Harvesting and Summarizing User-Generated Content for Advanced Speech-Based HCI

Jingjing Liu, Stephanie Seneff, and Victor Zue, *Member, IEEE*

***Abstract*—There are many Web-based platforms where people could share user-generated content such as reviews, posts, blogs, and tweets. However, online communities and social networks are expanding so rapidly that it is impossible for people to digest all the information. To help users obtain information more efficiently, both the interface for data access and the information representation need to be improved. An intuitive and personalized interface, such as a dialogue system, could be an ideal assistant, which engages a user in a continuous dialogue to garner the user's interest, assists the user via speech-navigated interactions, harvests and summarizes the Web data as well as presenting it in a natural way. This work, therefore, aims to conduct research on a universal framework for developing a speech-based interface that can aggregate user-generated content and present the summarized information via speech-based human-computer interactions. The challenge is two-fold. Firstly, how to interpret the semantics and sentiment of user-generated data and aggregate them into structured yet concise summaries? Secondly, how to develop a dialogue modeling mechanism to present the highlighted information via natural language? This work explores plausible approaches to tackling these challenges. We will investigate a parse-and-paraphrase paradigm and a sentiment scoring mechanism for information extraction from unstructured user-generated content. We will also explore sentiment-involved opinion summarization and dialogue modeling approaches for aggregated information representation. A restaurant-domain prototype system has been implemented for demonstration.**

***Index Terms*—Spoken dialogue systems, user-generated content processing.**

I. Introduction

THE Web has been exploding dramatically over the past decade, especially with user-generated-content (UGC). Social networks and community-contributed sites have become pervasive in people's daily life, such as wikis (e.g., Wikipedia), review sites (e.g., Yelp, TripAdvisor), video/photo sharing platforms (e.g., YouTube, Flicker), social networks (e.g., Facebook) and blogs (e.g., Twitter).

Manuscript received July 20, 2012; revised October 15, 2012; accepted November 11, 2012. Date of publication November 21, 2012; date of current version January 03, 2013. This work was supported by Quanta Computers, Inc. through the Qmulus project. The associate editor coordinating the review of this manuscript and approving it for publication was Mr. Roberto Pieraccini.

The authors are with the Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: jingl@csail.mit.edu; seneff@csail.mit.edu; zue@csail.mit.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JSTSP.2012.2229690

At the same time, there is a rapidly increasing usage of mobile devices such as smart phones and tablets along with the rapid development of application software (Apps). For example, as of June 2012, there are over 1,250,000 Apps available on various "App Stores[1]". More and more people rely on mobile devices to access the Web, especially for updating social networks and visiting online communities.

Helpful as these mobile applications are, the data available on the Web are growing exponentially and it is impossible for people to digest all the information even with instant Web access. To help users obtain information more efficiently, both the information representation and the interface for content access need to be improved. Text-formed representation is not efficient enough because of the limited screen real estate. The current search paradigm of typing in a search string and obtaining hundreds of relevant hits is also primitive when compared to how humans collaborate to gather information. For example, in a restaurant search iPhone App, dozens of restaurants could show up on the screen, along with many customer reviews. Given that there are hundreds of thousands reviews, it is a rather time-consuming task, not to mention the issue of reading them on the small screen.

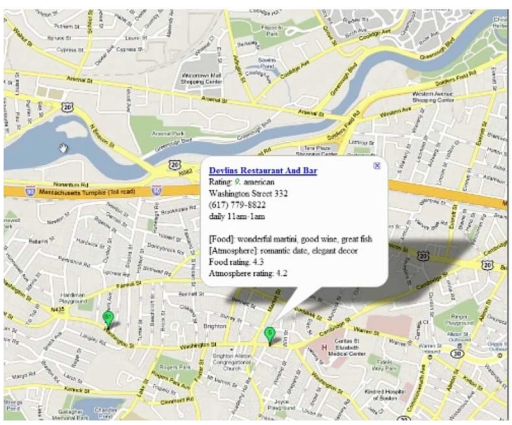
Therefore, there is a great demand for a condensed information representation, i.e., information aggregation or summarization. It would be ideal if one can have a virtual assistant that can summarize the information on the Web in a concise manner, and present it to the user in a natural way, such as a direct spoken dialogue.

Thus, an intuitive and personalized interface in a more timely informative way could be a human-like assistant system, which can engage a user in a continuous spoken dialogue to learn the user's interests and capture the user's intent proactively, subsequently providing the relevant information in a summarized representation in both spoken and textual forms. For example, Fig. 1(a) shows an example conversation between a user and a prototype spoken assistant system. The system not only understood the user's questions and found the target restaurants accordingly, but also summarized multiple retrieved entries in a natural sentence. The system also summarized the reviews on each restaurant automatically and made recommendations based on the user's interest precisely. Fig. 1(b) shows a possible graphical interface of the prototype system, locating the target restaurants on a map and showing the condensed review summary as well as the detailed information on each entry.

[1]Approximately 650,000 on Apple App store, 500,000 on Android Market, and 100,000 on Windows Phone Marketplace.

U: Are there any American restaurants in Brighton?
S: There are 61 American restaurants in Brighton. Many of the rated restaurants have good or excellent ambiance. Most are on Washington Street, Commonwealth Avenue and Market Street.
U: Show me the ones that have nice ambiance.
S: I found 4 entries. Devlin's Restaurant and Bar is perfect for romantic date, Stockyard Restaurant has friendly atmosphere, Tasca Tapas Restaurant has excellent ambiance, and Wings Express has friendly vibes.
U: Which one has good martinis?
S: Devlin's Restaurant and Bar serves American food. It has wonderful martinis, good wine, great fish. It is perfect for romantic date. Great place.

(a)



(b)

Fig. 1. An example conversation between a user and a dialogue system, with a graphical interface locating the target restaurant on the map.

This work, therefore, aims to conduct research on a universal framework for developing such conversational systems that can harvest user-generated content and present the summarized information with natural dialogue interaction [14]. The goal is to investigate a platform that marries UGC harvesting and dialogue system development in an effective and portable way. A platform supporting multimodal interfaces for efficient user-generated data access could promisingly benefit human computer interaction (HCI) experience, as well as potentially advance the technology frontier in industries of consumer electronics and mobile applications.

II. Problem Formulation

User-generated content (e.g., public reviews/posts on forums, blogs, and tweets) provide an extensive collection of free-style comments published by general users, which in return provide grassroots-contributions to users interested in a particular topic or service as assistance. But, valuable as they are, user-generated contents are unstructured and contain very noisy data, as they were freely edited by general users; not to mention that there are hundreds of thousands of community-edited documents available on the Web. Therefore, to filter out context-irrelevant information and to present these unstructured data in a concise dialogue, a summarization mechanism is needed to extract the essence from the large number of reviews/posts/tweets and aggregate them into a condensed yet informative summary.

Summarization and opinion mining from user-generated content have been well studied for years, with many interesting derived topics [1], [2], [4], [6], [9], [10], [13], [20], [21], [26], [27], [28], [31], [32]. Summarization techniques, when applied to spoken dialogue systems, however, are much more complicated than those in pure-text systems. In a text-based system, users can browse through multiple reviews and obtain information very quickly by scanning the text. In contrast, when interacting with spoken dialogue systems, the information space (i.e., the number of words) in a dialogue turn is often very limited. As speech is inherently serial and cannot be skipped and scanned easily, the information feedback from the system is only a couple of utterances spoken by the system. A dialogue system which speaks long diatribes in each single conversation turn would likely not be well received. Thus, the generally used review summarization techniques, although very effective in text-based systems, are not quite suitable for interactive dialogue systems. The missing piece is a dialogue oriented, fine-grained, informative yet condensed summarization mechanism.

