Dialogue-Oriented Review Summary Generation for Spoken Dialogue Recommendation Systems

Jingjing Liu, Stephanie Seneff, Victor Zue
MIT Computer Science & Artificial Intelligence Laboratory
32 Vassar Street, Cambridge, MA 02139
{jingl, seneff, zue}@csail.mit.edu

Abstract

In this paper we present an opinion summarization technique in spoken dialogue systems. Opinion mining has been well studied for years, but very few have considered its application in spoken dialogue systems. Review summarization, when applied to real dialogue systems, is much more complicated than pure text-based summarization. We conduct a systematic study on dialogue-system-oriented review analysis and propose a three-level framework for a recommendation dialogue system. In previous work we have explored a linguistic parsing approach to phrase extraction from reviews. In this paper we will describe an approach using statistical models such as decision trees and SVMs to select the most representative phrases from the extracted phrase set. We will also explain how to generate informative yet concise review summaries for dialogue purposes. Experimental results in the restaurant domain show that the proposed approach using decision tree algorithms achieves an outperformance of 13% compared to SVM models and an improvement of 36% over a heuristic rule baseline. Experiments also show that the decision-tree-based phrase selection model can achieve rather reliable predictions on the phrase label, comparable to human judgment. The proposed statistical approach is based on domain-independent learning features and can be extended to other domains effectively.

1 Introduction

Spoken dialogue systems are presently available for many purposes, such as weather inquiry (Zue et al., 2000), bus schedules and route guidance (Raux et al., 2003), customer service (Gorin et al., 1997), and train timetable inquiry (Eckert et al., 1993). These systems have been well developed for laboratory research, and some have become commercially viable.

The next generation of *intelligent* dialogue systems is expected to go beyond factoid question answering and straightforward task fulfillment, by providing active assistance and subjective recommendations, thus behaving more like human agents. For example, an intelligent dialogue system may suggest which airline is a better choice, considering cost, flight duration, take-off time, available seats, etc.; or suggest which digital camera is the most popular among teenagers or highest rated by professional photographers; or which restaurant is a perfect spot for a semi-formal business meeting or a romantic date.

Luckily, there are enormous amounts of reviews published by general users on the web every day. These are perfect resources for providing subjective recommendations and collective opinions. If there exists a systematic framework that harvests these reviews from general users, extracts the essence from the reviews and presents it appropriately in human-computer conversations, then we can enable dialogue systems to behave like a human shopping assistant, a travel agent, or a local friend who tells you where to find the best restaurant.

Summarization from online reviews, therefore, plays an important role for such dialogue systems. There have been previous studies on review analysis for text-based summarization systems (Mei et al., 2007; Titov and McDonald, 2008a; Branavan et al., 2008). Mixture models and topic models are used to predict the underlying topics of each document and generate a phrase-level summary. An aspect rating on each facet is also automatically

learned with statistical models (Snyder and Barzilay, 2007; Titov and McDonald, 2008b; Baccianella et al., 2009). These approaches are all very effective, and the review databases generated are well presented.

So the first thought for developing a recommendation dialogue system is to use such a categorized summary in a table-lookup fashion. For example, a dialogue system for restaurant recommendations can look up a summary table as exemplified in Table 1, and generate a response utterance from each row: "Restaurant A has good service and bad food; restaurant B has good service and good food; restaurant C has great service and nice atmosphere; restaurant D has poor service and reasonable price."

Restaurant	Summary
A	Good service, bad food,
B	Good service, good food
C	Great service, nice atmosphere
D	Poor service, reasonable price

Table 1. A partial table of categorization-based review summaries.

Such a dialogue system is, however, not very informative. First of all, there is too much redundancy. Long utterances repeated in the same pattern on the same topic are quite boring, and the information density is very low. Second, such a summary is too coarse-grained to be helpful. A user querying a restaurant recommendation system expects more *fine-grained* information such as house specials, wine selections and choices on desserts rather than just general 'good food.'

