Science, Virginia Tech [b]Department of Computer Science, University of Toronto

Abstract

User reviews, like those found on Yelp and Amazon, have become an important reference for decision making in daily life, for example, in dining, shopping and entertainment. However, large amounts of available reviews make the reading process tedious. Existing word cloud visualizations attempt to provide an overview. However their randomized layouts do not reveal content relationships to users. In this paper, we present ReCloud, a word cloud visualization of user reviews that arranges semantically related words as spatially proximal. We use a natural language processing technique called grammatical dependency parsing to create a semantic graph of review contents. Then, we apply a force-directed layout to the semantic graph, which generates a clustered layout of words by minimizing an energy model. Thus, ReCloud can provide users with more insight about the semantics and context of the review content. We also conducted an experiment to compare the efficiency of our method with two alternative review reading techniques: random layout word cloud and normal text-based reviews. The results showed that the proposed technique improves user performance and experience of understanding a large number of reviews.

Index Terms: H.5.2 [Information interfaces and presentation (e.g., HCI)]: User Interfaces—Natural language

1 Introduction

Many websites, such as Amazon and Yelp, provide customers with a platform for sharing product reviews, which has become a critical references resource for making decisions. However, the usefulness of those reviews is limited in practice, because reviews exist in large quantities and the detailed contents are unstructured (i.e., in plain text form). People find it tedious and time-consuming to read a large amount of text, so they either leverage the structured quantitative aspects of reviews, such as star ratings, or quickly skim the text, both of which overlook important information for decision making.

Word clouds (or tag clouds) are popular methods for visually summarizing large amounts of text, which presents the content in a space-filling, concise and aesthetically appealing manner, with the font size and color of words mapped to the word frequency, popularity or importance. Word cloud visualizations have been widely used in both business and research, e.g., Opinion Cloud [7], Tirra [15], Review Spotlight [26] and Wordle [8, 12].

However, most word clouds arrange the words randomly. Although they are useful and informative tools, the randomness of word layout does not provide a meaningful representation of the data. First, it requires significant mental demand for users to understand the review content, because users need to scan the entire visualization to gain an overview or to find specific keywords of interest. Second, it only provides one dimension of information, such as word frequency, without semantic relationships among keywords, which is critical for understanding the review content [14, 26]. For example, related words in a single concept ”chicken salad sandwich” or description ”sushi is delicious” could be placed at different places in a random word cloud, making it difficult for users to recognize the intent.

In this paper, we present ReCloud, a word cloud visualization of user reviews, which seamlessly integrates semantic context of review keywords into the visualization layout. This provides an important additional dimension of information for users to better comprehend reviews in a quick and easy manner (Figure 1). For example, users will recognize that ”delicious” goes with ”sushi”, and that ”chicken” and ”salad” and ”sandwich” are a single concept. Our layout algorithm is based on grammatical dependency parsing, an effective natural language processing (NLP) approach [6] that generates a *grammatical dependency graph* (GDG) from text, which has been used in many applications to enhance users' understanding of textual sources [5, 24]. We propose a novel approach of (1) parsing user review semantics to generate a GDG, (2) clustering the GDG with a force-directed graph layout algorithm based on an energy model, and (3) embedding the clustered GDG into the word cloud. Using real review data from the Yelp Academic Dataset [27], we also conduct a formal experiment to compare ReCloud with two alternative review browsing techniques: (a) normal text reading, and (b) random layout word cloud. The results indicate that ReCloud improves user performance and experience in exploring reviews, identifying criteria, and making decisions.

2 Related Work

2.1 Review Visualization

Visualization of online reviews can be categorized into two types. First, visualization of quantitative features of reviews are often used to display customer ratings, price level, and other numerical measurements of a product or service. For example, Wu *et al.* presented a system to show hotel user feedback based on quantitative review features [25]. However, many products or services cannot be simply described with quantitative values in reviews; deeper insights about actual review content are needed for users to make better decisions.

Second, visualizations of the textual content of reviews can provide a deeper view. Liu and Street first used a NLP approach to analyze reviews and then present extracted user opinions using a bar chart [14]. Along the same lines, Caternini and Rizoli presented a multimedia interface to visualize fixed features summarized from review contents that reflect a user's opinions [2]. Review Spotlight presents a word cloud based visualization by showing adjective plus noun word pairs with color-coded word sentiment [26]. Huang *et al.* proposed RevMiner, a smartphone interface that also applies NLP techniques (e.g., bootstrapping) to analyze and display user reviews in a categorical layout [10].

The major advance in ReCloud is that the NLP context is reflected in the spatial layout of the tag cloud. Thus, in general, the spatial proximity of tags in the cloud represents the frequency and path length between the tags in the NLP grammatical parse of all review text. In contrast, RevMiner uses categorical and sorted lists, and Review Spotlight uses randomized layout.

[*] e-mail: wji@cs.vt.edu

[†] e-mail: jianzhao@dgp.toronto.edu

[‡] e-mail: guos@vt.edu. Sheng Guo is at LinkedIn Corp. now.

[§] e-mail: north@vt.edu

[¶] e-mail: naren@cs.vt.edu

Graphics Interface Conference 2014
7-9 May, Montreal, Quebec, Canada

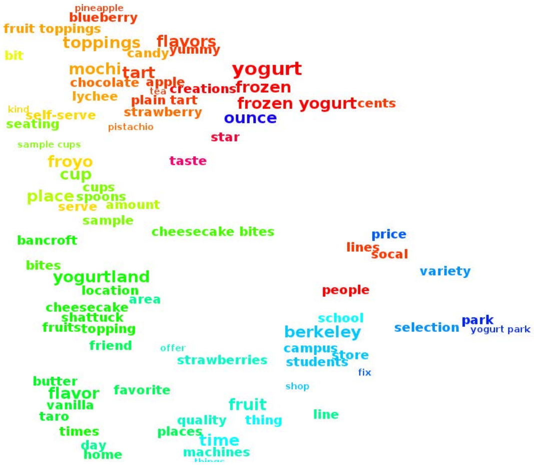


Figure 1: ReCloud of a Yogurtland store near UC Berkeley campus.

2.2 Word Cloud Layout and Evaluation

Word clouds have become very popular in showing textual content, where the font size of a keyword could reflect its frequency in the text. There exist many approaches for the layout of words. Kaser and Lemire presented an algorithm to draw word clouds in a limited space on webpages using HTML table components [11]. Viegas *et al.* proposed a greedy space-filling approach for placing words that generates more compact word clouds [8, 23].