Spoken dialogue systems are presently available both in laboratories and commercially for many purposes, such as train timetable inquiry [5], weather inquiry [33], flight reservations [25], and bus schedule guidance [23]. There are also some groups who have developed interesting multimodal applications on mobile platforms or backed by a geographical database, such as "AdApt" [8], "MATCH" [12], "SmartWeb" [29], and "CHAT" [30].

Most of these systems are mainly for factoid question-answering tasks and have pre-programmed dialogue templates to perform restricted dialogue routines in a specific domain. For more complicated tasks such as aggregated data access, however, the syntax and semantics are very complex, not to mention the ambiguity of discourse in multiple-turn conversation. Thus, we have to go beyond simple question-answering routines or manually designed templates, and employ a more sophisticated dialogue modeling mechanism in order to present the highlighted information of summarized UGC in natural and interactive dialogue, as exemplified in Fig. 1.

Naturally, the task boils down to two challenges: 1) how to equip a standard dialogue system with capabilities of extracting context-relevant information from rich yet unstructured data like user-generated content and summarizing it into an aggregated form; and 2) how to present the condensed information to users in sophisticated dialogues with natural responses.

III. Dialogue-Oriented Unstructured Data Processing

An example of user-generated content is shown in Fig. 2. An information aggregation system should be able to obtain and summarize user-generated content into a condensed information representation and utilize it as a knowledge base for multimodal data access services. A possible representation format is shown in Table I, which summarizes the example reviews into

Eclectic but awesome *by Alice* *Rating: 4.5*

- ***Pros:*** *Fantastic food; super friendly staff*
- **Cons:** none really

This food was fantastic. I didn't go here with stellar expectations, I think b/c I couldn't quite make sense of the menu. I came here for my friend's bday. 5 of us, 2 vegetarians. We ordered the Asian salad…

An Underated Jewel of a Restaurant *by Bob* *Rating: 5*

- ***Pros:*** *sexy ambience, sassy crowd + satisfying small plates*
- ***Cons:*** *the wait is often too long later in the week.*

By now, most of you know (or otherwise should) that Cuchi Cuchi remains the best place to go when you want both delicious, authentic Spanish Tapas and a warm, romantic setting…

Fig. 2. User-generated reviews on a restaurant called "Cuchi Cuchi" published on www.citysearch.com. Each review mainly contains a "title," an "overall rating," "Pros," "Cons" and a free-style comment.

TABLE I
EXAMPLE OF A SUMMARY GENERATED FROM THE REVIEWS IN FIG. 2.

Aspect	Extracted phrases	Rating
Atmosphere	*sexy ambience, sassy crowd, warm romantic setting*	*4.8*
Food	*satisfying small plates, fantastic food, authentic Spanish Tapas*	*4.1*
Service	*super friendly staff*	*4.3*
General	*awesome restaurant, best place to go, long wait*	*3.8*

representative aspects and calculates an average rating for each aspect.

To achieve this goal, there are a few problems to tackle. Firstly, the representative phrases (e.g., opinion-related expressions) have to be identified and extracted from the original unstructured data. Secondly, we need to estimate the sentiment in these extracted opinion-related phrases, ideally on a numerical scale, in order to calculate aspect ratings. Thirdly, to generate a condensed summary of the original unstructured data, we have to filter out irrelevant or low quality phrases and catalogue the high-quality and relevant phrases into representative aspects. Furthermore, an advanced dialogue modeling mechanism is required to represent the catalogued information in natural sentences.

In this work, we will explore an unstructured data aggregation process, with a combination of linguistic and statistical approaches to analyzing the semantics and the sentiment of data as well as generating a summarized database. Fig. 3 (the bottom layer) shows the pipeline of the process. Briefly speaking, user-generated documents will be subjected to a linguistic parser for context-relevant phrase extraction (Section III.A), and a cumulative offset model can estimate the sentiment degrees of the extracted expressions (Section III.B). A classification model can be used to select high-quality phrases for further topic clustering and aspect rating (Section III.C), in order to create a summary database that can be accessed by the dialogue system (the upper layer of Fig. 3). A sentiment-support dialogue modeling mechanism is also explored to generate recommendation-like conversations (Section III.D).

A. Parse-and-Paraphrase Paradigm for Phrase Extraction

Firstly, we investigate an approach to extracting opinion-relevant phrases from user-generated content. There have been

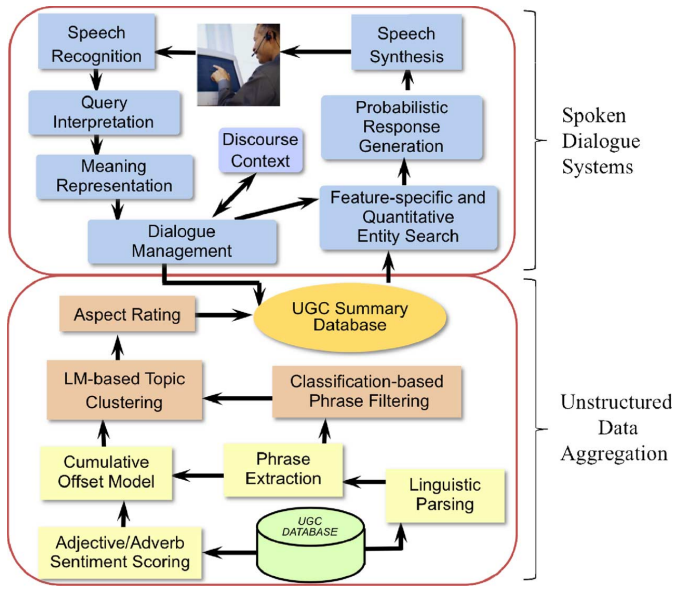


Fig. 3. The framework of the proposed approaches. The bottom layer is the aggregation process of user-generated content. The upper level is spoken dialogue systems.

many studies on utilizing statistical methods such as topic models for user-generated text analysis. Latent topics and underlying semantic concepts can be revealed by these methods [9], [26], [27]. For the application of dialogue systems, however, the focus is not only learning the general concepts, but also extracting representative topics from each user-generated document (e.g., "chicken tikka masala," "spaghetti carbonara").

Thus, we propose a parse-and-paraphrase paradigm to extract adverb-adjective-noun phrases from unstructured documents based on clause structure obtained by parsing sentences into a hierarchical representation [15]. Instead of the flat structure of a surface string, the parser provides a hierarchical representation, which we call a linguistic frame. It preserves linguistic structure by encoding different layers of semantic dependencies. The grammar captures syntactic structure through a set of carefully constructed context free grammar rules, and employs a feature-passing mechanism to enforce long distance constraints.

An example linguistic frame is shown in Fig. 4, which encodes the parsing results of the sentence "The caesar with salmon or chicken is really quite good." In this example, for the adjective "good," the nearby noun "chicken" would be associated with it if only proximity is considered. From the linguistic frame, however, we can easily associate "caesar" with "good" by extracting the head of the topic sub-frame and the head of the predicate sub-frame, which are encoded in the same layer (root layer) of the linguistic frame. In this way, long-distance dependencies are taken into consideration based on the semantic structure of sentences.

To produce the opinion-relevant phrases, a set of generation rules is carefully constructed to only extract sets of related adverbs, adjectives and nouns. For example, the adjective-noun relationships for opinion-relevant phrases can be captured from the following linguistic patterns: (1) all adjectives attached directly to a noun in a noun phrase, (2) adjectives embedded in a relative clause modifying a noun, and (3) adjectives related to

```
{c cstatement
 :topic {q caesar
        :quantifier "def"
        :pred {p with
                :topic {q salmon
                        :pred {p conjunction
                                :or {q chicken }}}}
 :adv "really"
 :pred {p adj_complement
        :pred {p adjective
                :adv "quite"
                :pred {p quality
                        :topic "good" }}}}
```

Fig. 4. An example of the hierarchical linguistic frame generated for the sentence, "the caesar with salmon or chicken is really quite good".

nouns in a subject-predicate relationship in a clause. These patterns are compatible, i.e., if a clause contains both a modifying adjective and a predicate adjective related to the same noun, two adjective-noun pairs are generated by different patterns. As in, "*The efficient waitress was nonetheless very courteous*." It is a "parse-and-paraphrase-like" paradigm: the paraphrase tries to preserve the original words intact, while reordering them and/or duplicating them into multiple noun phrase units. Since they are based on syntactic structure, the generation rules can also be applied in any other domain.

