Intensional Summaries as Cooperative Responses in Dialogue: Automation and Evaluation

Joseph Polifroni
Nokia Research Center
3 Cambridge Center
Cambridge, MA 02142, USA
joseph.polifroni@nokia.com

Marilyn Walker
Department of Computer Science
University of Sheffield
Sheffield, S1 4DP, United Kingdom
M.A.Walker@sheffield.ac.uk

Abstract

Despite its long history, and a great deal of research producing many useful algorithms and observations, research in cooperative response generation has had little impact on the recent commercialization of dialogue technologies, particularly within the spoken dialogue community. We hypothesize that a particular type of cooperative response, *intensional summaries*, are effective for when users are unfamiliar with the domain. We evaluate this hypothesis with two experiments with CRUISER, a DS for in-car or mobile users to access restaurant information. First, we compare CRUISER with a baseline system-initiative DS, and show that users prefer CRUISER. Then, we experiment with four algorithms for constructing *intensional summaries* in CRUISER, and show that two summary types are equally effective: summaries that maximize domain coverage and summaries that maximize utility with respect to a user model.

U1:	Tell me about restaurants in London.
SI1:	What kind of cuisine are you interested in?
C1:	I know of 596 restaurants in London. I know of 3 inexpensive vegetarian restaurants and 14 inexpensive Chinese restaurants. I also know of 4 inexpensive Greek restaurants.
U2:	Chinese.
SI2:	Do you have a price range in mind?
C2:	I know of 27 restaurants in London that serve Chinese food. There are 8 inexpensive Chinese restaurants in Chinatown and 2 inexpensive Chinese restaurants in Hampstead/Kilburn. I also know of 1 inexpensive Chinese restaurant in Soho.
U3:	How about a cheap one?
SI3:	What neighborhood would you like?
C3:	I know of 1 inexpensive Chinese restaurant in Hampstead/Kilburn with very good food quality and 1 in Bayswater with good food quality. I also know of 2 in Chinatown with medium food quality.

Figure 1: Intensional summaries (C = CRUISER) as compared with a *system initiative* (SI) strategy in the London restaurant domain. U = User

1 Introduction

Research in cooperative response generation has had a long history and produced many useful algorithms and observations (Mays 1980; Pollack et al.1982; Joshi et al., 1986; Kalita et al., 1986) *inter alia*. However, it has had little impact on the recent commercialization of dialogue technologies, particularly within the spoken dialogue community. We believe that this lack of interest in cooperative response generation arises from two limitations of previous work: (1) There has been relatively little empirical research showing that cooperative responses lead to more natural, effective, or efficient dialogues (Litman et al.1998; Demberg and Moore, 2006); and (2) Previous work has hand-crafted such responses, or hand-annotated the database to support them (Kaplan, 1984; Kalita et al., 1986; Cholvy, 1990; Polifroni et al., 2003; Benamara, 2004), which has made it difficult to port and scale these algorithms.

Moreover, we believe that there is an even greater need today for cooperative response generation. Larger and more complex datasets are daily being created on the Web, as information

Proceedings of ACL-08: HLT, pages 479–487,
Columbus, Ohio, USA, June 2008.

is integrated across multiple sites and vendors. Many users will want to access this information from a mobile device and will have little knowledge of the domain. We hypothesize that these users will need cooperative responses that select and generalize the information provided.

In particular, we hypothesize that a particular type of cooperative response, *intensional summaries*, when provided incrementally during a dialogue, are effective for large or complex domains, or when users are unfamiliar with the domain. These intensional summaries have the ability to describe the data that forms the knowledge base of the system, as well as relationships among the components of that database. We have implemented *intensional summaries* in CRUISER (Cooperative Responses Using Intensional Summaries of Entities and Relations), a DS for in-car or mobile users to access restaurant information (Becker et al.2006; Weng et al.2005; Weng et al.2006). Figure 1 contrasts our proposed *intensional summary* strategy with the *system initiative* strategy used in many dialogue systems (Walker et al., 2002; VXML, 2007).