Recently, researchers have proposed several methods for embedding text NLP results into word cloud layouts, for example, as in Spotlight [26] and RevMiner [10] mentioned above. In addition, Cui *et al.* present a context preserving tag cloud for news based on term co-occurrence in both time and text sentences, using a statistical information theoretic approach [4]. ProjCloud clusters documents into a set of polygons, then fills the polygons with high frequency keywords from those documents and arranges the keywords according to statistical co-occurrence distance metric [20]. ReCloud goes beyond simple co-occurrence metrics to semantic grammatical structure of the text and focuses on the term level, not the document level.

Moreover, several studies have been conducted to assess the effects of word clouds on text browsing tasks. For example, font size and font weight were found to catch a user's attention the most [1]. Lohmann *et al.* evaluated the effect of word cloud layouts on user task performance and found that thematic layouts were good for finding words that belonged to a specific topic [16]. However, their work was not based on real data. Rivadeneira *et al.* conducted a user study to obtain performance metrics of four types of tasks using word clouds, including search, browsing, impression formation, and recognition [21], where impression formation was later adopted in the evaluation of Review Spotlight [26].

3 ReCloud

3.1 System Overview

We designed ReCloud following two principles summarized from the previous work [12, 20, 26]: 1) the word cloud should arrange its layout to present semantic information about the text, and 2) the word cloud should support interaction for retrieval of review content. The entire ReCloud system consists of two main parts: the back-end data processing pipeline and the front-end interactive visualization.

The back-end data processing pipeline takes raw user reviews as the input and generates a word cloud visualization. The pipeline contains the following three steps:

1. **Grammatical dependency parsing.** We first process the review content using NLP techniques to generate the grammatical dependency graph that reflects the semantic relationships between keywords in the reviews.
2. **Initial word cloud layout.** We apply the LinLogLayout algorithm [18] to the grammatical dependency graph, using an energy model to optimize the force-directed graph layout process [17]. This generates keyword clusters and their initial layout positions.
3. **Final word cloud rendering.** After the initial word placements, we then use an approach similar to Wordle [23, 8] to perform fine-grain adjustments to the word cloud. This avoids word overlapping in the final visualization.

As shown in Figure 2(c), the front-end visualization contains three main components: a main view of showing the word cloud (F), a historical view of keywords clicked by the user (G), and a detail view of review texts (H). ReCloud also supports basic user interaction for accessing review content based on keywords. When a user clicks a word tag in the word cloud, Component H shows all the reviews that contain the keyword, where the keyword is highlighted in red color.

3.2 Data

In this paper, we used the Yelp Academic Dataset [27] as our test bed. The dataset provides profiles and user reviews of 250 businesses near 30 universities, such as shopping centers and restaurants. The data includes three objects in JSON format: *business profile objects*, *user profile objects* and *review content objects*. We utilized the business profile objects to select businesses in our experiment (see Section 4) and the review content objects to generate the word cloud visualization in ReCloud.

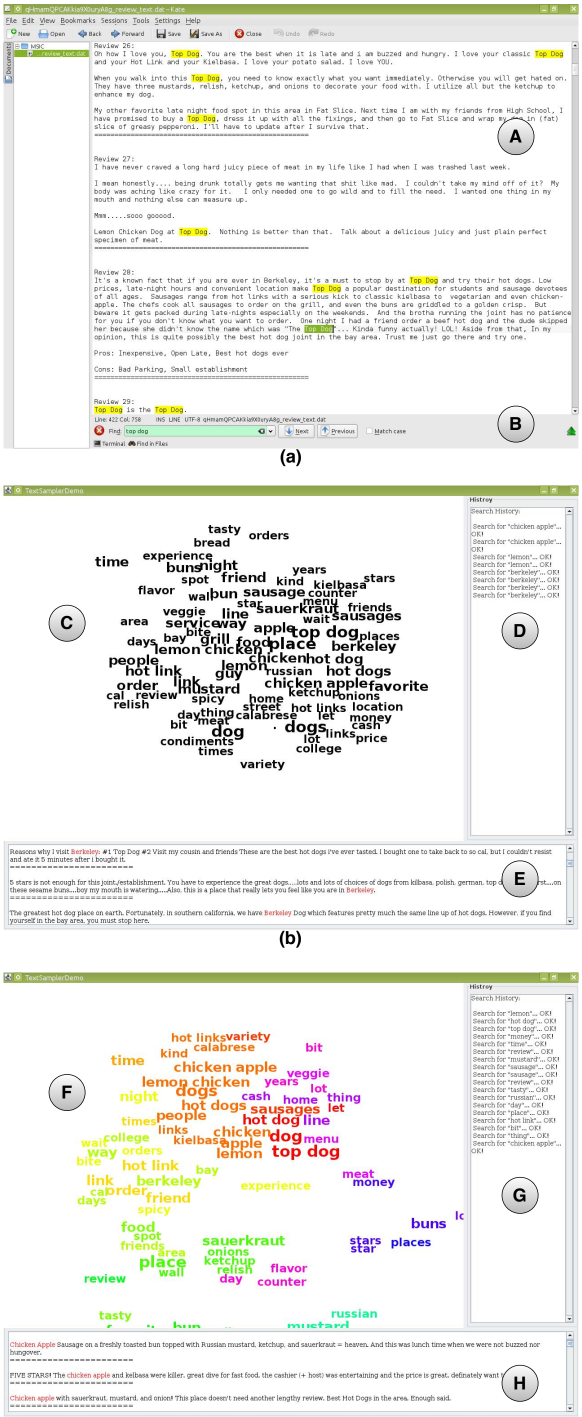
3.3 Grammatical Dependency Parsing

In order to obtain the semantic information, we use NLP tools to compute the *grammatical dependency graph* of the review text for each business, resulting in a graph of key phrases. To construct the graph, the review content for a specific restaurant was first extracted from the raw dataset and chunked into sentences. Then, the sentences were parsed based on grammatical relations and eventually the relationship information was filtered to form a context graph.We used the Stanford Parser [6, 22] and the OpenNLP toolkits [19] to create the context graphs.

First, for each review, we broke down each sentence into typed-dependency parse graph using the Stanford Parser. In Figure 3, (a) shows the typed dependency parse for a sample sentence, and (b) shows the filtered sentence level grammatical relations. We filtered edges that represent unimportant grammatical relations such as *aux*, *auxpass*, *punct*, *det*, *cop*, etc. Because nouns, verbs, and adjectives are most important in our domain (user generated reviews), we retained only terms with those specific part-of-speech tags [6, 22], such as *VB*, *VBD*, *VBG*, *VBN*, *NN*, *NNP*, *NNS*, *JJ*, *JJR*, *JJS*, etc. In this first part of the process, we extracted the main grammatical relations within a sentence.