Generation rules can also be constructed to extract adverbials that are associated with descriptive adjectives. For example, in Fig. 4, there is an adverb "quite" modifying the head word "good" in the predicate sub-frame. The linguistic frame also encodes an adverb "really" in the layer immediately above. A set of well-constructed generation grammar rules can create customized adverb-adjective-noun phrases such as "really quite good caesar".

The linguistic parsing approach relies on linguistic features that are independent of word frequencies. Therefore, it can retrieve very rare phrases, which are very hard to derive from correlated topic models or frequency statistics (e.g., "very greasy chicken tikka masala").

B. Linear Additive Model for Sentiment Degree Scoring

After extracting context-relevant phrases, the next task is to explore a robust general solution for assessing the sentiment values of the extracted phrases. Our goal is to estimate a numerical sentiment degree for each expression on the phrase level. Given a user's spoken input query, the dialogue system needs to understand the sentiment expressed in the user's utterance in order to provide appropriate responses. A unified numerical sentiment scale would be easier for the system to interpret and handle rather than various textual expressions.

Our primary approach to sentiment scoring is to make use of community-generated data such as users' ratings. We assume that the rating by a user is normally consistent with the tone of the text published by the same user. By associating the rating with review texts (pros/cons and free-style comment) from each user, we can easily associate numerical scores with textual sentiment.

When calculating the sentiment score, we consider adverbs and adjectives separately, treating each modifying adverb as a universal quantifier, which consistently scales up/down the strength of sentiment for the adjectives it modifies. This allows

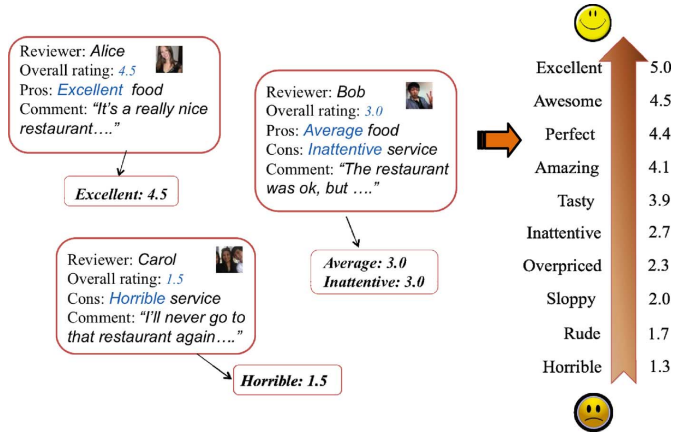


Fig. 5. Illustration of generating the sentiment scale for adjectives from original reviews and ratings published by different users.

us to pool all instances of a given adverb regardless of which adjective it is associated with, in order to compute the absolute value of the perturbation score for that adverb. A novelty of our system is the consistent treatment of negations, which are handled in the same way as modifying adverbs.

Thus, for each adjective, we collect all the occurrences of this adjective in the corpus, and average all the ratings from each user who published a comment that contains this adjective:

$$Score(adj) = \frac{\sum_{i \in P} \frac{N}{n_{r_i}} \cdot r_i}{\sum_{r_i} \frac{N}{n_{r_i}}} \tag{1}$$

where P represents the set of appearances of adjective adj, r_i represents the associated user rating in each appearance of adj, N represents the number of entities (e.g., restaurants, hotels) in the entire data set, and n_{r_i} represents the number of entities with rating r_i. The score is averaged over all the appearances, weighted by the frequency count of each category of rating to remove bias towards any category.

Fig. 5 illustrates the process of generating averaged sentiment scores for adjectives from user-generated comments and ratings. From each user, the adjectives in the "Pros" and "Cons" are associated with the "Overall rating" given by the same user. The ratings on each adjective are then averaged among all the data within the corpus.

As for adverbs, using a slightly modified version of (1), we can get an average rating for each adverb-adjective pair ($adv - adj$). For each adverb adv, we get a list of all its possible combinations with adjectives. Then, for each adjective adj in the list, we calculate the distance between the rating of $adv - adj$ pair and the rating of the adj alone. We then aggregate the distances among all the pairs of $adv - adj$ and adj in the list, weighted by the frequency count of each $adv - adj$ pair:

$$Score\,(adv) = \sum_{t \in A} \frac{count(adv, adj_t)}{\sum_{j \in A} count\,(adv, adj_j)} \cdot Pol(adj_t) \cdot (r\,(adv, adj_t) - r\,(adj_t)) \tag{2}$$

where $count(adv, adj_t)$ represents the count of the combination $adv - adj_t$, A represents the set of adjectives that co-occur with adv, $r\,(adv, adj_t)$ represents the sentiment rating of the

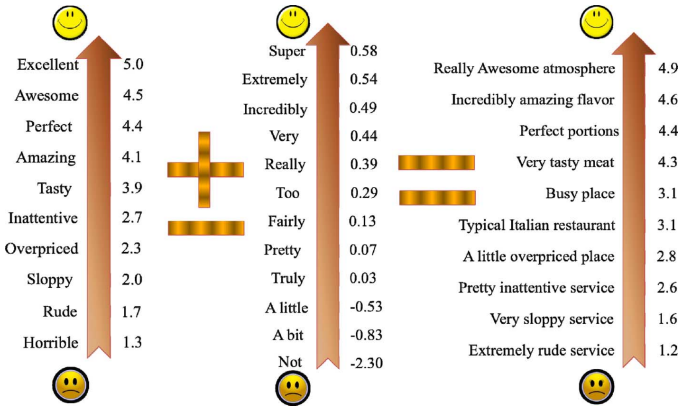


Fig. 6. Illustration of sentiment computation with the additive model, with the scales of sentiment strength for adjectives, adverbs, and phrases from left to right, and positive to negative from top to bottom.

combination $adv - adj_t$, and $r(adj_t)$ represents the sentiment rating of the adjective adj_t alone. $Pol(adj_t)$ represents the polarity of adj_t, which is assigned a value 1 if adj_t is positive (if the sentiment score of adj_t is higher than the pivot of the sentiment scale), and -1 if negative (if the sentiment score of adj_t is lower than the pivot of the sentiment scale).

Specifically, negations are well handled by the same scoring strategy, treated exactly the same way as modifying adverbs, except that they get such strong negative scores that the sentiment of the associated adjectives is pushed to the other side of the polarity scale.

After obtaining the averaged sentiment rating for adjectives and adverbs, we could assign a linearly combined score as the measurement of sentiment degree to each phrase (negation-adverb-adjective-noun) extracted by linguistic analysis, as given by:

$$\begin{aligned} &Score\left(neg\left(adv\left(adj\right)\right)\right) \\ &\quad = r\left(adj\right) + Pol\left(adj\right) \cdot r\left(adv\right) + Pol(adj) \cdot r(neg) \end{aligned} \quad (3)$$

where $r\left(adj\right)$ represents the rating of adjective adj, $r\left(adv\right)$ represents the rating of adverb adv, and $r(neg)$ represents the rating of negation neg. $Pol\left(adj\right)$ represents the polarity of adj, which is assigned a value 1 if adj is positive, and -1 if negative. Thus, if adj is positive, we assign a combined rating $r\left(adj\right) + r\left(adv\right)$ to this phrase. If it is negative, we assign $r\left(adj\right) - r\left(adv\right)$. Specifically, if it is a negation case, we further assign a linear offset $r(neg)$ if adj is positive or $-r(neg)$ if is negative. Fig. 6 shows an illustration of the cumulative offset model for phrase sentiment scoring.