Previous research on cooperative responses has noted that summary strategies should vary according to the context (Sparck Jones, 1993), and the interests and preferences of the user (Gaasterland et al., 1992; Carenini and Moore, 2000; Demberg and Moore, 2006). A number of proposals have emphasized the importance of making generalizations (Kaplan, 1984; Kalita et al., 1986; Joshi et al., 1986). In this paper we explore different methods for constructing intensional summaries and investigate their effectiveness. We present fully automated algorithms for constructing intensional summaries using knowledge discovery techniques (Acar, 2005; Lesh and Mitzenmacher, 2004; Han et al., 1996), and decision-theoretic user models (Carenini and Moore, 2000).

We first explain in Sec. 2 our fully automated, domain-independent algorithm for constructing intensional summaries. Then we evaluate our intensional summary strategy with two experiments. First, in Sec. 3, we test the hypothesis that users prefer summary responses in dialogue systems. We also test a refinement of that hypothesis, i.e., that users prefer summary type responses when they are unfamiliar with a domain. We compare several versions of CRUISER with the system-initiative strategy, exemplified in Fig. 1, and show that users prefer CRUISER. Then, in Sec. 4, we test four different algorithms for constructing *intensional summaries*, and show in Sec. 4.1 that two summary types are equally effective: summaries that maximize domain coverage and summaries that maximize utility with respect to a user model. We also show in Sec. 4.2 that we can predict with 68% accuracy which summary type to use, a significant improvement over the majority class baseline of 47%. We sum up in Sec. 5.

2 Intensional Summaries

This section describes algorithms which result in the four types of intensional summaries shown in Fig. 2. We first define *intensional summaries* as follows. Let D be a domain comprised of a set R of database records $\{r_i, ...r_n\}$. Each record consists of a set of attributes $\{A_j, ..., A_n\}$, with associated values v: $D(A_i)$=$\{v_{i,1}, v_{i,2}, ..., v_{i,n}\}$. In a dialogue system, a constraint is a value introduced by a user with either an explicit or implied associated attribute. A constraint c is a function over records in D such that $c_j(R)$ returns a record r if $r \subseteq D$ and $r : A_i = c$. The set of all dialogue constraints $\{c_i, ..., c_n\}$ is the *context* C at any point in the dialogue. The set of records R in D that satisfy C is the *focal information*: R is the *extension* of C in D. For example, the attribute *cuisine* in a restaurant domain has values such as "French" or "Italian". A user utterance instantiating a constraint on cuisine, e.g., "I'm interested in Chinese food", results in a set of records for restaurants serving Chinese food. *Intensional summaries* as shown in Fig. 2 are descriptions of the focal information, that highlight particular subsets of the focal information and make generalizations over these subsets.

The algorithm for constructing intensional summaries takes as input the focal information R, and consists of the following steps:

- Rank attributes in context C, using one of two ranking methods (Sec. 2.1);

Type	Ranking	#atts	Clusters	Scoring	Summary
Ref-Sing	Refiner	3	Single value	Size	*I know of 35 restaurants in London serving Indian food. All price ranges are represented. Some of the neighborhoods represented are Mayfair, Soho, and Chelsea. Some of the nearby tube stations are Green Park, South Kensington and Piccadilly Circus.*
Ref-Assoc	Refiner	2	Associative	Size	*I know of 35 restaurants in London serving Indian food. There are 3 medium-priced restaurants in Mayfair and 3 inexpensive ones in Soho. There are also 2 expensive ones in Chelsea.*
UM-Sing	User model	3	Single value	Utility	*I know of 35 restaurants in London serving Indian food. There are 6 with good food quality. There are also 12 inexpensive restaurants and 4 with good service quality.*
UM-Assoc	User model	2	Associative	Utility	*I know of 35 restaurants in London serving Indian food. There are 4 medium-priced restaurants with good food quality and 10 with medium food quality. There are also 4 that are inexpensive but have poor food quality.*

Figure 2: Four intensional summary types for a task specifying restaurants with Indian cuisine in London.

