Anticipating User Intentions in Customer Care Dialogue Systems

Anticipating User Intentions in Customer Care Dialogue Systems

Alessandro Mazzei♥ Luca Anselma♥ Manuela Sanguinetti♦ Amon Rapp♥
Dario Mana♣ Md Murad Hossain♥ Viviana Patti♥ Rossana Simeoni♣ Lucia Longo♣

♥Dipartimento di Informatica, Università degli Studi di Torino, Italy ♣TIM, Torino, Italy
♦Dipartimento di Matematica e Informatica, Università degli Studi di Cagliari, Italy

***Abstract*—In this work, we investigate the case of human-machine dialogues in the specific domain of commercial customer care. We built a corpus of conversations between users and a customer-care chatbot of an Italian telecom company, focusing on a sample of conversations where users contact the service asking for explanations about billing issues or overcharges. We observed that users' requests are often vague, generic or incomprehensible. In such cases, commercial dialogue systems typically ask for clarifications or further details to fully understand users' specific requests. However, from the corpus analysis it appeared that chatbot's clarifying requests may result in ineffective interactions, with users eventually giving up the conversation or switching to a human agent for a faster query resolution. A recovery strategy is thus needed to anticipate users' information needs, or intentions. We address this issue resorting to GEN-DS, a dialogue system based on symbolic data-to-text generation. GEN-DS analyzes the user-company contextual relational knowledge, with the aim to generate more relevant answers to unclear questions. In this paper, we describe the GEN-DS architecture along with the experiments we carried out to evaluate its output. Results from an offline human evaluation show significant improvements of GEN-DS compared to the original system. These improvements concern properties such as utility, necessity, understandability, and quickness of the information communicated in the dialogue. We believe that GEN-DS techniques may find application in all the dialogue systems that need to manage vague requests and must rely on relational knowledge.**

***Index Terms*—Natural Language Processing, Human-computer Interface, Man-Machine Systems.**

I. INTRODUCTION

Dialogue systems (DSs) may have different forms and pursue different aims. Grudin and Jacques [1] proposed a taxonomy of DSs based on their conversational focus: *virtual companions* usually engage on any topic and keep the conversation going, while *intelligent assistants* converse on any topic but aim to keep the conversation short; instead, *task-oriented agents* aim to solve specific problems and are addressed to perform short conversations.

An ever-increasing number of companies are adopting task oriented DSs as a preferred way of interacting with their customers. DSs provide several benefits to companies, to their customers, and to the customer care human operators (e.g., [2][3][4]). They are available 24/7 and can keep the context of an ongoing conversation for hours or even days, allowing users to solve their problems even when they get distracted from the support session. Moreover, DSs allow the collection, directly from customers, of unconstrained natural language text, which may then be interpreted through computational linguistics techniques [5]. They would thus provide an extremely valuable source of knowledge about customers' expectations, preferences, and behavior.

Customers commonly approach the DS with a variety of attitudes and expectations (e.g., [6], [7]), which are often not met by the technology. Users report a satisfying experience when the agent can correctly interpret their requests, provide appropriate and relevant responses, and communicate clearly what it can do [8]. Users holding high expectations toward the DS often approach it as if it were a human operator and explain their situation and problem at length, including details that give a lot of contextual information, but may not be directly useful (or usable) by an automatic system. It appears that these kinds of users get easily frustrated or angry, when the agent does not meet their assumptions or does not solve their problem in the way they desire [6]. This may lead them to close the chat.

In short, research highlights that users' expectations shape the interaction experience with the chatbot, influencing their overall satisfaction. Expectations may revolve around the chatbot's capabilities of correctly interpreting the user's intent, providing timely information, and giving appropriate and relevant responses. In particular, expectations of receiving explanations about issues that are relevant to the user (e.g., unusual situations related to subscribed services) are certainly fundamental for customer care.

Some approaches related to users' expectations focus on the development of systems that generate either more accurate chatbot's clarifying questions (similarly to conversational search systems, e.g., [9]), or tailored responses aimed at anticipating users' information needs, or intentions. It is worth pointing out that the term "intention" here does not denote the user's "purchase intention" (see [10]), nor the user's "intent", a term that in task-oriented dialogue systems specifically indicates a user's goal expressed in an utterance.