C. Phrase Classification and Opinion Summary Generation

Given the set of opinion phrases extracted from user-generated data and a sentiment value assigned to each phrase, the next step is to choose the most representative (i.e., informative and relevant) phrases to generate an opinion summary database [16]. The task of phrase selection can be defined as a classification problem:

$$y = \overline{\theta} \cdot \overline{x} = \sum_{i=1}^{n} \theta_i x_i \quad (4)$$

where y is the label of a phrase, which is assigned a value '1' if the phrase is highly informative and relevant, and '-1' if the phrase is uninformative. $\overline{x}$ is the feature vector extracted from the phrase, and $\overline{\theta}$ is the coefficient vector.

Classification models such as SVMs [11] and decision trees [22] can be trained to automatically classify high/low informative phrases. From each phrase, we extract a set of features for model training. These features are treated as x_i in (4) and a classification model can be learned from the training data. Phrases in the test set labeled with "1" by the classification model are considered as highly informative phrases and can be further pruned as well as catalogued to create UGC summaries.

We take the sentiment score of each phrase generated by the cumulative offset model (as aforementioned) as a sentiment feature, which shows not only the polarity of sentiment but also the degree of orientation level. We also employ a set of standard statistical features for model training, such as the unigram probability of the adjective or noun in a phrase, the unigram probability of the phrase and the bigram probability of the adjective-noun pair in a phrase.

Statistical features, however, fail to reveal the underlying semantic meaning of phrases. To capture the semantic importance of each phrase, we first cluster the topics of phrases into generic semantic categories. There are often multiple topics mentioned in each review and even in each review sentence, so standard document-level or sentence-level topic clustering methods would bring in a lot of noise. Thus, for this particular task, to take only adjacent words into account, we use a language-model-based phrase-level topic-clustering algorithm:

$$\begin{aligned} P(t_c|t_i) &= \sum_{a \in A} P\left(t_c|a\right) \cdot P\left(a|t_i\right) \\ &= \sum_{a \in A} \frac{P\left(a, t_c\right)}{P\left(a\right)} \cdot \frac{P\left(a, t_i\right)}{P\left(t_i\right)} \\ &= \frac{1}{P\left(t_i\right)} \sum_{a \in A} \frac{1}{P\left(a\right)} \cdot P\left(a, t_c\right) \cdot P\left(a, t_i\right) \end{aligned} \quad (5)$$

where A represents the set of all the adjectives in the corpus. We first select a small set of initial topics with the highest frequency counts (e.g., "food," "service" and "atmosphere" in the restaurant domain). Then, for each of the other topics t_c (e.g., "chicken," "waitress" and "décor"), we calculate its similarity with each initial topic t_i based on the adjective-noun bigram statistics. For those topics with conditional probability higher than a threshold for an initial topic t_i, we assign them to the cluster of t_i, assuming intuitively that these topics have high semantic similarity with the cluster topic t_i, given that they co-occur most frequently with the same set of adjectives. We then use this as a semantic feature, e.g., whether the topic of a phrase belongs to a generic semantic category. Table II gives some topic clustering examples.

This language-model-based method relies on bigram probability statistics and can well cluster highly frequent topics to generic topic categories. Domain-specific categories, however, may contain a very large vocabulary. For example, in the restaurant domain, the category of "food" contains various topics from

TABLE II
TOPIC TO SEMANTIC CATEGORY CLUSTERING.

Category	Relevant Topics
food	*appetizer, beer, bread, fish, fries, ice cream, margaritas, menu, pizza, pasta, rib, roll, sauce, seafood, sandwich, steak, sushi, dessert, cocktail, brunch*
service	*waiter, staff, management, server, hostess, chef, bartender, waitstaff*
atmosphere	*décor, ambiance, music, vibe, setting, environment, crowd*
price	*bill, pricing, prices*

generic sub-categories (such as "sushi," "dessert" and "sandwich") to specific courses (such as "bosc pear bread pudding" and "herb roasted vermont pheasant wine cap mushrooms"). These domain-specific topics normally have very low frequencies in a UGC corpus, yet they are highly context-relevant and valuable. But many of them are discarded by the frequency-based topic clustering.

To recover these context-relevant yet low-frequency topics, we employ external context resources such as a context-related ontology (e.g., a restaurant-domain ontology), which can be constructed from web resources such as online menus of restaurants. Based on such a context-relevant ontology, another set of semantic features covering low-frequency topics can be extracted (e.g., whether a phrase contains the name of a specialty) for the classification model training.

After the classification, phrases identified with positive labels (highly informative and relevant ones) are further clustered into different aspects according to the semantic categories and the hierarchical ontology. An average sentiment score for each aspect is calculated by:

$$ave(s_t) = \frac{\sum_{j \in N_s} r_j}{|N_s|} \tag{6}$$

where s_t represents the aspect s of entry t (t can be a restaurant, a movie, or a consumer product), N_s represents the set of phrases in the cluster of aspect s, and r_j represents the sentiment score of phrase j within the cluster.

The opinion-related phrases are extracted from a large number of documents, and many of them may include the same topic (e.g., "good fish," "not bad fish" and "above-average fish" from different reviews for one restaurant). Thus, redundancy elimination is required. In each category, among those phrases with the same topic, we select the phrase whose sentiment score is closest to the average score of this aspect as the most representative phrase:

$$j^* = argmin_{j \in N_i}(|r_j - ave(s_t)|) \tag{7}$$

where $ave(s_t)$ represents the average sentiment score of aspect s, N_i represents the set of phrases on the same topic i, and r_j represents the sentiment score of phrase j within N_i. The goal is to find the phrase j^* for each topic i, the sentiment score of which has the smallest distance to the average aspect rating.

This sequence of phrase classification, topic categorization, phrase pruning and redundancy elimination results in a summary database. An example database entry is exemplified in Table III, which contains lists of descriptive phrases in major aspects ("Atmosphere," "Food," "Service," "Specialty," and "General") as well as ratings (e.g., "Atmosphere_rating," "Food_rating," "Service_rating," and "General_rating").

TABLE III
EXAMPLE OF A UGC SUMMARY DATABASE.

Name	"Devlin's restaurant and bar"
City	*"Brighton"*
Cuisine	*"American"*
Atmosphere	*"romantic date" "elegant decor"*
General	*"great place"*
Food	*"wonderful martinis" "good wine" "great fish"*
Service	*"fast service"*
Specialty	*"martinis" "wine" "fish"*
Atmosphere_rating	*"4.2"*
General_rating	*"4.2"*
Food_rating	*"4.3"*
Service_rating	*"3.9"*

D. Dialogue Modeling

To make the system present the highlighted information to users via interactive conversations, an adaptive dialogue modeling mechanism [25] driven by the UGC summary database is required to handle discourse and dialogue. To be consistent, we will continue using the restaurant domain for demonstration.

Users' feature-specific queries can be handled well with keyword search (e.g., search by "martinis," "sushi," or "fish"). For high-level qualitative questions (e.g., "show me some American restaurants with nice ambience"), however, the keyword search method is problematic, as there are normally multiple variants of expressions with the same qualitative meaning. For example, given the query "nice ambience," entities with "friendly vibes" or "excellent atmosphere" also satisfy the query and should be retrieved, but they would have been missed by keyword search methods due to different expressions from the query words.

As aforementioned, we proposed a method for calculating a sentiment score for each opinion-expressing adjective and adverb (e.g., "bad: 1.5," "good: 3.5," "great: 4.0," on a scale of 1 to 5). Here, we make use of these sentiment scores to convert the qualitative queries into measurable values. The numerical sentiment values can be used to search the database on aspect ratings. In this way, opinion expressions can be interpreted by a measurable scale and database entries with descriptive words different from the user's query, but with similar sentiment values, can be recovered.