In contrast to a 'text' summarization system, the textual space in a dialogue turn is often very limited. Speech is inherently serial, and it cannot be skipped and scanned easily. A dialogue system which speaks long diatribes in each single conversation turn would likely not be well received. We believe that the best unit of review summary for dialogue purposes is short succinct phrases (e.g., 'amazing sauce' and 'romantic ambiance') which are catalogued into categories to answer *high-level questions* such as "How is the *food* at this restaurant?" or "What about the *atmosphere*?" Also, domain-specific phrases are greatly needed to generate correct answers to *specific queries*, such as "Which restaurants have *fresh sushi*?" and "Show me restaurants that serve *fantastic cocktails*."

Thus, the generally used correlated topic models, although very effective in text-based systems, are not quite suitable for interactive dialogue systems. The missing piece is an *HCI-oriented* (*human computer interaction*), *fine-grained*, *informative* yet *concise* review summarization. A good recommendation system should be able to provide sufficient yet specific information to help users choose a restaurant, a movie or a consumer product. Figure 1 shows an example of a conversation with a good recommendation system, which 1) provides a *generalized yet high-density* review summary in *human-friendly dialogue*; 2) provides both a *coarse-grained* summary (e.g., 'authentic food') and *fine-grained* information such as house specialties.

U: Are there any French restaurants in Cambridge?
S: There are *five French restaurants* in Cambridge. *Most of them are on Backbay Street. And most of them have attentive service.*
U: Show me the ones that have nice ambiance.
S: *Two of them have nice ambiance*, Pierrot Bistrot Francais and Coriander.
U: How about the food?
S: *Pierrot Bistrot Francais has authentic food, fresh salad, and absolutely delicious spaghetti carbonara. The food at Coriander is below average. It has very greasy chicken tikka masala and dry bread.*

Figure 1. Example of a conversation with a good recommendation dialogue system ('U' denotes the user and 'S' denotes the dialogue system. The responses to the user queries are produced by our system and the actual dialogue was an illustration of system capacities).

Therefore, the task of developing recommendation dialogue systems is decomposed into three problems: 1) how to extract *context-related* phrases, both *coarse-grained* and *fine-grained*, from online reviews; 2) how to select a *representative set* from the extracted phrases to create an *informative yet concise dialogue-oriented* summary database; 3) how to generate *human-friendly dialogue responses* from the review summary database.

To tackle these problems, we propose a *three-level framework*. In previous work (Liu and Seneff, 2009), we explored the first level by proposing a linguistic parse-and-paraphrase paradigm for re-

view phrase extraction. In this paper, we address the second problem: dialogue-oriented review summary generation. We propose an automatic approach to classifying high/low informative phrases using statistical models. Experiments conducted on a restaurant-domain dataset indicate that the proposed approach can predict phrase labels consistently with human judgment and can generate high-quality review summaries for dialogue purposes.

The rest of the paper is organized as follows: Section 2 gives an overview of the three-level framework for recommendation dialogue systems. In Section 3, we explain the proposed approach to dialogue-oriented review summary generation. Section 4 provides a systematic evaluation of the proposed approach, and Section 5 gives a further discussion on the experimental results. Section 6 summarizes the paper as well as pointing to future work.

2 System Overview

The three-level framework of a review-summary-based recommendation dialogue system is shown in Figure 2. The bottom level is linguistic phrase extraction. In previous work (Liu and Seneff, 2009), we employed a probabilistic lexicalized grammar to parse review sentences into a hierarchical representation, which we call a *linguistic frame*. From the linguistic frames, phrases are extracted by capturing a set of adjective-noun relationships. Adverbs and negations conjoined with the adjectives are also captured. We also calculated a numerical score for sentiment strength for each adjective and adverb, and further applied a cumulative offset model to assign a sentiment score to each phrase.

The approach relies on linguistic features that are independent of frequency statistics; therefore it can retrieve very rare phrases such as 'very greasy chicken tikka masala' and 'absolutely delicious spaghetti carbonara', which are very hard to derive from correlated topic models. Experimental results showed that the linguistic paradigm outperforms existing methods of phrase extraction which employ shallow parsing features (e.g., part-of-speech). The main contribution came from the linguistic frame, which preserves linguistic structure of a sentence by encoding different layers of semantic dependencies. This allows us to employ more sophisticated high-level linguistic features (e.g., long distance semantic dependencies) for phrase extraction.