A recent work considered the application of end-to-end task-completion neural DSs for the specific task of booking cinema tickets [11]. The authors also analyzed the impact of the errors of the natural language understanding module.

Di Lascio et al. [12] emphasized that the linguistic knowledge provided by users is often not sufficient to fulfill their expectations. We believe that in these cases the DS should use the knowledge on the domain context to produce a better interaction by predicting user intentions.

In this article, we propose a solution for situations in which users ask for an explanation of a certain unusual situation regarding the services that they have subscribed to, an issue that remained unexplored in previous research. We collected and analyzed a corpus of almost 3,000 *customer conversations with a DS* using the log files of the DS of a telecommunication company (called *COM-DS*) and then selected the conversations where customers ask the DS for explanations. We thus found that about 5% of the total number of conversations from the corpus involves requests for explanations and 50% of this specific type of conversations is about additional or unexpected charges on the customers' telephone accounts. This is a particularly dangerous situation for the company, as this kind of issue may result in customer churn.

Leveraging content selection mechanisms and Natural Language Generation (NLG) techniques, a new DS (called *GEN-DS*) was developed to adequately address vague or ungrammatical requests for explanations regarding unrecognized charges on the users' telephone accounts. Unlike other systems such as [11], GEN-DS does not rely on a grammatical and comprehensible linguistic input.

GEN-DS considers the history of the transactions on the user phone balance and discriminates between transactions that are typical and transactions that are uncommon. Furthermore, the developed model can distinguish between transactions that have a relevant impact on users' phone account and those that are negligible from the economic point of view. An important feature in the design and construction of GEN-DS is that, when asked about an unrecognized charge on the phone account, it produces a synthetic and useful answer for the user, as opposed to a more straightforward, but long-to-read and less immediate, response listing all the recent transactions on the user's account.

Our belief is that the developed techniques can be useful in other contexts, whenever a user observes some unusual state of a product, service, or system and asks a DS for clarifications about it. For example, the same kind of solution can be employed for managing the point account of frequent flyers of an airline, when they observe some anomaly in their point balance and ask for an explanation. The same goes for accumulated discounts in stores, or whenever there is a need to manage a balance of points, money, or whatever is affected by customer-company transactions.

The paper is structured as follows. In Section II, we describe the corpus development, the design and implementation of GEN-DS, the design and execution of an experimentation with humans to evaluate GEN-DS. In Section III, we discuss the results of the experimentation and in Section IV we conclude the paper.

II. Materials and Methods

This section provides the main contributions of the paper. In Section II.A we describe the construction of a dialogue corpus concerning explanation requests in the customer-care domain. In Section II.B we provide formalization of the *evidence* notion to the customer transaction domain. In Section II.C we describe GEN-DS, a DS specifically designed for managing the request emerged from corpus analysis and that produces answers based on evidence. In Section II.D we provide a methodology for building experimental scenarios starting from real dialogues. Using these scenarios, in Section II.E we describe an experimentation with users designed to evaluate the quality impact of GEN-DS for the case of explanation requests in the customer care domain. Finally, in Section II.F we provide the results of the experimentations.

A. Building a Corpus of Explanation Requests in Customer-Care Dialogues Domain

For the purposes of this study, we first collected a sample corpus of conversations in Italian to find an empirical basis of our working hypothesis [13]. The supplementary material contains the corpus annotation. Due to the corporate privacy policy, the actual content was made available to the authors for the sole purpose of this research and cannot be publicly released.

The primary goal of the corpus analysis was to identify the main characteristics of these interactions, in terms of basic features – such as average number of turns per conversation and average turn length per user/agent – and to verify whether recurring linguistic behaviors could be found in customers' requests for explanations.