- Select top-N attributes and construct clusters using selected attributes (Sec. 2.2);
- Score and select top-N clusters (Sec. 2.3);
- Construct frames for generation, perform aggregation and generate responses.

2.1 Attribute Ranking

We explore two candidates for attribute ranking: User model and Refiner.

User model: The first algorithm utilizes decision-theoretic user models to provide an attribute ranking specific to each user (Carenini and Moore, 2000). The database contains 596 restaurants in London, with up to 19 attributes and their values. To utilize a user model, we first elicit user ranked preferences for domain attributes. Attributes that are unique across all entities, or missing for many entities, are automatically excluded, leaving six attributes: *cuisine, decor quality, food quality, price, service*, and *neighborhood*. These are ranked using the SMARTER procedure (Edwards and Barron, 1994). Rankings are converted to weights (w) for each attribute, with a formula which guarantees that the weights sum to 1:

$$w_k = \frac{1}{K} \sum_{i=k}^{K} \frac{1}{i}$$

where K equals the number of attributes in the ranking. The absolute rankings are used to select attributes. The weights are also used for cluster scoring in Sec. 2.3. User model ranking is used to produce **UM-Sing** and **UM-Assoc** in Fig. 2.

Refiner method: The second attribute ranking method is based on the Refiner algorithm for summary construction (Polifroni et al., 2003). The Refiner returns values for every attribute in the focal information in frames ordered by frequency. If the counts for the top-N (typically, 4) values for a particular attribute, e.g., *cuisine*, exceeded M% (typically 80%) of the total counts for all values, then that attribute is selected. For example, 82% of Indian restaurants in the London database are in the neighborhoods Mayfair, Soho, and Chelsea. *Neighborhood* would, therefore, be chosen as an attribute to speak about for Indian restaurants. The thresholds M and N in the original Refiner were set *a priori*, so it was possible that no attribute met or exceeded the thresholds for a particular subset of the data. In addition, some entities could have many unknown values for some attributes.

Thus, to insure that all user queries result in some summary response, we modify the Refiner

method to include a ranking function for attributes. This function favors attributes that contain fewer unknown values but always returns a ranked set of attributes. Refiner ranking is used to produce **Ref-Sing** and **Ref-Assoc** in Fig. 2.

2.2 Subset Clustering

Because the focal information is typically too large to be enumerated, a second parameter attempts to find interesting clusters representing subsets of the focal information to use for the content of intensional summaries. We assume that the *coverage* of the summary is important, i.e., the larger the cluster, the more general the summary.

The simplest algorithm for producing clusters utilizes a specified number of the top-ranked attributes to define a cluster. Single attributes, as in the **Ref-Sing** and **UM-Sing** examples in Fig. 2, typically produce large clusters. Thus one algorithm uses the top three attributes to produce clusters, defined by either a single value (e.g., **UM-Sing**) or by the set of values that comprise a significant portion of the total (e.g., **Ref-Sing**).

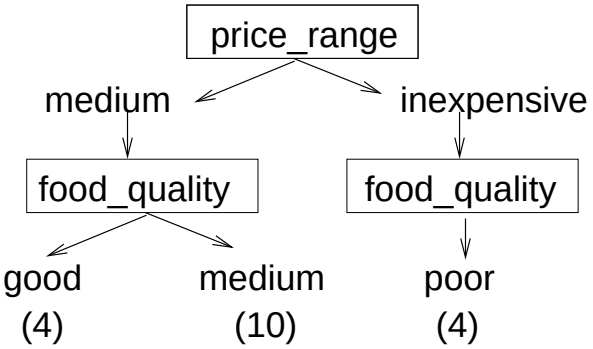

Figure 3: A partial tree for Indian restaurants in London, using price range as the predictor variable and food quality as the dependent variable. The numbers in parentheses are the size of the clusters described by the path from the root.