However, the linguistic approach fails to distinguish highly informative and relevant phrases from uninformative ones (e.g., 'drunken husband', 'whole staff'). To apply these extracted phrases within a recommendation dialogue system, we have to filter out low quality or irrelevant phrases and maintain a concise summary database. This is the second level: dialogue-oriented review summary generation.

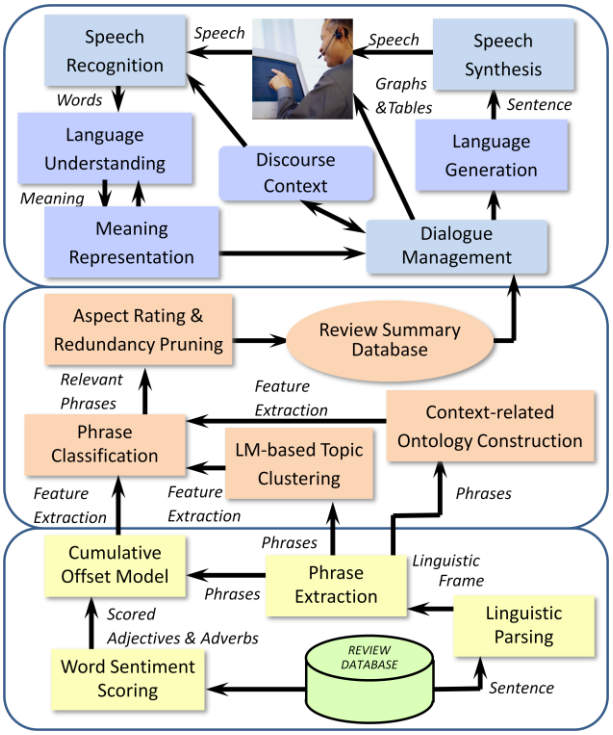

Figure 2. Three-level framework of review-based recommendation dialogue systems.

The standard of *highly informative and relevant* phrases is a very subjective problem. To gain insights on human judgment on this, the first two authors separately labeled a set of review-related phrases in a restaurant domain as 'good' and 'bad' summary phrases. We surveyed several subjects, all of whom indicated that, when querying a dialogue system for information about a restaurant, they care much more about special dishes served in this restaurant than generic descriptions such as 'good food.' This knowledge informed the annotation task: to judge whether a phrase delivered by a dialogue recommendation system would be help-

ful for users to make a decision. Surprisingly, although this is a difficult and subjective problem, the judgment from the two annotators is substantially consistent. By examining the annotations we observed that phrases such as 'great value' and 'good quality' are often treated as 'uninformative' as they are too common to be representative for a particular product, a restaurant or a movie. Phrases with neutral sentiment (e.g., 'green beans' and 'whole staff') are often considered as uninformative too. Phrases on specific topics such as house specialties (e.g., 'absolutely delicious spaghetti carbonara') are what the annotators care about most and are often considered as highly relevant, even though they may have only been seen once in a large database.

Driven by these criteria, from each phrase we extract a set of *statistical* features such as unigram/bigram probabilities and *sentiment* features such as sentiment orientation degree of the phrase, as well as underlying *semantic* features (e.g., whether the topic of the phrase fits in a domain-specific ontology). Classification models such as SVMs and decision tree algorithms are then trained on these features to automatically classify high/low informative phrases. Phrases identified as 'good' candidates are further pruned and catalogued to create concise summaries for dialogue purposes.

After generating the review summary database, the third level is to modify the response generation component in dialogue systems to create generalized and interactive conversations, as exemplified in Figure 1. The utterance from users is piped through speech recognition and language understanding. The meaning representation is then sent to the dialogue management component for *review-summary database lookup*. A response is then generated by the language generation component, and a speech utterance is generated by the synthesizer and sent back to the user. The dialogue system implementation is beyond the scope of this paper and will be discussed later in a separate paper.