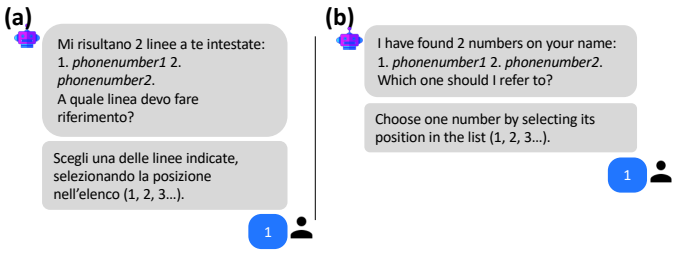


Fig. 1. Example of DS-customer interaction, with longer DS turns as opposed to more concise responses by the customer. On the left, the original dialogue excerpt in Italian, on the right its English translation.

As mentioned in Section I, the dataset consists of real dialogues between customers and the DS of an Italian telecommunication company (COM-DS). This was created by selecting from a sample held over 24 hours a reduced subset that included requests for explanations from customers, thus using a simple string-matching method that extracted all conversations where the strings *perché* ("why", along with its orthographical variations, e.g., *perchè* or *xkè*) and *come mai* ("how come") occurred in the users' messages. The resulting corpus consists of 142 dialogues, for a total amount of 1540 turns, where each turn consists of one or more messages from one party. The collection features an average of about 11 turns per dialogue, and an average length of 9 tokens in customer turns and 38 tokens in the agent turns. The average dialogue length reported for this corpus is not in line with past literature that showed how task-oriented dialogues are typically shorter

[27]. This could be explained by the fact that, especially in the case of the DS, a single turn most often consists of more than one message. Another striking difference is the average length of customers' messages compared to the ones from the chatbot. This difference is also due to the way the agent responses are currently structured; as a matter of fact, they usually include detailed information (for example, on invoice items or available options); conversely, customers' messages are generally more concise. In some cases, the latter are basic yes/no answers, or digits (1, 2, ...) corresponding to the options provided by the agent in its previous message (see the example in Fig. 1). These divergences often lead to loops in communication in which both the user and the chatbot find themselves repeating the same requests or statements several times, within the same conversation, or providing irrelevant information that do not contribute to achieving the goal of the conversation, which is to provide the user with a clear and exhaustive explanation.

The architecture of many commercial DSs relies on the assumption that some relevant information is provided by the user utterance [5]. However, the exploratory analysis of this corpus proved that this assumption is sometimes false or only partially true. In fact, linguistic input in users' messages can be vague (see Example 1 below–we provide a rough English translation trying to replicate the vagueness and ungrammatical nature of the Italian original utterances), sometimes ungrammatical or not easy to follow (Examples 2-3), or too long and confusing (Example 4). In all such cases, the dialogue manager might need to ask for additional clarifications or to access some contextual information to compensate the lack of linguistic information.

(1) *Perché mi sono stati scalati dei soldi.*
(Why has some money been deducted (from my account)).

(2) *Salve. Vorrei sapere perché ho pagato 0,50 cent. Per sms se li ho gratis E i 2,00 euro in piu per che cosa sono Grazie*
(Hello. I would like to know why I paid 0.50 cents. For texts if I have them for free And what about the additional 2,00 euros for what are they for Thanks.)

(3) *Bg come mai mi è addebitato altri euro ho qualche cosa attivato a pagamento*
(GM how come I was charged other euros I have something activated for a fee)

(4) *Come mai mi vengono addebitati costi di <serviceName> quando non è stato mai richiesto da me E come mai la bolletta è passata da 36 a 57 euro Ho già disdetto <serviceName> dai cellulari, mi sa che devo dare disdetta anche dal fisso poichè mi sento costantemente vessato e truffato dalla vostra compagnia. Inutile dire che è praticamente impossibile parlare con un operatore al telefono. Vergogna*
(Why am I being charged for <serviceName> when it has never been requested for And why the bill has gone from 36 to 57 euros I have already canceled <serviceName> from mobile phones, I guess I'll have to cancel it from the landline as well because I constantly feel harassed and cheated by your company. Needless to say that it is impossible to speak to an operator on the phone. Shame on you)

(5) *Scusami ma vorrei sapere come mai mi vengono fatti certi addebiti*
(Sorry but I'd like to know why there are some charges*)*