Fig. 7 shows an exemplified procedure of handling qualitative queries. When a user's utterance is submitted to the system and passed through speech recognition, a linguistic parser parses the sentence into a linguistic frame, from which a set of key-value pairs is extracted as a meaning representation of the utterance (the second step in the Figure).

As shown in the third step of Fig. 7, by mapping the descriptive word "great" into its sentiment score "4.0" (the sentiment scores for descriptive words are learned automatically by the sentiment scoring method as aforementioned), the key-value pairs "property: food, quality: great" are converted to "food_rating: 4.0." An algorithm can be defined as filtering the database for entities that have scores higher than the inquired value (e.g., ": $\mathrm{food\_rating} > 4.0$"). In this way, the qualitative

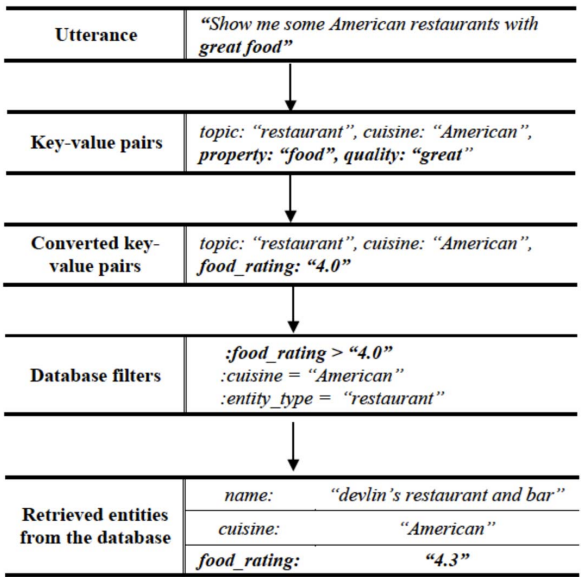


Fig. 7. Illustration of the procedure for handling a qualitative query.

query can be easily converted to measurable values; and the entities that are in the same range of sentiment degree as the user's query can be retrieved from the database.

IV. Experiments and Evaluation

In this section, we present a systematic evaluation of the proposed approaches with real user-generated data. We took the restaurant domain as an example and harvested a collection of 137,569 user-published reviews on 24,043 restaurants in 9 cities in the U.S. from an online restaurant evaluation website[2]. 99,147 reviews containing pros/cons were taken as the experimental set (72.7% of the original set), and 857,466 review sentences remained after a noise filtering process. This set was then subjected to parse analysis [24], and 78.6% of them were parsable. Given the parsing results in the format of a linguistic frame, we used a set of generation rules to extract context-relevant adverb-adjective-noun phrases (as explained in Section III).

A. Linguistic Parsing for Phrase Extraction

To evaluate the performance of the proposed approach (*LING*) to phrase extraction, we compared it with a baseline method similar to [10]. We performed part-of-speech tagging on both parsable and unparsable sentences, extracted each pair of noun and adjective that has the smallest proximity, and filtered out those with low frequency counts. Adverbs and negation words that are adjacent to the identified adjectives were also extracted along with the adjective-noun pairs. We call this the "neighbor baseline" (*NB*).

The proposed method is unable to make use of the non-parsable sentences, which make up over 20% of the data. Hence, it seems promising to utilize a back-off mechanism for these sentences via a combined system (*COMB*) incorporating *NB* for the sentences that fail to parse.

The phrases in the pros/cons of each review are considered as the ground truth. Performance was evaluated in terms of recall (percentage of phrases in the ground truth that are also identified from the review body) and precision (percentage of phrases extracted by the system that are also in the ground truth).

TABLE IV
Experimental Results of Phrase Extraction.

	NB	LING	COMB
Recall	44.4%	57.0%	61.9%
Precision	56.8%	61.1%	60.8%

As shown in Table IV, the *LING* approach gets both higher recall and higher precision than the *NB* baseline. The *COMB* approach gets the highest recall, with a 4.9% and 17.5% increase from the *LING* approach and the *NB* baseline, respectively. The precision is quite close to that of the *LING* approach (60.8% vs. 61.1%). This shows that the linguistic parsing approach can retrieve more context-relevant phrases by preserving the hierarchical semantic structure of a sentence; and, by combining a keyword matching method for unparsable sentences, the approach can get even higher coverage, without sacrificing much precision.

As shown in the results, the best-preforming system could achieve a precision up to 60%. We suspected that the over-generated phrases (the 40% of phrases that find no mappings in the pros/cons) might not really be a problem. To test this hypothesis, we selected 100 reviews for their high density of extracted phrases, and manually evaluated all the over-generated phrases. We found that over 80% were well formed, correct, and informative. Therefore, a lower precision here does not necessarily mean poor performance, but instead shows that the pros/cons provided by users are often incomplete. By extracting phrases from free-style review texts we can recover additional valuable information at the expense of additional processing.

B. Sentiment Analysis

We evaluate the sentiment scoring approach with the same restaurant-review corpus. The pros/cons in a review entry often have clear sentiment orientations. Thus, we use pros/cons to estimate the sentiment values of adjectives, which requires strong polarity association. On the other hand, the frequencies of adverbs in free-style texts are much higher than those in pros/cons, as pros/cons mostly contain adjective-noun patterns. Thus, we used free-style texts instead of pros/cons to calculate the sentiment strength of adverbs.

To obtain reliable ratings, we arbitrarily associated the adjectives in the "pros" of review entries that have a user rating of 4 or 5, and associated the adjectives in the "cons" of review entries with user ratings of 1 or 2 (on a scale of user rating from 1 to 5). Reviews with rating 3 express neutral sentiment, so we associated both "pros" and "cons" with the overall rating in these cases. Using the algorithms for sentiment scoring ((1) and (2)), we calculated sentiment scores for each adjective and common adverb that appeared in the review corpus [15].

To evaluate the performance of sentiment scoring, we randomly selected a subset of 1,000 adjective-noun phrases from the set extracted by our linguistic analysis and asked two annotators to independently rate the sentiment of each phrase on a

[2]http://www.citysearch.com

scale of 1 to 5. We compared the sentiment scoring between our system and the annotations in a measurement of mean distance:

$$distance = \frac{1}{|S|}\sum_{p \in S} |r_{ip} - r_{ap}| \tag{8}$$

where S represents the set of phrases, p represents each phrase in the set S, r_{ip} represents the rating on phrase p from our sentiment scoring system, and r_{ap} represents the annotated rating on phrase p.

The kappa agreement [3] between the two annotation sets is 0.68, indicating high consistency between the annotators. The obtained mean distance between the scoring from our approach and that from each annotation set is 0.46 and 0.43, respectively, based on the absolute rating scale from 1 to 5. This shows that the scoring of sentiment from our system is relatively close to human annotation. This is easy to understand as the sentiment score of each adjective/adverb is averaged on the ratings over a large user base. The reliability of these results gives us sufficient confidence to make use of these scores as indications of sentiment values.

To examine the prediction accuracy of sentiment polarity, for each annotation set, we pooled the phrases with rating 4–5 into "positive," rating 1–2 into "negative," and rating 3 into "neutral." Then we rounded up the sentiment scores from our system to integers and pooled the scores into three polarity sets ("positive," "negative" and "neutral") in the same way. The obtained kappa agreement between the result from our system and that from each annotation set is 0.55 and 0.60 respectively. This shows reasonably high agreement on the polarity of sentiment between our system and human evaluation.