3 Dialogue-oriented Review Summary Generation

Given an inquiry from users, the answer from a recommendation system should be helpful and relevant. So the first task is to identify a phrase as 'helpful' or not. The task of identifying a phrase as informative and relevant, therefore, is defined as a classification problem:

$$y = \bar{\theta} \cdot \bar{x} = \sum_{i=1}^{n} \theta_i x_i \qquad (1)$$

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

We employ statistical models such as SVMs (Joachims, 1998) and decision trees (Quinlan, 1986) to train the classification model. For model learning, we employ a feature set including *statistical features*, *sentiment features* and *semantic features.*

Generally speaking, phrases with neutral sentiment are less informative than those with strong sentiment, either positive or negative. For example, 'fried seafood appetizer', 'baked halibut', 'electronic bill' and 'red drink' do not indicate whether a restaurant is worth trying, as they did not express whether the fried seafood appetizer or the baked halibut are good or bad. Therefore, we take the sentiment score of each phrase generated from a cumulative offset model (Liu and Seneff, 2009) as a *sentiment feature*. Sentiment scores of phrases are exemplified in Table 2 (on a scale of 1 to 5).

Phrase	Sc.	Phrase	Sc.
really welcoming atmosphere	4.8	truly amazing flavor	4.6
perfect portions	4.4	very tasty meat	4.3
busy place	3.1	typical Italian restaurant	3.1
a little bit high price	2.2	pretty bad soup	1.8
sloppy service	1.8	absolute worst service	1.4

Table 2. Examples of sentiment scores of phrases.

We also employ a set of *statistical features* for model training, such as the unigram probability of the adjective in a phrase, the unigram probability of the 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. For example, phrases 'greasy chicken tikka masala' and 'drunken husband' have the same *n*-gram probabilities in our corpus (a single observation), but

they should certainly not be treated as the same. To capture the semantic meanings of phrases, we first cluster the topics of phrases into generic semantic categories. The language-model based algorithm is given by:

$$P(t_c|\, t_i) = \sum_{a\in A} P(t_c|a)\cdot P(a|t_i)$$
$$= \sum_{a\in A} \frac{P(a,t_c)}{P(a)}\cdot\frac{P(a,t_i)}{P(t_i)}$$
$$= \frac{1}{P(t_i)}\sum_{a\in A}\frac{1}{P(a)}\cdot P(a,t_c)\cdot P(a,t_i) \qquad (2)$$

where *A* represents the set of all the adjectives in the corpus. We select a small set of initial topics with the highest frequency counts (e.g., 'food', 'service' and 'atmosphere'). 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 bigram probability 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. We use this as a *semantic feature*, e.g., whether the topic of a phrase belongs to a generic semantic category. Table 3 gives some clustering examples.

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

Table 3. Topic to semantic category clustering.

This language-model-based method relies on bigram probability statistics and can well cluster highly frequent topics. Categories such as 'service' and 'atmosphere' contain very limited related topics, most of which have high frequencies (e.g., 'waiter', 'staff', 'ambiance' and 'vibe'). The category 'food', however, is very domain-specific and contains a very large vocabulary, from generic sub-categories such as 'sushi', 'dessert' and 'sandwich' as shown in the examples, to specific courses such as 'bosc pear bread pudding' and 'herb roasted vermont pheasant wine cap mushrooms'. These domain-specific topics have very low frequencies, yet they are very relevant and valuable. But many of them are discarded by the clustering. It would be a similar case in other domains. For example, consumer products, movies and books all have domain-independent semantic categories (e.g., 'price' and 'released date') and domain-specific categories (e.g., technical features of consumer products, casts of movies and authors of books).

To recover these context-relevant topics, we employ domain context relations such as a *context-related ontology*. A context-related ontology can be constructed from structured web resources such as online menus of restaurants, names of actors and actresses from movie databases, and specifications of products from online shops. An example of a partial online menu of a restaurant is shown in Figure 3. From these structured web resources, we can build up a hierarchical ontology, based on which a set of *semantic features* can be extracted (e.g., whether a phrase contains a course name, or an actress's name, or a dimension of technical features of a consumer product).