However, we hypothesize that more informative and useful intensional summaries might be constructed from clusters of discovered *associations* between attributes. For example, associations between *price* and *cuisine* produce summaries such as *There are 49 medium-priced restaurants that serve Italian cuisine.* We apply c4.5 decision tree induction to compute associations among attributes (Kamber et al., 1997; Quinlan, 1993). Each attribute in turn is designated as the dependent variable, with other attributes used as predictors. Thus, each branch in the tree represents a cluster described by the attribute/value pairs that predict the leaf node. Fig. 3 shows clusters of different sizes induced from Indian restaurants in London. The cluster size is determined by the number of attributes used in tree induction. With two attributes, the average cluster size at the leaf node is 60.4, but drops to 4.2 with three attributes. Thus, we use two attributes to produce associative clusters, as shown in Fig. 2 (i.e., the **Ref-Assoc** and **UM-Assoc** responses), to favor larger clusters.

2.3 Cluster Scoring

The final parameter scores the clusters. One scoring metric is based on cluster size. Single attributes produce large clusters, while association rules produce smaller clusters.

The second scoring method selects clusters of high utility according to a user model. We first assign scalar values to the six ranked attributes (Sec. 2.1), using clustering methods as described in (Polifroni et al., 2003) The weights from the user model and the scalar values for the attributes in the user model yield an overall utility U for a cluster h, similar to utilities as calculated for individual entities (Edwards and Barron, 1994; Carenini and Moore, 2000):

$$U_h = \sum_{k=1}^{K} w_k(x_{hk})$$

We use cluster size scoring with Refiner ranking and utility scoring with user model ranking. For conciseness, all intensional summaries are based on the three highest scoring clusters.

2.4 Summary

The algorithms for attribute selection and cluster generation and scoring yield the four summary types in Table 2. Summary **Ref-Sing** is constructed using (1) the Refiner attribute ranking; and (2) no association rules. (The quantifier (e.g., *some, many*) is based on the cover-

age.) Summary **Ref-Assoc** is constructed using (1) the Refiner attribute ranking; and (2) association rules for clustering. Summary **UM-Sing** is constructed using (1) a user model with ranking as above; and (2) no association rules. Summary **UM-Assoc** is constructed using (1) a user model with ranking of price, food, cuisine, location, service, and decor; and (2) association rules.

3 Experiment One

This experiment asks whether subjects prefer intensional summaries to a baseline system-initiative strategy. We compare two types of intensional summary responses from Fig. 2, **Ref-Assoc** and **UM-Assoc** to system-initiative.

The 16 experimental subjects are asked to assume three personas, in random order, chosen to typify a range of user types, as in (Demberg and Moore, 2006). Subjects were asked to read the descriptions of each persona, which were available for reference, via a link, throughout the experiment.

The first persona is the *Londoner*, representing someone who knows London and its restaurants quite well. The Londoner persona typically knows the specific information s/he is looking for. We predict that the system-initiative strategy in Fig. 1 will be preferred by this persona, since our hypothesis is that users prefer intensional summaries when they are *unfamiliar* with the domain.

The second persona is the *Generic tourist* (GT), who doesn't know London well and does not have strong preferences when it comes to selecting a restaurant. The GT may want to *browse* the domain, i.e. to learn about the structure of the domain and retrieve information by recognition rather than specification (Belkin et al., 1994). We hypothesize that the **Ref-Assoc** strategy in Fig. 2 will best fit the GT, since the corresponding clusters have good domain coverage.

The third persona is the *UM tourist* (UMT). This persona may also want to *browse* the database, since they are unfamiliar with London. However, this user has expressed preferences about restaurants through a previous interaction. The UMT in our experiment is concerned with price and food quality (in that order), and prefers restaurants in Central London. After location, the UMT is most concerned with cuisine type. The intensional summary labelled **Um-Assoc** in Fig. 2 is based on this user model, and is computed from discovered associations among preferred attributes.

As each persona, subjects rate responses on a Likert scale from 1-7, for each of four dialogues, each containing between three and four query/response pairs. We do not allow tie votes among the three choices.

3.1 Experimental results

The primary hypothesis of this work is that users prefer summary responses in dialogue systems, without reference to the context. To test this hypothesis, we first compare Londoner responses (average rating 4.64) to the most highly rated of the two intensional summaries (average rating 5.29) for each query/response pair. This difference is significant ($df = 263, p < .0001$), confirming that over users prefer an intensional summary strategy to a system-initiative strategy.