(6) *Salve vorrei sapere perchè mi sono stati presi 12€ invece che dieci dall'ultima ricarica*
(Hi I'd like to know why you charged 12€ instead of ten since last top-up)

(7) *Buongiorno, vorrei sapere perché ho il credito in negativo, nonostante abbia fatto una ricarica da 15€ proprio stamattina*
(Good morning, I'd like to know why I have a negative balance, despite I made a 15€ top-up just this morning)

Prior to the GEN-DS design and development, some annotation experiments were carried out on the corpus, with the aim to explore possible recurring patterns underlying the user-chatbot interactions (see [13] for a more detailed description of the annotation scheme). In this process, we also observed that users' explanation requests typically fall under three main categories of request, that we briefly define below.

Category I: (58% of the occurrences in the corpus) a charge in the account is claimed, but no further information is provided (see Examples 1, 3, 5).

Category II: (31% of the occurrences) the customer asks for an explanation about a charge providing vague information (Examples 2, 4, 6).

Category III: (11% of the occurrences) the customer asks for an explanation about a negative balance (Example 7).

The corpus analysis thus provided evidence of the fact that in customer care interactions user requests can be vague and not informative enough, and they can be identified with (at least) one of the major categories we described above. Considering this, we designed a new DS based on standard symbolic NLG techniques exploiting domain knowledge (see Section II.B), which can produce a response to the request (see Section II.C). We evaluated this DS based on the three categories identified in the corpus (see Section II.D).

B. Importance, Effect and Evidence in relational domain-context knowledge

The need to connect domain-specific data to factual linguistic explanations has drawn much attention in the recent past [15]. A key role in this task is played by *content selection*, which determines what kind of information should be communicated to the user. Symbolic, statistical, and neural approaches have been proposed for this task (see [16] for a recent neural approach reporting a detailed survey on the state of the art).

In this work, we adapt the approach to content selection proposed by Biran and McKeown [17], by formalizing the notions of *effect*, *importance* and *evidence* to the specific context of our study, i.e., the customers' transactions stored in a relational database. We define the latter as *domain context knowledge* (DC-knowledge henceforth, cf. Table II). The original proposal in Biran and McKeown's work considers statistical classifiers based on linear discriminant functions, as linear SVMs. The notion of *effect* is anchored to the weight of a feature in the classification into class y of a single data instance. In contrast, the notion of *importance* to the weight of a feature in the classification into a class y of all instances of the training set. The authors proposed to combine these two notions

in one single notion called *evidence.* They show that evidence can be used with the aim to select and order the features that should be communicated to the users, in an NLG system, for describing the trends of financial stock prices. In particular, importance values are narrative roles, that "*... represent semantically clear concepts that non-experts readily understand and are rooted in the true details of the prediction*" [17, p. 1493]. Two roles played by a feature correspond to normal and exceptional evidence. Normal evidence is the case of a feature that is relevant both in the training set classifications (high importance) and in the current classification (high effect). In contrast, exceptional evidence is the case of a feature that is not relevant in the training set classifications (low importance) but is relevant in the current classification (high effect). The evidence model is defined only for statistical classifiers, and it is based on the existence of a training phase for defining importance and a classification phase for defining effect. Therefore, an application of this model to other settings needs a new definition of these two notions.

In GEN-DS we reformulate these notions for relational knowledge, that is typical of several applicative domains. Our original contribution is to use the evidence for giving priority to a specific transaction. The *importance* reflects the past relevance of a transaction, while the *effect* evaluates the current relevance of a transaction. The combination of these two notions determines the narrative role of a transaction, i.e., the transaction *evidence*. The importance sets out a sort of "expectation" for a transaction in contrast to the effect, which, if it does not match the importance, results in a "surprise" that is worth mentioning. Thus, a transaction has the narrative role of exceptional evidence in the case of low importance and high effect. A key idea in GEN-DS is that normal evidence is not surprising and should not be mentioned, at least primarily, in the generated message. In contrast, the cases of exceptional evidence are surprising and should be mentioned prominently.