Entree
Roasted Pork Loin Wrapped In Bacon with watermelon and red onion salad spicy honey-mustard bbq sauce
Spicy Halibut And Clam Roast with bacon braised greens, white beans and black trumpet mushrooms
Parmesan and Caramelized Shallot Wrapper Style Ravioli turnip greens and white truffle oil
Herb Roasted Vermont Pheasant Wine Cap Mushrooms, Pearl Onions and Fava Beans
Dessert
Chocolate Tasting Plate of white chocolate bombe milk chocolate creme brulée and dark chocolate flourless cake
White Fruit Tasting Plate of warm apple strudel butterscotch, Bosc Pear bread pudding and toasted coconut panna cotta

Entrée	Pork loin, bacon, watermelon, red onion salad, honey, mustard, bbq sauce
Dessert	Chocolate, milk, crème brulee, cake

Figure 3. Example of a partial online menu and an exemplary ontology derived.

After the classification, phrases identified as 'highly informative and relevant' are clustered into different aspects according to the semantic category clustering and the hierarchical ontology. An average sentiment score for each aspect is then calculated:

$$ave(s_t) = \frac{\sum_{j\in N_s} r_j}{|N_s|} \qquad (3)$$

where s_t represents the aspect s of entry t (e.g., 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 in the cluster.

The set of phrases selected for one entry may come from several reviews on this single entry, and many of them may include the same noun (e.g., 'good fish', 'not bad fish' and 'above-average fish' for one restaurant). Thus, the next step is multi-phrase redundancy resolution. We select the phrase with a sentiment score closest to the average score of its cluster as the most representative phrase on each topic:

$$m = argmin_{j \in N_i}(|r_j - ave(s_t)|) \quad (4)$$

where $ave(s_t)$ represents the average sentiment score of aspect s, N_i represents the set of phrases on the same topic i in the cluster s, and r_j represents the sentiment score of phrase j.

This sequence of topic categorization, ontology construction, phrase pruning and redundancy elimination leads to a summary database, which can be utilized for dialogue generation in spoken recommendation systems. A review summary database entry generated by the proposed approaches is exemplified in Figure 4.

```
{ restaurant "dali restaurant and tapas bar"
   :atmosphere ( "wonderful evening", "cozy atmos-
                 phere", "fun decor", "romantic date" )
   :atmosphere_rating "4.1"
   :food ( "very fresh ingredients", "tasty fish",
         "creative dishes", "good sangria" )
   :food_rating "3.9"
   :service ( "fast service" )
   :service_rating "3.9"
   :general ("romantic restaurant","small space" )
   :general_rating "3.6"            }
```

Figure 4. Example of a review summary database entry generated by the proposed approaches.

4 Experiments

In this project, we substantiate the proposed approach in a restaurant domain for our spoken dialogue system (Gruenstein and Seneff, 2007), which is a web-based multimodal dialogue system allowing users to inquire about information about restaurants, museums, subways, etc. We harvested a data collection of 137,569 reviews on 24,043 restaurants in 9 cities in the U.S. from an online restaurant evaluation website[1]. From the dataset, 857,466 sentences were subjected to parse analysis; and a total of 434,372 phrases (114,369 unique ones) were extracted from the parsable subset (78.6%) of the sentences.

Most pros/cons consist of well-formatted phrases; thus, we select 3,000 phrases extracted from pros/cons as training data. To generate a human judgment-consistent training set, we manually label the training samples with 'good' and 'bad' labels. We then randomly select a subset of 3,000 phrases extracted from review texts as the test set and label the phrases. The kappa agreement between two sets of annotations is 0.73, indicating substantial consistency. We use the two annotation sets as the ground truth.

To extract context-related semantic features, we collect a large pool of well-formatted menus from an online resource[2], which contains 16,141 restaurant menus. Based on the hierarchical structure of these collected menus, we build up a context-related ontology and extract a set of semantic features from the ontology, such as whether the topic of a phrase is on *category-level* (e.g., 'entrée', 'dessert', 'appetizers', 'salad'), whether the topic is on *course-level* (e.g., 'Roasted Pork Loin', 'Spicy Halibut and Clam Roast'), and whether the topic is on *ingredient-level* (e.g., 'beans', 'chicken', 'mushrooms', 'scallop').