Table 1 shows ratings as a function of persona and response type. Overall, subjects preferred the responses tailored to their persona. The Londoner persona signifcantly preferred Londoner over UMT responses ($df = 95, p < .05$), but not more than GT responses. This confirms our hypothesis that users prefer incremental summaries in dialogue systems. Further, it disconfirms our refinement of that hypothesis, that users prefer summaries only when they are unfamiliar with the domain. The fact that no difference was found between Londoner and GT responses indicates that GT responses contain information that is perceived as useful even when users are familiar with the domain.

The Generic Tourist persona also preferred the GT responses, significantly more than the Londoner responses ($df = 95, p < .05$), but not significantly more than the UMT responses. We had hypothesized that the optimal summary type for users completely new to a domain would describe attributes that have high coverage of the focal information. This hypothesis is disconfirmed by these findings, that indicate that user

	Response Type		
Persona	London	GT	UMT
London	5.02	4.55	4.32
GT	4.14	4.67	4.39
UM tourist	3.68	4.86	5.23

Table 1: Ratings by persona assumed. London = Londoner persona, GT = Generic tourist, UMT = User Model tourist

	User model	Refiner
Association rules	3.4	2.9
Single attributes	3.0	3.4
	User model	Refiner
Small dataset	3.1	3.4
Large dataset	3.2	2.9

Table 2: User ratings showing the interaction between clustering method, attribute ranking, and dataset size in summaries.

model information is helpful when constructing summaries for any user interested in browsing.

Finally, the UM Tourist persona overwhelmingly preferred UMT responses over Londoner responses ($df = 95, p < .0001$). However, UMT responses were not significantly preferred to GT responses. This confirms our hypothesis that users prefer summary responses when they are unfamiliar with the domain, but disconfirms the hypothesis that users will prefer summaries based on a user model. The results for both the Generic Tourist and the UM Tourist show that both types of intensional summaries contain useful information.

4 Experiment Two

The first experiment shows that users prefer intensional summaries; the purpose of the second experiment is to investigate what makes a good intensional summary. We test the different ways of constructing such summaries described in Sec. 2, and illustrated in Fig. 2.

Experimental subjects were 18 students whose user models were collected as described in Sec. 2.3. For each user, the four summary types were constructed for eight tasks in the London restaurant domain, where a task is defined by a query instantiating a particular attribute/value combination in the domain (e.g., *I'm interested in restaurants in Soho*). The tasks were selected to utilize a range of attributes. The focal information for four of the tasks (*large set tasks*) were larger than 100 entities, while the focal information for the other four tasks were smaller than 100 entities (*small set tasks*). Each task was presented to the subject on its own web page with the four intensional summaries presented as text on the web page. Each subject was asked to carefully read and rate each alternative summary response on a Likert scale of $1 \ldots 5$ in response to the statement, *This response contains information I would find useful when choosing a restaurant.* The subjects were also asked to indicate which response they considered the best and the worst, and to provide free-text comments about each response.

4.1 Hypothesis Testing Results

We performed an analysis of variance with attribute ranking (user model vs. refiner), clustering method (association rules vs. single attributes), and set size (large vs. small) as independent variables and user ratings as the dependent variable. There was a main effect for set size ($df = 1, f = 6.7, p < .01$), with summaries describing small datasets (3.3 average rating) rated higher than those for large datasets (3.1 average rating).

There was also a significant interaction between attribute ranking and clustering method ($df = 1, f = 26.8, p < .001$). Table 2 shows ratings for the four summary types. There are no differences between the two highest rated summaries: **Ref-Sing** (average 3.4) and **UM-Assoc** (average 3.4). See Fig. 2. This suggests that discovered associations provide useful content for intensional summaries, but only for attributes ranked highly by the user model.

In addition, there was another significant interaction between ranking method and setsize ($df = 1, f = 11.7, p < .001$). The ratings at the bottom of Table 2 shows that overall, users rate summaries of small datasets higher, but users rate summaries higher for large datasets when a user model is used. With small datasets, users prefer summaries that don't utilize user model information.