Then, for each restaurant, we concatenated the parse graphs of all the sentences of all the reviews for that restaurant into a single graph. If relations amongst different sentences shared a term, the shared term was merged as a single shared vertex. The grammatical dependency graphs for each restaurant were usually large due to the large number of reviews for each restaurant in the dataset (e.g. one restaurant had 110 reviews). Thus, we only retained the most important and meaningful types of nodes (nouns, verbs, and adjectives) in the graphs for later processing (e.g. 1500 nodes for the same restaurant).

In our final step, we assigned weight values to both the vertices and the edges for later use in the graph layout phase. The vertex weight W_i of term T_i was computed using the traditional *IDF* value:



Review 26:
Oh how I love you, Top Dog. You are the best when it is late and i am buzzed and hungry. I love your classic Top Dog and your Hot Link and your Kielbasa. I love your potato salad. I love YOU.

When you walk into this Top Dog, you need to know exactly what you want immediately. Otherwise you will get hated on. They have three mustards, relish, ketchup, and onions to decorate your food with. I utilize all but the ketchup to enhance my dog.

My other favorite late night food spot in this area in Fat Slice. Next time I am with my friends from High School, I have promised to buy a Top Dog, dress it up with all the fixings, and then go to Fat Slice and wrap my dog in (fat) slice of greasy pepperoni. I'll have to update after I survive that.
===

Review 27:
I have never craved a long hard juicy piece of meat in my life like I had when I was trashed last week.

I mean honestly.... being drunk totally gets me wanting that shit like mad. I couldn't take my mind off of it? My body was aching like crazy for it. I only needed one to go wild and to fill the need. I wanted one thing in my mouth and nothing else can measure up.

Mmm.....sooo gooood.

Lemon Chicken Dog at Top Dog. Nothing is better than that. Talk about a delicious juicy and just plain perfect specimen of meat.
===

Review 28:
It's a known fact that if you are ever in Berkeley, it's a must to stop by at Top Dog and try their hot dogs. Low prices, late-night hours and convenient location make Top Dog a popular destination for students and sausage devotees of all ages. Sausages range from hot links with a serious kick to classic kielbasa to vegetarian and even chicken-apple. The chefs cook all sausages to order on the grill, and even the buns are griddled to a golden crisp. But beware it gets packed during late-nights especially on the weekends. And the brotha running the joint has no patience for you if you don't know what you want to order. One night I had a friend order a beef hot dog and the dude skipped her because she didn't know the name which was "The Top Dog"... Kinda funny actually! LOL! Aside from that, In my opinion, this is quite possibly the best hot dog joint in the bay area. Trust me just go there and try one.

Pros: Inexpensive, Open Late, Best hot dogs ever

Cons: Bad Parking, Small establishment
===

Review 29:
Top Dog is the Top Dog.

Line: 422 Col: 758 INS LINE UTF-8 qHmamQPCAKkia9X0uryA8g_review_text.dat

Find: top dog ↓ Next ↑ Previous □ Match case

Terminal Find in Files

(a)

Search History:

Search for "chicken apple"... OK!
Search for "chicken apple"... OK!
Search for "lemon"... OK!
Search for "lemon"... OK!
Search for "berkeley"... OK!
Search for "berkeley"... OK!
Search for "berkeley"... OK!
Search for "berkeley"... OK!

Reasons why I visit Berkeley: #1 Top Dog #2 Visit my cousin and friends These are the best hot dogs i've ever tasted. I bought one to take back to so cal, but I couldn't resist and ate it 5 minutes after i bought it.
========================

5 stars is not enough for this joint./establishment. You have to experience the great dogs,,,,,lots and lots of choices of dogs from kilbasa, polish, german, top d rst....on these sesame buns....boy my mouth is watering.....Also, this is a place that really lets you feel like you are in Berkeley.
========================

The greatest hot dog place on earth. Fortunately, in southern california, we have Berkeley Dog which features pretty much the same line up of hot dogs. However, if you find yourself in the bay area, you must stop here.

(b)

Search History:

Search for "lemon"... OK!
Search for "hot dog"... OK!
Search for "top dog"... OK!
Search for "money"... OK!
Search for "time"... OK!
Search for "review"... OK!
Search for "mustard"... OK!
Search for "sausage"... OK!
Search for "sausage"... OK!
Search for "review"... OK!
Search for "tasty"... OK!
Search for "russian"... OK!
Search for "day"... OK!
Search for "place"... OK!
Search for "hot link"... OK!
Search for "bit"... OK!
Search for "thing"... OK!
Search for "chicken apple"... OK!

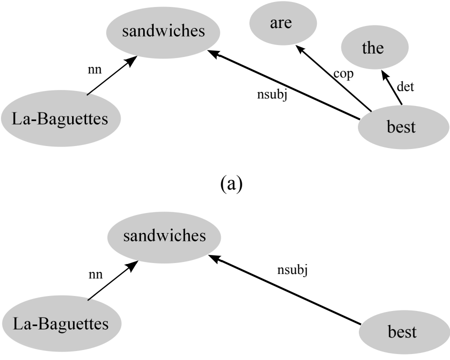
Chicken Apple Sausage on a freshly toasted bun topped with Russian mustard, ketchup, and sauerkraut = heaven. And this was lunch time when we were not buzzed nor hungover.
========================

FIVE STARS! The chicken apple and kelbasa were killer. great dive for fast food. the cashier (+ host) was entertaining and the price is great. definately want t
========================

Chicken apple with sauerkraut, mustard, and onion! This place doesn't need another lengthy review. Best Hot Dogs in the area. Enough said.
========================

(c)

Figure 2: Different review reading techniques for showing the same Yelp review data: (a) Normal Text (NT), (b) Random Word Cloud (RW), and (c) ReCloud (RC). In the NT interface, Component A displays the raw review content and Component B is a search box for finding specific keywords. In the user interface of both RW and RC, Component C and F are the word cloud visualizations; Component D and G shows the historical keywords clicked by users; and Component E and H displays the review contents that match the currently clicked keyword highlighted in red.