We employ the three types of features as aforementioned to train the SVMs and the decision tree models. To select the most valuable features for model training, we conducted a set of leave-one-feature-out experiments for both the SVMs and the decision tree 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, which makes the adjective unigram probability fail to become a dominant factor 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 noun in the phrase. Thus, we eliminate this feature for both the SVMs and the decision tree learning.

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

To evaluate the performance of the classification models, we take a set of intuitively motivated heuristic rules as the baseline. Figure 5 gives the pseudo-code of the heuristic rule algorithm, which uses variations of all the features except the unigram probability of adjectives.

```
If(sentiment score of the phrase exists)
  if(sentiment score is within neutral range) label=-1;
  else
     if(phrase appeared in the training data)
        if((3<frequency of phrase < 100))   label = 1;
        else
           if(frequency of phrase >= 100)   label = -1;
           else    if(topic belongs to ontology)  label = 1;
                   else   label = -1;
      else
         if(topic belongs to ontology)   label = 1;
         else   label = -1;
else
   if(phrase appeared in the training data)
        if((3<frequency of phrase < 100))
             if(topic belongs to ontology)  label = 1;
              else   label = -1;
        else
           if(frequency of phrase >= 100)   label = -1;
           else
              if(topic belongs to ontology)  label = 1;
              else   if(frequency of noun > 100) label = 1;
                     else   label = -1;
   else
       if(topic belongs to ontology)  label = 1;
       else     if(frequency of noun > 100)   label = 1;
                else   label = -1;
```

Figure 5. Pseudo-code of the heuristic rule algorithm.

The performance of classification by different models is shown in Table 4. Although the heuristic rule algorithm is complicated and involves human knowledge, the statistical models trained by SVMs and the decision tree algorithms both outperform the baseline significantly. 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, which is quite comparable to human judgment (the precision of one annotation set based on the other is 74%). This shows that the decision tree model can predict phrase labels as reliably as human judgment.

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

Table 4. Precision of phrase classification using the heuristic rule baseline, the SVM model, and the decision tree algorithm.

To gain further insight on the contributions of each feature set to the decision tree learning, Table 5 gives the experimental results on leaving each feature out of model training. As shown, without semantic features, the precision is 70.6% and 65.4% on the two annotation sets, lower by 7.3% and 9.1% than the case of training the model with all the features (77.9% and 74.5%). This shows that the semantic features significantly contribute to the decision tree learning.

Feature set	**A1**	**A2**
all features	**77.9%**	**74.5%**
without bigram probability of adjective-noun pair	56.6% (-21.3%)	63.9% (-10.6%)
without unigram probability of the phrase	57.6% (-20.3%)	64.3% (-10.2%)
without unigram probability of the noun	59.8% (-18.1%)	67.8% (-6.7%)
without sentiment score of the phrase	63.4% (-14.5%)	66.6% (-7.9%)
without underlying semantic features	70.6% (-7.3%)	65.4% (-9.1%)

Table 5. Performance of the decision tree model by leaving each feature out of model training ('A1' and 'A2' represent the annotation set 1 and 2 respectively).

The experimental results also show that the feature of bigram probability of the adjective-noun pair contributes the most to the model learning. Without this feature, the precision drops by 21.3% and 10.6%, reaching the lowest precision among all the leave-one-out experiments. This confirms our observation that although a single adjective is not dominant, the pair of the adjective and the noun that co-occurs with it plays an important role in the classification.

The sentiment of phrases also plays an important role. Without sentiment features, the precision

drops to 63.4% and 66.6% respectively on the two annotations, decreasing by 14.5% and 7.9%. This shows that the sentiment features contribute significantly to the classification.