We also calculate the average utility for each response (Sec. 2.1) and find a strong correlation between the rating and its utility ($p < .005$). When considering this correlation, it is important to remember that utility can be calculated for all responses, and there are cases where the Refiner responses have high utility scores.

4.2 Summary Type Prediction

Our experimental data suggest that characteristics associated with the set of restaurants being described are important, as well as utility information derived from application of a a user model. The performance of a classifier in predicting summary type will indicate if trends we discovered among user judgements carry over to an automated means of selecting which response type to use in a given context.

In a final experiment, for each task, we use the highest rated summary as a class to be predicted using C4.5 (Quinlan, 1993). Thus we have 4 classes: **Ref-Sing**, **Ref-Assoc**, **UM-Sing**, and **UM-Assoc**. We derive two types of feature sets from the responses: features derived from each user model and features derived from attributes of the query/response pair itself. The five feature sets for the user model are:

- *umInfo*: 6 features for the rankings for each attribute for each user's model, e.g. a summary whose user had rated *food quality* most highly would receive a '5' for the feature *food quality*;
- *avgUtility*: 4 features representing an average utility score for each alternative summary response, based on its clusters (Sec. 2.3).
- *hiUtility*: 4 features representing the highest utility score among the three clusters selected for each response;
- *loUtility*: 4 features representing the lowest utility score among the three clusters selected for each response;
- *allUtility*: 12 features consisting of the high, low, and average utility scores from the previous three feature sets.

Three feature sets are derived from the query and response pair:

- *numRests*: 4 features for the coverage of each response. For summary **Ref-Assoc** in Table 2, *numRests* is 43; for summary **UM-Assoc**, *numrests* is 53.;
- *task*: A feature for the type of constraint used to generate the focal information (e.g., *cuisine, price range*).
- *dataset*: A feature for the size of the focal information subset (i.e., *big, small*), for values greater and less than 100.

Sys	Feature Sets	Acc(%)
S1	*allUtility*	47.1
S2	*task, numRests*	51.5
S3	*allUtility,umInfo*	62.3*
S4	*allUtility,umInfo,numRests,task*	63.2*
S5	*avgUtility,umInfo,numRests,task*	62.5*
S6	*hiUtility,umInfo,numRests,task*	66.9*
S7	*hiUtility,umInfo,numRests,task,dataset*	68.4*
S8	*loUtility,umInfo,numRests,task*	60.3*
S9	*hiUtility,umInfo*	64.0*

Table 3: Accuracy of feature sets for predicting preferred summary type. * = $p < .05$ as compared to the Baseline (S1)).

Table 3 shows the relative strengths of the two types of features on classification accuracy. The majority class baseline (System S1) is 47.1%. The S2 system uses only features associated with the query/response pair, and its accuracy (51.5%) is not significantly higher than the baseline. User model features perform better than the baseline (S3 in Table 3), and combining features from the query/response pair and the user model significantly increases accuracy in all cases. We experimented with using all the utility scores (S4), as well as with using just the average (S5), the high (S6), and the low (S8). The best performance (68.4%)is for the (S7) system combination of features.

The classification rules in Table 4 for the best system (S7) suggests some bases for users' decisions. The first rule is very simple, simply stating that, if the highest utility value of the **Ref-Sing** response is lower than a particular threshold, then use the **UM-Assoc** response. In other words, if one of the two highest scoring response types has a low utility, use the other.

The second rule in Table 4 shows the effect that the number of restaurants in the response has on summary choice. In this rule, the **Ref-Sing** response is preferred when the highest util-

```
IF (HighestUtility: Ref-Sing) < 0.18
      THEN USE UM-Assoc
---------------------------------------
IF (HighestUtility: Ref-Assoc) > 0.18) &&
      (NumRestaurants: UM-Assoc < 400) &&
         (HighestUtility: UM-Assoc < .47)
         THEN USE Ref-Sing
---------------------------------------
IF (NumRestaurants: UM-Assoc < 400) &&
      (HighestUtility: UM-Assoc < .57) &&
         (HighestUtility: Ref-Assoc > .2)
         THEN USE Ref-Assoc
```

Table 4: Example classification rules from System 7 in Table 3.

ity value of that response is over a particular threshold.