(a)

(b)

Figure 3: (a) A collapsed typed dependency parse for the sentence La-Baguettes sandwiches are the best!. (b) The sentence level graph filtered from the parse in (a).

$$W_i(T_i) = \log(N/df_i)\cdot(\log df_i + 1) \tag{1}$$

N is the number of sentences, and df_i is the document frequency which denotes the number of sentences that have this term. For weighting the edges, the same strategy as vertex weighting was used, the only difference being that the variable df_j means the number of sentences that have this edge type.

$$W_j(E_j) = \log(N/df_j)\cdot(\log df_j + 1) \tag{2}$$

3.4 Force-Directed Graph Layout with Energy Model

To create an initial two-dimensional clustered representation of the grammatical dependency graph, we applied a force-directed graph layout. Force-directed graph layout is widely used in drawing large graphs in an aesthetically pleasing way. The layout algorithm assigns ”attraction forces” or ”repulsive forces” between graph nodes based on edges and iteratively simulates such physical behaviors for drawing the graph. Studies have indicated that the force-directed graph layout of text semantic graph can produce easily understood representations [4, 20]. However, the force-directed graph layout has a scalability problem; the layout computation process is very time-consuming when many layout iterations are needed due to large quantities of edges and nodes.

When the graph is large, an energy model based method of performing the force-directed graph layout is better, which directly influences the layout quality and speed [18]. The essential idea of the energy model is to map the layout to an energy function, and the value of such energy is related to the optimal goal of the whole layout. Then, the algorithm iteratively searches all the possible solutions having the lowest energy of the entire layout. A good layout is considered to have the minimal energy [17]. In this paper, we used the LinLogLayout toolkit [13] as the energy model, which is fast enough to generate word clouds on-the-fly during user interaction. The resulting graph layout serves as the initial placement of words for the next phase. LinLogLayout also provides cluster label information, which we utilize in ReCloud for coloring.

3.5 Word Cloud Rendering

Based on the initial word placements output by the above force directed layout algorithm, we then performed a similar approach as [8] to generate our final ReCloud visualization. The largest difference is that the initial positions of word tags in ReCloud are not random, as opposed to traditional word clouds. We implemented this word cloud rendering process using the Java 2D Graphics library. More specifically, the fine-grain adjustment and rendering process can be described in the following steps:

1. Sort the vertex list of the grammatical dependency graph according to vertex weight (Eq. (1)) to generate the word list for rendering. High frequency (and hence large size) words are rendered first.
2. For each word in the sorted rendering list, the initial position and color is defined by the force-directed layout algorithm, and its font size is determined by the vertex weight. Unique colors are assigned to each cluster label on a simple hue scale.
3. Collision detection is performed to see whether the word spatially overlaps with previously rendered words. We use a double buffer mask for the test. We render previous words in one image buffer, render the new word in the second image buffer, and then conduct a logical AND of the bits in these two buffers to quickly check for a collision.
4. If a collision occurs, we place the word by following the Archimedean spiral [8] around the words initial position (from step 2) until there is no collision.
5. Step 2-4 are repeated until all the words are rendered, or until a predefined maximum word threshold is reached.

4 Evaluation

The major goal of our user study was to assess the effectiveness of the ReCloud concept in decision-making tasks. The primary research question is how the grammatically semantic layout affects users' performance and satisfaction in comparison to traditional random layout and normal text reading. We chose a common daily task as our study scenario: finding good restaurants and judging restaurants based on customer review text. We specifically focus on the text content of the reviews, not the quantitative review scores, to emphasize the role of the word cloud in comprehension. These kinds of tasks are familiar to users who struggle in making informed decisions about restaurants.

4.1 Participants and Apparatus

We recruited 15 participants (7 females), aged 20 to 32 (24.4 on average), for our study. All participants were familiar with normal word clouds such as Wordle. They were all native English speakers and undergraduate (4) or graduate students (11) from our university. The study lasted about 50 minutes and each participant was compensated with $30 cash.

All the tasks were performed on a laptop with Intel Centrino 2.10GHz CPU and 4 GB RAM with Kubuntu 12.04, with an external keyboard and mouse. The display used was one 19-inch LCD monitor with 1280×1024 resolution. The entire display was used to show reviews of one given restaurant. When two restaurants were compared, users could swap freely between the reviews of the two restaurants using a keyboard shortcut (Figure 2(a)). Task completion times were measured using a stopwatch and participants' mouse cursor movement data was collected by our system.

4.2 Review Reading Techniques

During this user study, we compared three conditions: ReCloud, Random Word Cloud, and Normal Text. For the two alternative techniques, Random Word Cloud was used as a baseline for comparison because it does not embed semantics in the layout, and Normal Text was used because this is commonly how users read reviews. We also removed the review scores to let participants focus on the review content itself.

Normal Text (NT). We listed the textual content of all customer reviews for a given restaurant in a normal scrolling text editor. The quantitative ratings for each review were omitted. Users could use the search box in the text editor to find and highlight keywords in the reviews.

Random Word Cloud (RW). We used a random layout method according to the algorithm described in [8] to generate this word cloud, as shown in Figure 2(b). The user interactions were the same as described in Section 3.1.

ReCloud (RC): In this technique, participants used the aforementioned ReCloud system to read reviews using the semantic layout word cloud (Figure 2(c)).

4.3 Tasks and Design

Two types of tasks were used to assess users performance: *decision making tasks* and *feature finding tasks*. Both tasks attempt to mirror events that regularly occur in daily life. As shown in Table 1, for each of the review reading conditions, we allowed participants to perform two types of tasks as below:

Decision Making Task. The goal of this overview-oriented task is to efficiently and correctly compare and distinguish similar types of restaurants of varying quality based on review content. In this task, participants must decide between a given pair of restaurants to patronize based on the reviews. There were two sub-tasks based on restaurant quality. For the ”good-good” sub-task, users compared two restaurants of good quality, meaning that both restaurants in these pairs had high ratings (4 or 5 stars on the business profiles of the Yelp dataset). For the ”good-bad” sub-task, users compared two restaurants of opposite quality, meaning that the two restaurants had significant differences in their ratings (good was 4-5 stars, and bad was 1-2 stars).

The ”good-good” pair serves as a difficult task. The ”good-bad” pair has a correct answer, in that we assume participants would want to choose restaurants that other people highly rated quantitatively, but should be able to identify the difference in quality from the review text only. Since we carefully chose the restaurant pairs based on matching cuisine, we expect the restaurant quality to be the deciding factor, rather than menu preferences. All paired restaurants had similar price levels, locations, and cuisines. Thus we did not employ a randomized pairing process, but instead carefully chose the restaurant pairs from the Yelp Academic Dataset based on these criteria. In this study, participants were not familiar with any of the restaurants and were unaware of how the good and bad restaurants were chosen.