5 Discussions

Experimental results show that the decision tree algorithm outperforms the SVMs on this particular classification problem, and it outperforms the heuristic rule baseline significantly. Thus, although the identification of informativeness and relevance of phrases is a rather subjective problem, which is difficult to predict using only human knowledge, it can be well defined by decision trees. 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 to a 'hierarchical *if-else* decision problem' in human cognition, where decision tree algorithms can fit well. Figure 6 shows a partial simplified decision tree learned from our model, which can give an intuitive idea of the decision tree models.

6 Related Work

Sentiment classification and opinion mining have been well studied for years. Most studies have focused on text-based systems, such as document-level sentiment classification and sentence-level opinion aggregation (Turney, 2002; Pang et al., 2002; Dave et al., 2003; Hu and Liu, 2004; Popescu and Etzioni, 2005; Wilson et al., 2005; Zhuang et al., 2006; Kim and Hovy, 2006).

There was a study conducted by Carenini et al. in 2006, which proposed a combination of a sentence extraction-based approach and a language generation-based approach for summarizing evaluative arguments. In our work, we utilize a lower-level phrase-based extraction approach, which utilizes high level linguistic features and syntactic structure to capture phrase patterns.

There was also a study on using reviews to generate a dictionary of mappings between semantic representations and realizations of concepts for dialogue systems (Higashinaka et al., 2006; Higashinaka, 2007). They also used the association between user ratings and reviews to capture semantic-syntactic structure mappings. A set of filtering rules was manually created to eliminate low-quality mappings. In our approach, we use an automatic approach to classifying high/low informative phrases. The learning features are domain-independent with no hand-crafted rules, and can be extended to other domains effortlessly.

7 Conclusions

In this paper we proposed a three-level framework for review-based recommendation dialogue systems, including linguistic phrase extraction, dialogue-oriented review summary generation, and human-friendly dialogue generation. The contributions of this paper are three-fold: 1) it identified and defined the research goal of utilizing opinion summarization for real human-computer conversation; 2) it formulated an evaluation methodology for high-density review summary for dialogue purposes; 3) it proposed an approach to automatic classification of high/low informative phrases using a decision tree model. Experimental results showed that the decision tree model significantly outperforms a heuristic rule baseline and the SVM model, and can resolve the phrase classification problem comparably to humans consistently.

Future work will focus on: 1) applying the sentiment scoring model to noun/verb sentiment assessment; 2) application of the review summary generation approach in other domains and other languages; 3) data collection on user engagement with our dialogue systems involving review-summary evaluation.

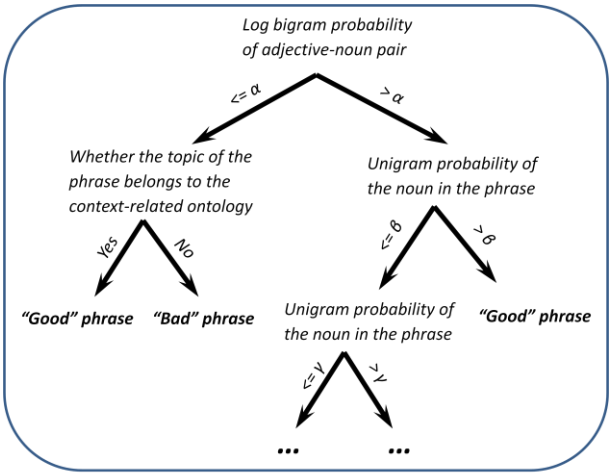

Figure 6. A partial simplified decision tree learned from our model.

References

Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. 2009. Multi-facet Rating of Product Reviews. In *Proceedings of European Conference on Information Retrieval*.

S.R.K. Branavan, Harr Chen, Jacob Eisenstein, and Regina Barzilay. 2008. Learning document-level semantic properties from free-text annotations. In *Proc. of ACL*.

Giuseppe Carenini, Raymond Ng, and Adam Pauls. 2006. Multi-Document Summarization of Evaluative Text. In *Proceedings of the Conference of the European Chapter of the Association for Computational Linguistics*.

Kushal Dave, Steve Lawrence, and David M. Pennock. 2003. Mining the peanut gallery: opinion extraction and semantic classification of product reviews. In *Proceedings of the International Conference on World Wide Web*.