C. Phrase Classification

To evaluate the phrase classification approach, we randomly selected 3,000 phrases as training data, extracted from the pros/cons of reviews by the linguistic parsing method (the phrases in pros/cons are considered as well-formatted). To generate a human-judgment-consistent training set, we manually labeled the training samples with "$\langle \mathrm{GOOD} \rangle$" and "$\langle \mathrm{BAD} \rangle$" labels, based on whether a phrase contains opinion-relevant information (e.g., "delicious pasta: $\langle \mathrm{GOOD} \rangle$"; "red wine: $\langle \mathrm{BAD} \rangle$"). We then randomly selected a subset of 3,000 phrases extracted from free-style review texts as the test set, and labeled the phrases with the same "$\langle \mathrm{GOOD} \rangle$" and "$\langle \mathrm{BAD} \rangle$" labels as the ground truth. The kappa agreement between the two sets of annotations is 0.73, indicating substantial consistency.

We employed the three types of features (statistical, sentiment and semantic features) as aforementioned to train the SVMs and the decision tree models for phrase classification. We extracted the unigrams/bigrams from the phrases as statistical features, and employed the sentiment scores calculated for the phrases as the sentiment features.

To extract context-related semantic features, we collected a large pool of well-formatted menus from an online resource[3], which contains 16,141 restaurant menus. Based on the hierarchical structure of these collected menus, we built up a context-related ontology and extracted a set of semantic features from the ontology. To extract topic-categorization semantic features, we selected 6 topics that had the highest frequencies in the corpus and represented appropriate dimensions for the restaurant domain ("place," "food," "service," "price," "atmosphere" and "portion") as the initial set, and clustered the topics of extracted phrases into different aspect categories with the bigram-based topic clustering method.

[3]http://www.menupages.com

TABLE V
PRECISION ON PHRASE CLASSIFICATION USING THE BASELINE, SVM MODEL, AND THE DECISION TREE ALGORITHM.

	Baseline	SVM	Decision tree
Annotation 1	61.5%	72.0%	77.9%
Annotation 2	51.3%	63.2%	74.5%

We used these features to train the SVMs and the decision trees as the classification models. To select the most valuable features for model training, we conducted a set of leaving-one-feature-out experiments for both models. We found that all the features except the adjective unigram probability contribute positively to model learning. From further data analysis we observed that many phrases with popular adjectives have context-unrelated nouns (e.g., "good friends" "nice weather"), which means that, although the adjective unigram probability might be high, the phrase is still context irrelevant and is a negative sample. Thus, adjective unigram probability is not a good indicator for phrase relevance. Using the adjective unigram probability as a learning feature will mislead the system into trusting an adjective that is common but has a poor bigram affinity to the context-relevant noun in the phrase. Therefore, we eliminated this feature for both the SVMs and the decision tree learning.

To evaluate the performance of the classification models, we took a set of intuitively motivated heuristic rules as the baseline, which uses variations of all the features except the unigram probability of adjectives. The performance of classification by different models is shown in Table V. Although the heuristic rule algorithm is complicated and involves human knowledge, both of the statistical models trained by SVMs and the decision tree algorithms outperform the baseline. The SVM model outperforms the baseline by 10.5% and 11.9% on the two annotation sets, respectively. The decision tree model outperforms the baseline by 16.4% and 23.2% (average relative improvement of 36%), and it also outperforms the SVM model by 5.9% and 11.3% (average relative improvement of 13%).

The classification model using the decision tree algorithm can achieve a precision of 77.9% and 74.5% compared with the ground truth. These values indicate the results are quite comparable to human judgment, considering that the precision of one annotation set based on the other is 74% (using one annotation set as the reference and the other as the comparison set). This shows that the decision tree model can predict phrase labels as reliably as human judgment. Part of the reason is that the decision tree algorithm can make better use of a combination of Boolean value features (e.g., whether a topic belongs to a context-related ontology) and continuous value features. Also, as the phrase classification task is very subjective, it is very similar

TABLE VI
A SCENARIO EXAMPLE IN OUR USER STUDY.

"You live in Brighton and you have a friend coming to visit you this weekend. You plan to take him to an American restaurant. You both like some place with nice martinis. "

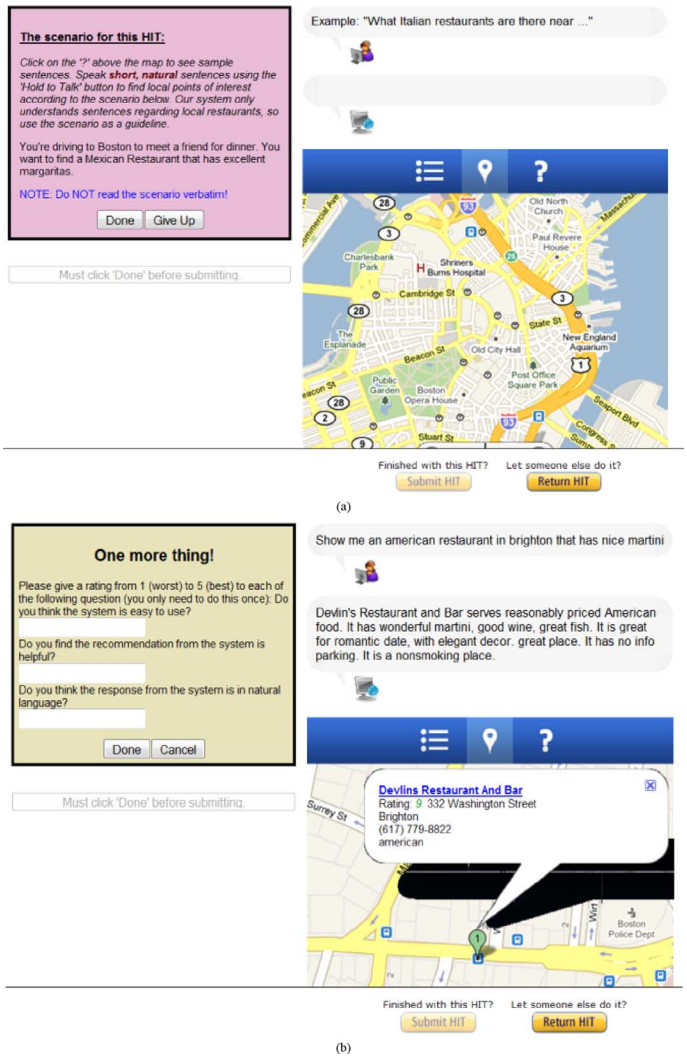


Fig. 8. Screen shots of our dialogue system before and after a user worked on an AMT HIT. The picture on the top (a) shows the instruction and the scenario of the task, and the picture on the bottom (b) shows the feedback sheet and highlights the recommended restaurant on the map.

to a 'hierarchical if-else decision problem' in human cognition, where decision tree algorithms can fit well.

D. Dialogue and Response

To evaluate our proposed framework of developing speech-based interfaces for UGC data access, we applied it to a restaurant-domain dialogue system. The web-based multimodal spoken dialogue system, CityBrowser [7], developed in our group, can provide users with information about various landmarks such as the address of a museum, or the opening hours of a restaurant. To evaluate our approaches, we selected the phrases identified as "$\langle$GOOD$\rangle$" by the classification model as the candidate pool. These phrases are further catalogued and pruned to create a structured aspect-based summary database. We applied the summary database to the CityBrowser system and implemented the sentiment-involved dialogue modeling algorithms to the system [17].

To collect data from real users, we utilized the platform of Amazon Mechanical Turk (AMT)[4]. We conducted a first AMT task by collecting restaurant inquiries from general users, and extracted a set of generic templates encoding the language patterns of all the sentences. We used these templates to automatically create 10,000 sentences for language model training for the speech recognizer.

To evaluate the quality of dialogue, we conducted another user study on AMT. We presented the system to real users and gave each subject a set of assignments to fulfill. Each assignment is a scenario involving finding a particular restaurant. There are ten HITs (Human Intelligence Tasks) available for each subject. A scenario is randomly assigned in each HIT, and the subject can decide to work on the HIT or skip it. An exemplary scenario is shown in Table VI.