We selected 6 pairs of restaurants for this task, three ”good-good” and three ”good-bad” pairs. Each participant used all three review reading conditions. Conditions were counterbalanced in a latin-square design. For each condition they performed one ”good-good” and one ”good-bad” pair. Thus, in total, each participant performed all 6 pairs.

Feature Finding Task. The goal of this detail-oriented task is to efficiently identify basic non-quantitative features of a given restaurant based on review content. In this task, participants typed a list of relevant features of the restaurant based on its reviews. This task was designed to represent the process of understanding qualitative features that would be considered in deciding to patronize a particular restaurant, such as flavor, value, service, atmosphere, etc. Thus, we defined two sub-tasks: finding food features, and finding non-food features.

We selected 6 restaurants, all had high ratings (4 or 5 stars). As shown in Table 1, restaurants 1, 3, and 5 were used for non-food feature finding sub-task. Restaurants 2, 4 and 6 were used for food feature finding sub-task. The restaurants varied in number of reviews available, with 49, 65, 66, 185, 283 and 2232 reviews in restaurant 1-6 respectively. Each participant performed all three review reading conditions. Conditions were counterbalanced in a latin-square design. For each condition, the participants performed one food and one non-food feature finding sub-task. Moreover, we imposed a two-minute time limit on half of the participants, while the other half did not have any time limit. This was to investigate whether the semantic layout would be particularly helpful when users have time constraints.

Task Type	Technique	Data	Task Question
Decision Making Task	NT	Good-Good Pair 1	Which Restaurant will you go?
	NT	Good-Bad Pair 1	
	RW	Good-Good Pair 2	
	RW	Good-Bad Pair 2	
	RC	Good-Good Pair 3	
	RC	Good-Bad Pair 3	
Feature Finding Task	NT	Restaurant 1	Non-food Feature
	NT	Restaurant 2	Food Feature
	RW	Restaurant 3	Non-food Feature
	RW	Restaurant 4	Food Feature
	RC	Restaurant 5	Non-food Feature
	RC	Restaurant 6	Food Feature

Table 1: Tasks Design and Study Procedure. The three technique conditions were counterbalanced in a latin-square design.

4.4 Procedure

Before the study started, participants had time to get familiar with the three different review reading techniques using the same sample dataset. Then participants were instructed to perform the two task sets using each of the three review reading conditions according to their latin-square assignment.

For the decision making task, we measured the completion times (for all) and error rates (for good-bad pairs) in each trial. For the feature finding task, completion time was only recorded for trials of participants who were in the no time limit group. After each review reading condition in both tasks, participants completed a Likert-style questionnaire based on NASA TLX [9] to collect their ratings of mental demand, physical demand, and other metrics to measure task difficulty levels. After each of the two tasks, participants were asked to provide a ranking of preferences among the three review reading conditions, with 1 being most preferred and 3 being least preferred. At the end of the study, we conducted a short informal interview to gather general comments for each review reading condition.

5 Results

5.1 Decision Making Task Results

5.1.1 Task Completion Time and Error Rate

We ran six repeated measure ANOVAs on task completion time for each review reading condition and restaurant pair sub-tasks in the decision making task. Results indicated that restaurant pairing (”good-good” vs.”good-bad”) had a significant effect ($F_{1,14}$=52.465, p <0.001) on task completion time, with ”good-bad” being significantly faster. But there was no significant effect of review reading condition on task completion time.

A post-hoc one-way ANOVA was run for ”good-good” restaurant pairs and ”good-bad” restaurant pairs. The result showed that review reading technique had a significant effect on task completion time ($F_{2,42}$=3.157, p=0.05) in ”good-good” restaurant pairs, but no significant effect in ”good-bad” restaurant pairs ($F_{2,42}$=0.253, p=0.78). From Figure 4, we can see that participants spent less time in the decision making tasks using ReCloud compared to the other two conditions for ”good-good” restaurant pairs.

In this decision making task, the error rates were calculated in good-bad restaurant pairs and we assumed that the correct answer was the good one. The error rates of *RW* and *RC* were the same (6.7%, 1 error out of 15 trials). There were no errors in *NT*.

5.1.2 Mouse Events Results

We recorded all mouse events when users were presented with techniques *RW* and *RC*. The metrics with which we evaluated the mouse events were: the number of word tags hovered over (for at least 0.1 sec) by the cursor, and the number of word tags clicked by the participant. The purpose of the former metric was to estimate a user’s amount of visual navigation, since quantity of mouse hovering likely relates to quantity of items attended to by the user. For example, Chen *et al.* [3] found a correlation between users’ eye movement and their mouse cursor movement in web browsing. The latter metric could reflect how many reviews users need to read in details. We discuss the mouse movement issues further in Section 6.2.

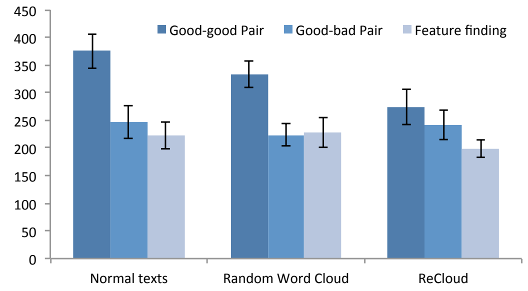


Figure 4: Completion time (seconds) for decision making task (good-good and good-bad restaurant pairs) and feature finding task.

We first compared the average number of clicked words for all 9 good restaurants (mean = 3.78, std = 0.49) and 3 bad restaurants (mean = 5.74, std = 0.41) in techniques *RW* and *RC* via one-way ANOVA. We found that the 3 bad restaurants had significantly fewer word tags clicked than the 9 good restaurants ($F_{1,22}$=5.77, p=0.02). From this, we may conclude that bad restaurants were easy to distinguish by looking at the word cloud without needing to read many reviews.

Then, we analyzed the mouse events from good-good restaurant pairs (3 pairs, 6 restaurants). Figure 5 shows the average number of hovered words (5a) and average number of clicked words (5b) of each restaurant. In Figure 5(a), *RC* had fewer (by at least a standard deviation) words hovered than *RW* for 4 of the pairs. But the average number of clicked words (Figure 5(b)) was similar in both word cloud techniques. Thus, *RC* required fewer mouse hovers than *RW* in order to accomplish similar levels of mouse clicks. Therefore, it is possible that this represents that *RC* users require less visual navigation to find useful search targets, perhaps due to the better semantic layout.