It is worth pointing out that the two most important elements in this specific context are money and time. Therefore, we formalize our intuitions that (a) the importance of customer-care service of a telecommunication company can be associated with the amount of money that the user usually spends for such service, and (b) that its effect can be associated with the amount of money that the user spent for the service in the last month [18]. Formally, a transaction is a money transfer operation between a customer and the company (i.e., an amount paid for a certain service). As a result, each transaction sequence represents the different amounts paid along a time period for a specific service (transaction type). We thus define the *importance of a transaction sequence* as the mean of the normalized values of the transactions in the past *K* months. In the following examples, we consider the previous six months (*K*=6). This value has been decided based on two considerations: a history going too deep in the past would generate messages that would be too verbose for the users and, moreover, from corpus analysis, it emerges that most of user requests do not concern very old transactions.

We define the *effect of a transaction sequence* as the normalized value of the transactions in the current month ($(K+1)^{th}$ month). Normalization is carried out by dividing the amount of the transactions by the maximum amount that the user has paid for that transaction. More formally, if S_i denotes the transactions sequence, M_j are the months and T_{ij} are the transactions regarding S_i occurred in month M_j, we can write Importance (Eq. 1) and Effect (Eq. 2) as:

$$Importance(S_i) = \frac{1}{K} \frac{\Sigma_{j=1,\dots,K} T_{ij}}{max_{j=1,\dots,K+1} T_{ij}} \quad (1)$$

$$Effect(S_i) = \frac{\Sigma_{s \in T_{iK+1}} s}{max_{j=1,\dots,K+1} T_{ij}} \quad (2)$$

These numeric real values need to be discretized to classify importance and effect as *high* or *low.* In accordance with the original model in [17], we determine the smallest subset *H* of transaction sequences such that the sum of their importance/effect values is at least a fraction *t* of the total importance/effect. When such a subset is not unique, we consider the union of all the smallest subsets. Note that the value of *t* is a tunable value that should be empirically validated on the specific domain. In the following, as in [17], we use *t*=75%. We now describe three examples of DC-knowledge to illustrate these notions.

Example DC-K-1. The first DC-knowledge in Table I has three transaction sequences: S_1, with an amount of 9.99 euros (M_1-M_7), S_2 with an amount of 2 euros (M_5-M_7, appearing twice in M_7), and S_3 with an amount of 1.59 euros (M_7). From this data, we calculate importance and effect for S_1, S_2 and S_3, and their narrative roles. The importance of S_1 is $Importance(S_1) = \frac{1}{6} \frac{9.99+9.99+9.99+9.99+9.99+9.99}{9.99} = 1$. The importance of S_2 is $Importance(S_2) = \frac{1}{6} \frac{2+2}{2} = 0.33$. The importance of S_3 is $Importance(S_3) = \frac{1}{6} \frac{0}{1.59} = 0$. So, the sum of the importance values is 1.33 and its 75% is 1. The smallest subset H_I such that the sum of the importance values is at least 1 is $H_I = \{S_1\}$, so S_1 has high importance, while S_2 and S_3 have low importance.

The effect of a transaction sequence is given by the values in the current month, so the effect of S_1 is $Effect(S_1) = \frac{9.99}{9.99} = 1$, the effect of S_2 is $Effect(S_2) = \frac{2+2}{2} = 2$, and the effect of S_3 is $Effect(S_3) = \frac{1.59}{1.59} = 1$. The sum of the effect values is 4 and its 75% is 3. The smallest subset H_E such that the sum of the effect is at least 3 is $H_E = \{S_1,S_2,S_3\}$, hence S_1, S_2 and S_3 all have high effect. As a result, combining the discrete values of importance and effect, S_1 is normal evidence, and S_2 and S_3 are both exceptional evidence.