An example of a HIT in this AMT task is shown in Fig. 8(a). The instructions on the left-hand side (titled "The scenario for this HIT") give a randomly selected scenario. The user can talk to the system via a microphone and ask for recommendations for restaurants. The map on the right-hand side of the picture locates the recommended restaurants.

To obtain a subjective evaluation from general users, we also gave each user a questionnaire and asked them to rate the system on different aspects. Fig. 8(b) shows the interface of an AMT HIT after the user has finished the scenario task, showing the questionnaire (on the left-hand) and the dialogue between the user and the system (above the map). The recommended restaurant is also shown on the map, providing detailed information such as the phone number and the address.

We collected 58 sessions and 34 surveys within 9 days through this AMT task. There are in total 270 utterances collected from the 58 sessions, with an average of 4.6 utterances per session. The length of the utterances varies significantly, from "Thank you" to "Restaurants along Brattle Street in Cambridge with nice cocktails." The average number of words per utterance is 5.3.

We examined the playback of each session. Among all the 58 sessions, 51 of them were successfully fulfilled, i.e., in 87.9% of the cases the system provided helpful recommendations upon the user's request and the user was satisfied with the system response. Among those seven failed cases, one was due to loud background noise, two were due to users' operation errors (e.g., clicking "DONE" before finishing the scenario), and four were due to recognition errors.

We also examined the feedback from the questionnaire and analyzed the results. On a scale of 1 to 5, the average rating on the ease of use of the system is 3.6. The average rating on the helpfulness of the system is 4.4. And the naturalness of the response from the system gets an average rating of 4.1. These numbers indicate that the system is helpful at providing recommendation upon users' inquiries, and the response from the system is presented in a natural way that people could easily understand.

[4]https://www.mturk.com/mturk/welcome.

The lower rating of ease of using the system is partially due to recognition errors. For example, a user asked for "pancakes," and the system recommended "pizza places" to him. In some audio clips recorded, the background noise is relatively high. This is unavoidable because many workers on AMT work from home. There were also some utterances that never occurred in our training sentences for the recognizer, such as "OK, I will take my date to that restaurant tonight," which our system had not yet been trained to handle.

V. Portability

In this section, we briefly describe our experiments in porting to novel domains and languages. Details can be found in the published literature.

To assess the portability of the proposed framework of unstructured data processing and speech interface implementation, we explored the effort required to port to a totally different domain — medical study and health care. Patient-provided drug reviews served as the unstructured user-generated content.

We implemented a spoken dialogue system [18] that allows consumers of prescription drugs to inquire about possible side effects that other patients have experienced. Summaries were extracted from a large number of patient-provided drug reviews on various health discussion sites. The construction of the spoken dialogue system inherited most of the components of the previously explored restaurant-domain system. Templates were created to generate synthetic training data for the speech recognizer language model training. In addition, the context-free grammar was expanded to handle domain-specific information, and the dialogue model was expanded to customize new commands for domain-specific queries. Details of the system implementation can be found in related work [14].

For language portability, we investigate the extension of the restaurant-guide system to a tone and character-based language — Mandarin Chinese. We derived a Mandarin dialogue system, CityBrower II [19], from its English predecessor, by focusing on the speech interface implementation regarding language differences, without handling Chinese reviews.

The system inherited most of the components of the predecessor English system and required only relatively minor changes to account for the differences between English and Mandarin Chinese. The major effort required is the translation of templates used to generate sentences for recognizer training, as well as the translation of language generation templates from the original language to the target one. For language understanding, we were able to leverage a pre-existing generic Mandarin grammar that had been developed in our lab, augmenting it with domain-specific content. The strategy is to maintain the meaning representation concepts in English, with only the values of database contents represented in the target language (Mandarin). We acquired restaurant-database content for two cities: Taipei and Beijing. The details of the Mandarin system implementation can be found in the related work [14].

VI. Conclusions and Future Work

In this work, we have explored a universal framework that supports multimodal access to user-generated content, with a speech-navigated web-based interface and a generalized platform for unstructured data processing. The contribution of this work lies in that it advances the integration of unstructured data summarization and speech-based human-computer interaction. With the help of such dialogue systems, users can access the online community-edited information more effectively and more efficiently.

To summarize large amounts of Web data into a condensed and structured database, we presented a framework for preprocessing unstructured UGC data. We proposed a parse-and-paraphrase approach to extracting representative phrases from sentences, as well as introducing an algorithm for assessing the degree of sentiment in opinion expressions based on user-provided ratings. We also used a phrase classification model to select context-relevant phrases automatically for creating succinct, descriptive and catalogued UGC summaries. To present the summarized information in natural responses, a dialogue-modeling framework was also introduced, which supports the generation of opinion-sharing conversations based on an aggregated UGC database.

To evaluate the framework, we collected a user-generated review corpus in the restaurant domain from the Web, which was processed through the proposed pipeline of unstructured data processing. A restaurant-domain recommendation system enhanced by this framework was implemented as a demonstration. Users can interact with the system via speech to inquire about restaurants and ask for recommendations on various dimensions such as service, food or ambiance. The interactions between real users and the prototype system were monitored for system evaluation. To demonstrate the portability of the approaches, we also applied the proposed framework in a different domain as well as in another language.

For future work, we will continue to improve the performance of the system through larger-scale data collections from general users. Another direction is to develop a speech interface for harvesting spoken UGC data (e.g., spoken reviews), which can allow users to add their own experience on restaurants, movies, drugs, etc., through natural speech and text. We will also explore crowd-sourcing methods to aid in the transcription of these recordings, such as relying on general users via Amazon Mechanical Turk. A more ambitious future goal is to develop a speech-based platform for general domains, an integrated system, which can interact with users in continuous conversations across multiple domains.

References

[1] S. R. K. Branavan, H. Chen, J. Eisenstein, and R. Barzilay, "Learning document-level semantic properties from free-text annotations," in *Proc. Annu. Conf. Assoc. for Comput. Linguist. (ACL)*, 2008.

[2] G. Carenini, R. Ng, and A. Pauls, "Multi-document summarization of evaluative text," in *Proc. Conf. Eur. Ch. Assoc. for Comput. Linguist. (EACL)*, 2006.

[3] J. Carletta, "Assessing agreement on classification tasks: The kappa statistic," *Comput. Linguist.*, vol. 22, no. 2, pp. 249–254, 1996.

[4] K. Dave, S. Lawrence, and D. M. Pennock, "Mining the peanut gallery: Opinion extraction and semantic classification of product reviews," in *Proc. Int. Conf. World Wide Web (WWW)*, 2003.

[5] W. Eckert, T. Kuhn, H. Niemann, S. Rieck, A. Scheuer, and E. G. Schukat-talamazzini, "A spoken dialogue system for german intercity train timetable inquiries," in *Proc. Eur. Conf. Speech Commun. Technol. (Eurospeech)*, 1993.