The final rule in Table 4 predicts **Ref-Assoc**, the lowest overall scoring response type. When the number of restaurants accounted for by **UM-Assoc**, as well as the highest utility for that response, are both below a certain threshold, and the highest utility for the **Ref-Assoc** response is above a certain threshold, then use **Ref-Assoc**. The utility for any summary type using the Refiner method is usually lower than those using the user model, since overall utility is not taken into account in summary construction. However, even low utility summaries may mention attributes the user finds important. That, combined with higher coverage, could make that summary type preferable over one constructed to maximize user model utility.

5 Conclusion

We first compared intensional summary cooperative responses against a system initiative dialogue strategy in CRUISER. Subjects assumed three "personas", a native Londoner, a tourist who was interacting with the system for the first time (GT), or a tourist for which the system has a user model (UMT). The personas were designed to reflect differing ends of the spectra defined by Belkin to characterize information-seeking strategies (Belkin et al., 1994). There was a significant preference for intensional summaries across all personas, but especially when the personas were unfamiliar with the domain. This preference indicates that the benefits of intensional summaries outweigh the increase in verbosity.

We then tested four algorithms for summary construction. Results show that intensional summaries based on a user model with association rules, or on the Refiner method (Polifroni et al., 2003), are equally effective. While (Demberg and Moore, 2006) found that their user model stepwise refinement (UMSR) method was superior to the Refiner method, they also found many situations (70 out of 190) in which the Refiner method was preferred. Our experiment was structured differently, but it suggests that, in certain circumstances, or within certain domains, users may wish to hear about choices based on an analysis of focal information, irrespective of user preferences.

Our intensional summary algorithms automatically construct summaries from a database, along with user models collected via a domain-independent method; thus we believe that the methods described here are domain-independent. Furthermore, in tests to determine whether a classifier can predict the best summary type to use in a given context, we achieved an accuracy of 68% as compared to a majority class baseline of 47%, using dialogue context features. Both of these results point hopefully towards a different way of automating dialogue design, one based on a combination of user modelling and an analysis of contextual information. In future work we hope to test these algorithms in other domains, and show that intensional summaries can not only be automatically derived but also lead to reduced task times and increased task success.

References

A.C. Acar and A. Motro. 2005. Intensional Encapsulations of Database Subsets via Genetic Programming. *Proc, 16th Int. Conf. on Database and Expert Systems Applications*. Copenhagen.

Tilman Becker, Nate Blaylock, Ciprian Gerstenberger, Ivana Kruijff-Korbayová, Andreas Korthauer, Manfred Pinkal, Michael Pitz, Peter Poller, and Jan Schehl. Natural and intuitive multimodal dialogue for in-car applications: The sammie system. In *ECAI*, pages 612–616, 2006.

N. J. Belkin, C. Cool, A. Stein and U. Thiel. 1994. Cases, Scripts, and Information Seeking Strategies: On the Design of Interactive Information Retrieval Systems. *Expert Systems and Applications*, 9(3):379–395.

F. Benamara. 2004. Generating Intensional Answers in Intelligent Question Answering Systems. *Proc. 3rd Int. Conf. on Natural Language Generation INLG*.

G. Carenini and J. Moore. 2000. A Strategy for Generating Evaluative Arguments. *Proc. First Int'l Conf. on Natural Language Generation*. 1307–1314.

Brant Cheikes and Bonnie Webber. Elements of a computational model of cooperative response generation. In *Proc. Speech and Natural Language Workshop*, pages 216–220, Philadelphia, 1989.

X. Chen and Y-F. Wu. 2006. Personalized Knowledge Discovery: Mining Novel Association Rules from Text. *Proc., SIAM Conference on Data Mining*.