5.1.3 Preference Ranking

The participants preference ranking of techniques was analyzed with a Friedman test to evaluate differences in median rank across three techniques. There was a significant difference between the three based on the preferential ranking (χ^2(2, N=15)=6.533, p=0.04). The follow-up pairwise Wilcoxon tests found that *NT* had significantly less preferred ranking than *RC* (p=0.05) and *RW* (p=0.04). There is no significant difference in preference ordering between *RC* and *RW*. The results are shown in Figure 6(a).

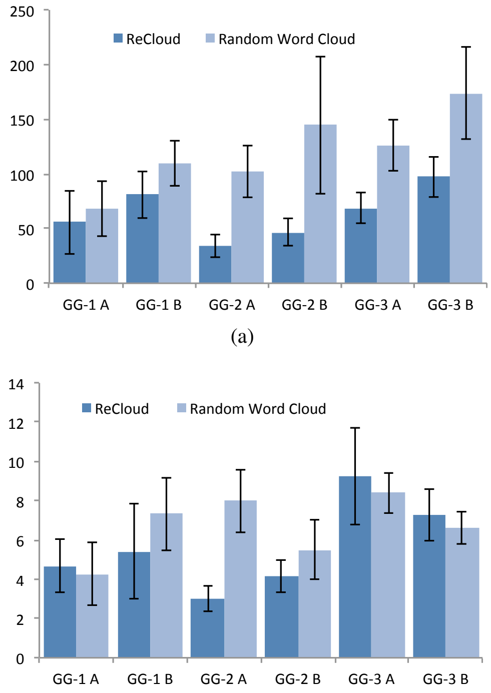
5.2 Feature Finding Task Results

5.2.1 Task Completion Time

We ran a one-way ANOVA for task completion time for those participants who did not have a time limit. We did not find a significant effect on task completion time between reading conditions ($F_{2,42}$=0.253, p=0.78), as shown in Figure 4. The reason might be that all restaurants in this task had different scales of review counts (see Section 4.3).

5.2.2 Mouse Event Results

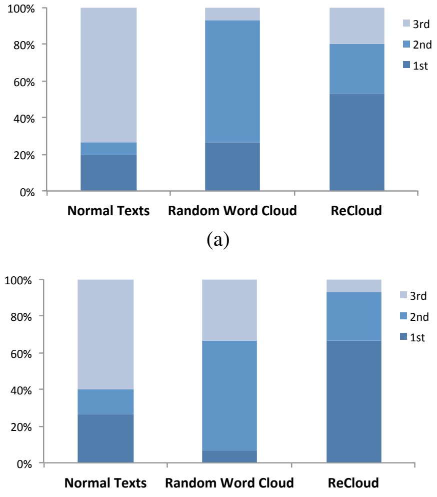
Figure 7 shows the average number of hovered words and average number of clicked words for each restaurant. The average number of clicked words was approximately 5 (for both *RC* and *RW*) in all 6 restaurants, even though the restaurants had vastly different number of reviews to read (ranging from 49 to 2,232). However, *RC* had



(a)

(b)

Figure 5: Mouse events data for decision making tasks (good-good pair): (a) word tag hovering data (Y-axis is average hovered word number of each restaurant) and (b) word tag clicking data (Y-axis is average clicked word number of each restaurant).



(a)

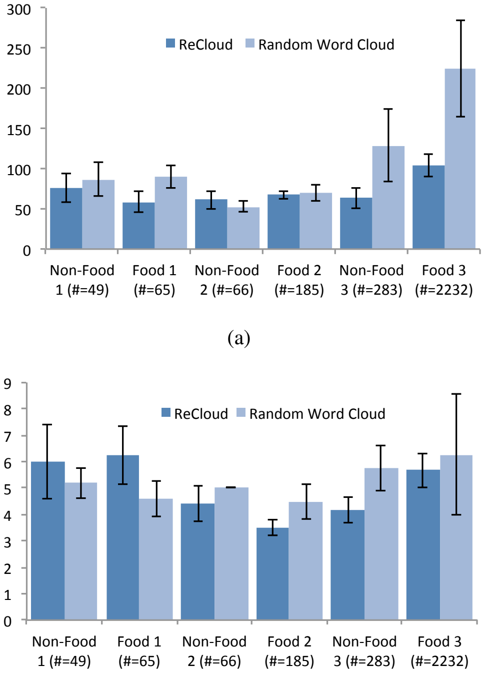
(b)

Figure 6: User preference ranking results in decision making tasks (a) and feature finding tasks (b).

significantly fewer hovered words than *RW* for the restaurants with large number of reviews (Non-food 3 had 283 reviews and Food 3 had 2,232 reviews). This potentially indicates that the semantic layout of *RC* enabled users to effectively navigate word clouds of a large number of reviews.

5.2.3 Preference Ranking

The user preference rankings of the three techniques in this task also had a significant difference across three techniques in a Friedman test (χ^2(2, N=15)=8.133, p=0.02). Follow-up pairwise Wilcoxon tests found that *RC* had a significantly more preferred than *NT* (p=0.02) and *RW* (p=0.005), shown in Figure 6(b).



(a)

(b)

Figure 7: Mouse events data for feature finding tasks: (a) word tag hovering data (Y-axis is average hovered word number of each restaurant) and (b) word tag clicking data (Y-axis is average clicked word number of each restaurant).

5.2.4 User Satisfaction Levels

The Likert-style questionnaire based on NASA TLX was used to acquire user feedback of the three review reading techniques from participants who had a two-minute time limit imposed on them during the feature finding task. As shown in Figure 8(b), a Friedman test was conducted to observe any differences in scores in the questionnaire. The test results showed that there were significant differences on mental demand (χ^2(2, N=8)=11.826, p=0.003), physical demand (χ^2(2, N=8)=6.5, p=0.04), temporal demand (χ^2(2, N=8)=10.129, p=0.006) and effort (χ^2(2, N=8)=7.00, p=0.03).

The follow-up pairwise Wilcoxon tests showed: for mental demand, *RC* is significantly lower than *NT* (p=0.02) and *RW* (p=0.03); for physical demand, *RC* is significantly lower than *NT* (p=0.04); for temporal demand, *RC* is significantly lower than *NT* (p=0.01); for effort, *NT* is significantly higher than *RW* (p=0.04) and *RC* (p=0.02).