[6] A. Goldberg and X. Zhu, "Seeing stars when there aren't many stars: Graph-based semi-supervised learning for sentiment categorization," in *Proc. HLT-NAACL Workshop Textgraphs: Graph-Based Algorithms for Natural Lang. Process.*, 2006.
[7] A. Gruenstein and S. Seneff, "Releasing a multimodal dialogue system into the wild: User support mechanisms," in *Proc. 8th SIGDIAL Workshop Discourse and Dialogue*, 2007.
[8] J. Gustafson, L. Bell, J. Beskow, J. Boye, R. Carlson, J. Edlund, B. Granström, D. House, and M. Wirén, "AdApt a multimodal conversational dialogue system in an apartment domain," in *Proc. Int. Conf. Spoken Lang. Process. (ICSLP)*, 2000.
[9] R. Higashinaka, R. Prasad, and M. Walker, "Learning to generate naturalistic utterances using reviews in spoken dialogue systems," in *Proc. Joint Conf. Int. Committee Comput. Linguist. and the Assoc. for Comput. Linguist. (COLING-ACL)*, 2006.
[10] M. Hu and B. Liu, "Mining and summarizing customer reviews," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2004.
[11] T. Joachims, "Text categorization with support vector machines: Learning with many relevant features," in *Proc. 10th Eur. Conf. Mach. Learn. (ECML)*, 1998.
[12] M. Johnston, S. Bangalore, G. Vasireddy, A. Stent, P. Ehlen, M. Walker, S. Whittaker, and P. Maloor, "MATCH: An architecture for multimodal dialogue systems," in *Proc. Annu. Conf. Assoc. Comput. Linguist. (ACL)*, 2002.
[13] S.-M. Kim and E. Hovy, "Automatic identification of pro and con reasons in online reviews," in *Proc. Joint Conf. Int. Committee Comput. Linguist. and the Assoc. for Comput. Linguist. (COLING-ACL)*, 2006.
[14] J. Liu, "Harvesting and summarizing user-generated content for advanced speech-based human-computer interaction," Ph.D. dissertation, MIT Dept. of Elect. Eng. and Comput. Sci., Cambridge, MA, 2012.
[15] J. Liu and S. Seneff, "Review sentiment scoring via a parse-and-paraphrase paradigm," in *Proc. Conf. Empir. Meth. Nat. Lang. Process. (EMNLP)*, 2009.
[16] J. Liu, S. Seneff, and V. Zue, "Dialogue-oriented review summary generation for spoken dialogue recommendation systems," in *Proc. Conf. North Amer. Ch. Assoc. for Comput. Linguist.: Human Lang. Technol. (NAACL-HLT)*, 2010a.
[17] J. Liu, S. Seneff, and V. Zue, "Utilizing review summarization in a spoken recommendation system," in *Proc. SIGDIAL Meeting on Discourse and Dialogue*, 2010b.
[18] J. Liu and S. Seneff, "A dialogue system for accessing drug reviews," in *Proc. Autom. Speech Recognit. Understand. Workshop (ASRU)*, 2011.
[19] J. Liu, Y. Xu, S. Seneff, and V. Zue, "CityBrowser II: A multimodal restaurant guide in Mandarin," in *Proc. Int. Symp. Chin. Spoken Lang. Process. (ISCSLP)*, 2008.
[20] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up? Sentiment classification using machine learning techniques," in *Proc. Conf. Empir. Meth. Nat. Lang. Process. (EMNLP)*, 2002.
[21] A.-M. Popescu and O. Etzioni, "Extracting product features and opinions from reviews," in *Proc. Conf. Empir. Meth. Nat. Lang. Process. (EMNLP)*, 2005.
[22] J. R. Quinlan, "Induction of decision trees," in *Machine Learning*. New York: Springer-Netherlands, 1986.
[23] A. Raux, B. Langner, A. W. Black, and M. Eskenazi, "LET'S GO: Improving spoken dialog systems for the elderly and non-natives," in *Proc. Eur. Conf. Speech Commun. Technol. (Eurospeech)*, 2003.
[24] S. Seneff, "TINA: A natural language system for spoken language applications," *Comput. Linguist.*, vol. 18, no. 1, pp. 61–86, 1992.
[25] S. Seneff and J. Polifroni, "Dialogue management in the mercury flight reservation system," in *Proc. Dialogue Workshop, ANLP-NAACL*, 2000.
[26] B. Snyder and R. Barzilay, "Multiple aspect ranking using the good grief algorithm," in *Proc. Joint Conf. North Amer. Ch. Assoc. for Comput. Linguist. and Human Lang. Technol. (NAACL-HLT)*, 2007.
[27] I. Titov and R. McDonald, "A joint model of text and aspect ratings for sentiment summarization," in *Proc. Annu. Conf. Assoc. for Comput. Linguist. (ACL)*, 2008.
[28] P. D. Turney, "Thumbs Up or Thumbs Down? Sentiment orientation applied to unsupervised classification of reviews," in *Proc. Annu. Conf. Assoc. for Comput. Linguist. (ACL)*, 2002.
[29] W. Wahlster, "Dialogue systems go multimodal: The SmartKom experience," in *SmartKom: Foundations of Multimodal Dialogue Systems*. New York: Springer, 2006, pp. 3–27.
[30] F. Weng, S. Varges, B. Raghunathan, F. Ratiu, H. Pon-barry, B. Lathrop, Q. Zhang, H. Bratt, T. Scheideck, K. Xu, M. Purver, and R. Mishra, "CHAT: A conversational helper for automotive tasks," in *Proc. Annu. Conf. Int. Speech Commun. Assoc. (Interspeech)*, 2006.
[31] J. M. Wiebe, R. F. Bruce, and T. P. O'Hara, "Development and use of a gold standard data set for subjectivity classifications," in *Proc. 37th Annu. Meeting Assoc. for Comput. Linguist. (ACL)*, 1999.
[32] T. Wilson, J. Wiebe, and P. Hoffmann, "Recognizing contextual polarity in phrase-level sentiment analysis," in *Proc. Human Lang. Technol. Conf. and the Conf. Empir. Meth. in Nat. Lang. Process. (HLT/EMNLP)*, 2005.
[33] V. Zue, S. Seneff, J. Glass, J. Polifroni, C. Pao, T. J. Hazen, and L. Hetherington, "JUPITER: A telephone-based conversational interface for weather information," *IEEE Trans. Speech Audio Process.*, vol. 8, no. 1, pp. 85–96, Jan. 2000.

Jingjing Liu is a Research Scientist in the Spoken Language Systems group at MIT Computer Science and Artificial Intelligence Laboratory (CSAIL). She received her Ph.D. degree in Electrical Engineering and Computer Science at MIT in 2012. She has conducted research in diverse areas, including semantic language understanding, opinion mining and summarization, video search, and text retrieval. Her current research interests are in the areas of spoken dialogue systems, natural language processing and information retrieval.

Dr. Liu is a technical reviewer of several scientific journals, such as Journal of Computer Science and Technology, Multimedia Tools and Applications, Computer Speech and Language, IEEE TRANSACTIONS ON AUDIO, SPEECH AND LANGUAGE PROCESSING, ACM Transactions on Intelligent Systems and Technology, and EURASIP Journal on Audio, Speech, and Music Processing. She has also served on the program committee of INTERSPEECH, ACL-HLT, and EMNLP conferences.

Stephanie Seneff is a Senior Research Scientist at MIT's Computer Science and Artificial Intelligence Laboratory. She has conducted research in diverse areas, including human auditory modeling, spoken dialogue systems, natural language processing, information retrieval and summarization, and computational biology. She has published nearly 200 refereed articles in technical journals and conferences on these subjects, and has been invited to give several keynote speeches. She is currently conducting research on information summarization and extraction in the medical domain, leading to spoken dialogue systems to support intelligent search and retrieval of user-provided reviews of medical services. Dr. Seneff is an ISCA Fellow.

Victor Zue is the Delta Electronics Professor of Electrical Engineering and Computer Science at MIT. Earlier in his career, Victor conducted research in acoustic phonetics and phonology, codifying the acoustic manifestation of speech sounds and the phonological rules in American English. Subsequently, his research interest shifted to the development of spoken language interfaces to make human-computer interactions easier and more natural. Subsequently, he led a team at MIT that pioneered the development of spoken dialogue systems. His current research interests are in the area of applying human language technologies to enable easy access of structured and unstructured information from the web, especially in applications such as education and healthcare.

Victor is a Fellow of the Acoustical Society of America, and a Fellow of the International Speech Communication Association. He is also a member of the U.S. National Academy of Engineering, and an Academician of the Academia Sinica in Taiwan.