W. Eckert, T. Kuhn, H. Niemann, S. Rieck, A. Scheuer, and E. G. Schukat-talamazzini. 1993. A Spoken Dialogue System for German Intercity Train Timetable Inquiries. In *Proc. European Conf. on Speech Technology*.

Alexander Gruenstein and Stephanie Seneff. 2007. Releasing a Multimodal Dialogue System into the Wild: User Support Mechanisms. *In Proceedings of the 8th SIGdial Workshop on Discourse and Dialogue, Antwerp, pages 111-119.*

A. L. Gorin, G. Riccardi and J. H. Wright. 1997. "How may I help you?" *Speech Communication*, vol. 23, pp. 113–127.

Ryuichiro Higashinaka, Rashmi Prasad and Marilyn Walker. 2006. Learning to Generate Naturalistic Utterances Using Reviews in Spoken Dialogue Systems. *In Proceedings of COLING-ACL*.

Ryuichiro Higashinaka, Marilyn Walker and Rashmi Prasad. 2007. An Unsupervised Method for Learning Generation Dictionaries for Spoken Dialogue Systems by Mining User Reviews. *Journal of ACM Transactions on Speech and Language Processing*.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the 2004 ACM SIGKDD international conference on Knowledge Discovery and Data mining*.

S.M. Kim and E.H. Hovy. 2006. Identifying and Analyzing Judgment Opinions. In *Proc. of HLT/NAACL*.

Jingjing Liu and Stephanie Seneff. 2009. Review Sentiment Scoring via a Parse-and-Paraphrase Paradigm. *In proceedings of EMNLP*.

Qiaozhu Mei, Xu Ling, Matthew Wondra, Hang Su, and ChengXiang Zhai. 2007. Topic Sentiment Mixture: Modeling Facets and Opinions in Weblogs. In *Proc. of WWW*.

Bo Pang, Lillian Lee, and S. Vaithyanathan. 2002. Thumbs up? Sentiment classification using machine learning techniques. In *Proceedings of EMNLP*.

A.M. Popescu and O. Etzioni. 2005. Extracting product features and opinions from reviews. In *Proceedings of EMNLP*.

JR Quinlan, 1986. Induction of decision trees. *Machine learning*, Springer-Netherlands.

A. Raux, B. Langner, A. Black, and M. Eskenazi. 2003. LET'S GO: Improving Spoken Dialog Systems for the Elderly and Non-natives. In *Proc. Eurospeech*.

Benjamin Snyder and Regina Barzilay. 2007. Multiple Aspect Ranking using the Good Grief Algorithm. In *Proceedings of NAACL-HLT.*

Ivan Titov and Ryan McDonald. 2008a. Modeling Online Reviews with Multi-Grain Topic Models. *In Proc. of WWW.*

Ivan Titov and Ryan McDonald. 2008b. A Joint Model of Text and Aspect Ratings for Sentiment Summarization. In *Proceedings of the Annual Conference of the Association for Computational Linguistics*.

Peter D. Turney. 2002. Thumbs up or thumbs down? Sentiment orientation applied to unsupervised classification of reviews. In *Proceedings of the Annual Conference of the Association for Computational Linguistics*.

T. Joachims. 1998. Text categorization with support vector machines: Learning with many relevant features. In *Proc. of ECML*, p. 137–142.

T. Wilson, J. Wiebe, and P. Hoffmann. 2005. Recognizing Contextual Polarity in Phrase-Level Sentiment Analysis. *In Proc. of HLT/EMNLP*.

Victor Zue, Stephanie Seneff, James Glass, Joseph Polifroni, Christine Pao, Timothy J. Hazen, and Lee Hetherington. 2000. JUPITER: A Telephone-Based Conversational Interface for Weather Information. In *IEEE Transactions on Speech and Audio Processing*, Vol. 8 , No. 1.

Li Zhuang, Feng Jing, and Xiao-Yan Zhu. 2006. Movie review mining and summarization. In *Proceedings of the 15th ACM international conference on Information and knowledge management.*