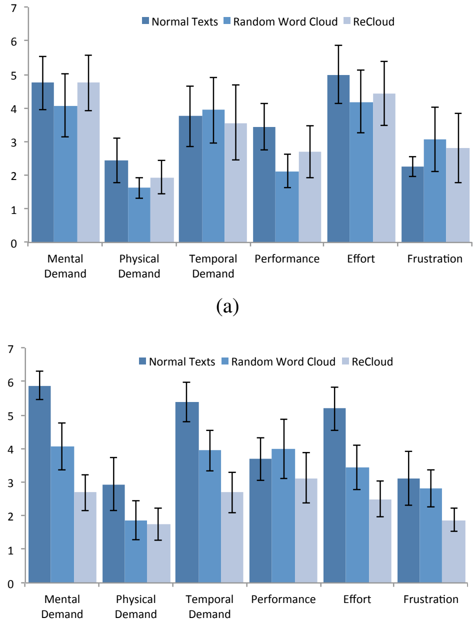
In the case of the no time limit feature finding task, the Friedman test results showed there was no significant difference in any of the responses, shown in Figure 8(a). Therefore, the above results indicated that *RC* had better user experience and user satisfaction in time-constrained tasks. This might relate to the previous argument about efficient visual navigation for *RC*.

5.3 Qualitative Feedback

5.3.1 Semantic Information Retrieval

Based on feedback, we found the semantic layout provided by ReCloud helped people navigate and find relevant information.

"It [ReCloud] makes it much easier to look for keywords that help when deciding on which option to pick. It's well organized and groups similar words and distinguishes them by color. The black and white words [Normal Word Cloud] with no grouping make it difficult to find tags."(Subject 15)



(a)

(b)

Figure 8: TLX-based Liker-style questionnaire results for feature finding tasks (where lower is better): (a) without time limit and (b) with two-minute time limit.

"...It [ReCloud] is an easy way to navigate through several reviews that use similar terminology to pinpoint specific aspects of a restaurant. It is easier to find out what I'm looking for." (Subject 14)

"Finding the keywords for 'service' or 'sandwich' was made easiest with ReCloud. It was easy to pick out the keywords that I needed to look at to make my decisions."(Subject 9)

The visual aspects of font size and color had positive impact on users' review reading process by ReCloud as well.

"The size of the words also made it easier to know what was important/more used in the reviews."(Subject 7)

5.3.2 Keywords Query by Interaction

The clickable interaction to query keywords in review content was found useful in ReCloud and Random Word Cloud.

"From these three ReCloud saves my time since I can click on the things I am interested in and quickly see them highlighted in the reviews."(Subject 2)

"The ReCloud was better because ... a good first move was to click the largest word which would give you a pretty good overview of what the place and then browse the other tags in case anything of particular interest or disinterest was there."(Subject 12)

"I still liked ReCloud over all the other techniques because it helped me find better keyword to search so I could read more details in the actual reviews."(Subject 9)

5.3.3 Natural Language Processing

There were mixed reviews on factors about our NLP techniques. A few of our participants did not feel that all the necessary word tags were presented in the word cloud.

"I would have ranked the tag clouds higher, but I was unable to finish one of the tasks because there were no tags regarding the quality of service at a restaurant. Normally, I liked ReCloud more, but I got the impression that fewer tags were included. I liked the clustering, but sometimes couldn't tell why terms were included in specific clusters."(Subject 1)

One participant wanted to assess the personality of individual reviewers based on completed reviews. But NLP based ReCloud did not provide specific information about individual review writers' personalities and interests.

"I preferred reading full reviews because I felt like I could better understand the personality and interests of the reviewers, which factors a great deal into the way I interpret the quality and reliability of the review." (Subject 8)

6 Discussion

6.1 Difficulty of Decision Making Tasks

In the decision making task, we found that ReCloud had a significantly faster task completion time than the random word cloud in good-good pairs of restaurants. But there was no significant difference in good-bad pairs. We believe that these results can be explained in the following two ways:

First, good-bad restaurant pairs are easy to distinguish in all three techniques. They have lower task completion times in Figure 4. At the same time, our mouse event records also support this fact. In bad restaurants, the average number of clicked words is significantly less than that in good restaurants (see Section 5.3.2).

Second, good-good restaurant pairs are difficult to distinguish. All of them have similar high ratings, so users need more context information to support their decisions. In other words, users need to spend more time to find evidence in good-good restaurant pairs. The mouse events in Figure 5 showed that users hover less in ReCloud. Combined with users' feedback described in Section 5.3.1, we can see that the semantic layout improved the visual search process in ReCloud. Moreover, participants significantly preferred ReCloud and Random Word Cloud over Normal Text.

The error rates of the two word clouds were the same, 6.7%, and users preferred ReCloud over the random word cloud. Thus, the content discrimination and bias in NLP techniques and word cloud visualizations did not have significant negative influence on the error rates in the word clouds.

Therefore, ReCloud with semantic layout offered improved user performance in both time and mouse events, and was preferred by users, especially in difficult decision making tasks when comparing similar quality businesses.

6.2 Review Scales and Time-Constrained Situations

In the feature finding tasks, we used restaurants with different numbers of reviews (see Section 4.3), and found that ReCloud had fewer mouse hovers in cases with large number of reviews. In Figure 7(a), ReCloud in No-Food 3 (283 reviews) and Food 3 (2,232 reviews) had significantly fewer hovered words than Random Word Cloud, for similar numbers of clicked words (Figure 7(b)). That might be because the nature of the feature finding task was to perform the categorization and clustering process in people's minds. Thus, we believe that the semantic information of ReCloud helped users perform this process easier.

In the time-limited feature finding tasks, we found that users' workload ratings of ReCloud were significantly higher than that of Random Word Cloud in terms of mental demand, physical demand, temporal demand and effort. However, there was no significant difference in feature finding tasks without time limit. Furthermore, participants significantly preferred ReCloud to Normal Texts (p=0.024) and Random Word Cloud (p=0.005).

As shown above, participants hovered over few tags, yet clicked on a similar number of tags (in some of the tasks, see Figure 5(b) and Figure 7(b)), when they used ReCloud. We hypothesize that it is because the ordered layout enabled users to more easily identify tags of interest, at both the perceptual and cognitive levels. Hovering in RW might indicate a more challenging visual search process and/or greater cognitive load in considering each tag as indicated by the TLX scores. In summary, ReCloud had fewer hovered word tags and better user satisfaction in a large mount of reviews and time-constrained tasks.

6.3 NLP Technique

Our ReCloud visualization is highly dependent on the results of the NLP technique applied. Currently, ReCloud uses grammatical dependency parsing for extraction of semantics from user reviews and the resulting dependency graph to govern the layout. Although we received mixed qualitative feedback from participants on the NLP results (see Section 5.3.3), the actual statistical analysis results indicated that the overall error rates in decision making tasks were very low (see Section 5.1.1). So we believe that further improvements of the NLP algorithms can enhance our ReCloud visualization, for example, more necessary word tags would be shown, word clustering and its font size would be more accurate, and personality context information of review writers would be available.