L. Cholvy. 1990. Answering Queries Addressed to a Rule Base. *Revue d'Intelligence Artificielle*. 1(1):79–98.

V. Demberg and J. Moore. 2006 Information Presentation in Spoken Dialogue Systems. *Proc. 11th Conf. EACL.*.

W. Edwards and F. Hutton Barron. 1994. Smarts and smarter: Improved simple methods for multiattribute utility measurement. *Organizational Behavior and Human Decision Processes*. 60:306–325.

T. Gaasterland and P. Godfrey and J. Minker. 1992. An Overview of Cooperative Answering. *Journal of Intelligent Information Systems*. 1(2):387–416.

J. Han, Y. Huang and N. Cercone. 1996. Intelligent Query Answering by Knowledge Discovery Techniques. *IEEE Transactions on Knowledge and Data Engineering*. 8(3):373–390.

Aravind Joshi, Bonnie Webber, and Ralph M. Weischedel. Living up to expectations: computing expert responses. In *HLT '86: Proceedings of the workshop on Strategic computing natural language*, pages 179–189, Morristown, NJ, USA, 1986. Association for Computational Linguistics.

J. Kalita and M.J. Colburn and G. McCalla. 1984. A response to the need for summary responses. *COLING-84)*. 432–436.

M. Kamber, L. Winstone, W. Gong, S. Cheng and J Han. 1997. Generalization and decision tree induction: efficient classification in data mining. *Proc. 7th Int. Workshop on Research Issues in Data Engineering (RIDE '97)*. 111–121.

S.J.Kaplan. 1984. Designing a Portable Natural Language Database Query System. *ACM Transactions on Database Systems*, 9(1):1–19.

N. Lesh and M. Mitzenmacher. Interactive data summarization: an example application. *Proc., Working Conference on Advanced Visual Interfaces*. Gallipoli, Italy. pages 183–187.

Diane J. Litman, Shimei Pan, and Marilyn A. Walker. Evaluating response strategies in a web-based spoken dialogue agent. In *COLING-ACL*, pages 780–786, 1998.

J. Polifroni, G. Chung, and S. Seneff. 2003. Towards the Automatic Generation of Mixed-Initiative Dialogue Systems from Web Content. *Proc. Eurospeech*. 2721–2724.

E. Mays. Correcting misconceptions about database structure. In *Proceedings of the CSCSI '80*, 1980.

Martha E. Pollack, Julia Hirschberg, and Bonnie L. Webber. User participation in the reasoning processes of expert systems. In *AAAI*, pages 358–361, 1982.

J.R. Quinlan 1993. *C4.5: Programs for Machine Learning*. Morgan Kaufmann. San Mateo, CA.

K. Sparck Jones. 1998. Automatic summarising: factors and directions. I. Mani and M. Maybury, eds. *Advances in Automatic Text Summarization*. MIT Press.

M. Walker, A. Rudnicky, J. Aberdeen, E. Bratt, J. Garofolo, H. Hastie, A. Le, B. Pellom, A. Potamianos, R. Passonneau, R. Prasad, S. Roukos, G. Sanders, S. Seneff and D. Stallard. 2002. DARPA Communicator Evaluation: Progress from 2000 to 2001. *Proc, ICSLP 2002*.

F. Weng, L. Cavedon, B. Raghunathan, D. Mirkovic, H. Cheng, H. Schmidt, H. Bratt, R. Mishra, S. Peters, L. Zhao, S. Upson, E. Shriberg, and C. Bergmann. Developing a conversational dialogue system for cognitively overloaded drivers. In *Proceedings, International Congress on Intelligent Transportation Systems*, 2005.

F. Weng, S. Varges, B. Raghunathan, F. Ratiu, H. Pon-Barry, B. Lathrop, Q. Zhang, T. Scheideck, H. Bratt, K. Xu, M. Purver, R. Mishra, M. Raya, S. Peters, Y. Meng, L. Cavedon, and L. Shriberg. Chat: A conversational helper for automotive tasks. In *Proceedings, Interspeech: International Conference on Spoken Language Processing*, 2006.

Voxeo. VoiceXML Development Guide. http://voicexml.org.