6.4 Word Color Encoding

In ReCloud design, we used colors to represent word tags in different semantic clusters generated by the LinlogLayout force directed algorithm [18]. The goal of this color-coding was to keep semantics clusters persistent in ReCloud. Sometimes, the final word cloud rendering algorithm might jeopardize the original semantic layout suggested by NLP techniques. For example, the initial positions of some keywords in the clustered layout might overlap. Each word tag has its own font size according to its frequency in reviews (see Section 3.5). In order to avoid the collisions among other placed word tags, the process of finding new placements of the word tags might locally modify the initial layout a small amount. Finally, the word cloud might not correctly present semantics in some local areas. In this situation, the color encoding of semantic clusters can help users better understand the semantic information by visually preserving the clustering in ReCloud.

7 Conclusion and Future Work

We have presented a novel visualization technique, ReCloud, based on the use of natural language processing techniques to extract a grammatical dependency graph from the raw content of user reviews. An energy based force directed graph layout algorithm was applied to the grammatical dependency graph that reflects the review semantics to create an initial layout of the keywords. Based on this initial layout, we generated a new word cloud visualization that embeds the semantic information. ReCloud also supports basic user interactions for accessing the review text, such as searching by clicking a specific word tag. We also conducted a user study to evaluate how ReCloud helps users in tasks that involve choosing and judging restaurants based on review content. We used the Yelp Academic Dataset as our testbed and designed two types of tasks in the study: decision making tasks and feature finding tasks. The results indicate that ReCloud improves user performance time in difficult decision-making, reduces unnecessary mouse hover actions, provides greater user preference, and decreased perceived workload, and produced positive user comments about the semantic layout. We believe these results indicate the value of the semantic layout in better representing context of a large amount of review text.

In the future, we plan to append more information on the clustered layout word cloud, like time-series restaurant reviews and sentiment analysis of review information. We will also apply a more sophisticated NLP technique for processing the review content data as well as enable a search box functionality for finding words easier within the word cloud. Furthermore, by manipulating the NLP algorithm, we will also try to expose keywords that previously didn't appear in the clustered layout word cloud and therefore provide a more customizable and possibly interactive review reading experience for the users.

Acknowledgements

We would like to thank all the users who participated in our study, Yelp Inc. for the award support, and the reviewers for their valuable suggestions.

References

[1] S. Bateman, C. Gutwin, and M. Nacenta. Seeing things in the clouds: the effect of visual features on tag cloud selections. HT '08, pages 193–202, 2008.

[2] G. Carenini and L. Rizoli. A Multimedia Interface for Facilitating Comparisons of Opinions. In *IUI '09*, pages 325–334, 2009.

[3] M. C. Chen, J. R. Anderson, and M. H. Sohn. What can a mouse cursor tell us more? In *CHI '01*, 2001.

[4] W. Cui, Y. Wu, S. Liu, F. Wei, M. Zhou, and H. Qu. Context-Preserving, Dynamic Word Cloud Visualization. *IEEE CG&A*, 30(6):42–53, 2010.

[5] A. Culotta and J. Sorensen. Dependency tree kernels for relation extraction. In *ACL '04*, 2004.

[6] C. M. de Marneffe, Marie-Catherine, Bill MacCartney. Generating typed dependency parses from phrase structure parses. In *Proc. of LREC '06*, pages 449–454, 2006.

[7] Economist Opinion Cloud. http://infomous.com/site/economist/.

[8] J. Feinberg. Wordle. In *Beautiful Visualization*, chapter 3. 2009.

[9] S. Hart and L. Staveland. Development of NASA-TLX: Results of empirical and theoretical research. In *Human mental workload*. 1988.

[10] J. Huang, O. Etzioni, L. Zettlemoyer, K. Clark, and C. Lee. RevMiner: An Extractive Interface for. Navigating Reviews on a Smartphone. In *UIST '12*, 2012.

[11] O. Kaser and D. Lemire. Tag-Cloud Drawing: Algorithms for Cloud Visualization, 2007.

[12] K. Koh, B. Lee, B. Kim, and J. Seo. ManiWordle: Providing Flexible Control over Wordle. *IEEE TVCG*, 16(6):1190–1197, Nov. 2010.

[13] LinLogLayout. http://code.google.com/p/linloglayout/.

[14] B. Liu and S. M. Street. Opinion Observer : Analyzing and Comparing Opinions on the Web. In *WWW '05*, 2005.

[15] S. Liu, M. X. Zhou, S. Pan, Y. Song, W. Qian, W. Cai, and X. Lian. TIARA : Interactive , Topic-Based Visual Text Summarization. *ACM TIST*, 3(2), 2012.

[16] S. Lohmann, J. Ziegler, and L. Tetzlaff. Comparison of Tag Cloud Layouts: Task-Related Performance and Visual Exploration. volume 5726 of *LNCS*, chapter 43, pages 392–404. 2009.

[17] A. Noack. Energy Models for Graph Clustering. *Journal of Graph Algorithms and Applications*, 11(2):453–480, 2007.

[18] A. Noack. Modularity clustering is force-directed layout. *Physical Review E*, 79(2), Feb. 2009.

[19] OpenNLP. http://opennlp.apache.org/.

[20] F. Paulovich, F. Toledo, and G. Telles. Semantic Wordification of Document Collections. *Computer Graphics*, 31:1145–1153, June 2012.

[21] A. W. Rivadeneira, D. M. Gruen, M. J. Muller, and D. R. Millen. Getting our head in the clouds: toward evaluation studies of tagclouds. CHI '07, pages 995–998, 2007.

[22] Stanford Parser. http://nlp.stanford.edu/software.

[23] F. B. Viégas, M. Wattenberg, and J. Feinberg. Participatory visualization with Wordle. *IEEE TVCG*, 15(6):1137–44, 2009.

[24] M. Wang, N. Smith, and T. Mitamura. What is the Jeopardy model? A quasi-synchronous grammar for QA. In *Proc. of EMNLP-CoNLL*, 2007.

[25] Y. Wu, F. Wei, S. Liu, N. Au, W. Cui, H. Zhou, and H. Qu. OpinionSeer: interactive visualization of hotel customer feedback. *IEEE TVCG*, 16(6):1109–18, 2010.

[26] 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 *CHI '11*, page 1541, 2011.

[27] Yelp Academic Dataset. http://www.yelp.com/academic_dataset.