Example DC-K-2. This example of DC-knowledge (the second example in Table I) has two transaction sequences: S_1, with an amount of 10 euros (M_1-M_7), and S_2 with an amount of 2 euros (M_6-M_7). Also from this data, we calculate importance and effect for S_1 and S_2. The importance of S_1 is $Importance(S_1) = \frac{1}{6} \frac{10+10+10+10+10+10}{10} = 1$. The importance of S_2 is $Importance(S_2) = \frac{1}{6} \frac{2}{2} = 0.17$. So, the sum of the importance values is 1.17 and its 75% is 0.88. The smallest subset H_I such that the sum of the importance values is at least 0.88 is $H_I = \{S_1\}$, so S_1 has high importance, while S_2 has low importance. The effect S_1 is $Effect(S_1) = \frac{10}{10} = 1$, and the effect of S_2 is $Effect(S_2) = \frac{2}{2} = 1$. The sum of the

effect values is 2 and its 75% is 1.5. The smallest subset H_E such that the sum of the effect is at least 1.5 is $H_E = \{S_1,S_2\}$, hence S_1 and S_2 have high effects. As a result, combining the discrete values of importance and effect, S_1 is normal evidence, and S_2 is exceptional evidence.

Example DC-K-3. This example of DC-knowledge (the third example in Table I) has three transaction sequences: S_1, with amounts of 13 euros (M_1-M_3) and 15 euros (M_4-M_7), S_2 with an amount of 0.9 euros (four times in M_7), and S_3 with an amount of 1.99 euros (in M_7). The importance of S_1 is $Importance(S_1) = \frac{1}{6}\frac{13+13+13+15+15+15}{15} = 0.94$, while $Importance(S_2) = Importance(S_3) = 0$. The sum of the importance values is 0.94 and its 75% is 0.71. The smallest subset H_I such that the sum of the importance values is at least 0.71 is $H_I = \{S_1\}$, so S_1 has high importance, while S_2 and S_3 have low importance.

The effect of S_1 and S_3 is 1, while the effect of S_2 is $Effect(S_2) = \frac{0,9+0.9+0.9+0.9}{0.9} = 4$. The sum of the effect values is 6 and its 75% is 4.5. The smallest subset H_E such that the sum of the effect is at least 4.5 can be $H_E = \{S_1,S_2\}$ or $H_E = \{S_2,S_3\}$. The subset H_E is the union of the two cases, i.e., $H_E = \{S_1,S_2, S_3\}$, hence S_1, S_2 and S_3 have high effect. Thus, S_1 is normal evidence, and S_2 and S_3 are exceptional evidence.

DC-K-1	M_1	M_2	M_3	M_4	M_5	M_6	M_7
S_1	9.99	9.99	9.99	9.99	9.99	9.99	9.99
S_2	0	0	0	0	2	2	2, 2
S_3	0	0	0	0	0	0	1.59

DC-K-2	M_1	M_2	M_3	M_4	M_5	M_6	M_7
S_1	10	10	10	10	10	10	10
S_2	0	0	0	0	0	2	2

DC-K-3	M_1	M_2	M_3	M_4	M_5	M_6	M_7
S_1	13	13	13	15	15	15	15
S_2	0	0	0	0	0	0	0.9,0.9, 0.9,0.9
S_3	0	0	0	0	0	0	1.99

Table I: DC-Knowledge for Examples DC-K-1, DC-K-2, and DC-K-3. Each row indicates the transactions of a specific category. We assume that all the transactions on the user's account are known.

In the next section, we describe how our reformulation of evidence can be used to guide the content selection process in GEN-DS.

C. *Designing and implementing GEN-DS*

GEN-DS follows the classical cascade architecture depicted in Fig. 2 [5]. GEN-DS is composed of three modules, which are *Natural Language Understanding* (NLU), *Natural Language Generation* (NLG), and *Dialogue Manager* (DM). The NLU module is devoted to the interpretation of the user's utterances. The NLG module is devoted to the final generation of the DS answer. The DM, that takes the input from the NLU module and produces the output for the NLG module, is devoted to managing all semantic and pragmatic elements that influence the future development of the dialogue. For instance, the DM can decide to answer with a question to a question. This classical architecture has a long history, but several advancements have recently been adopted. For instance, most modern DSs use NLU techniques based on machine learning to fill the important conceptual slots (e.g., *intents*

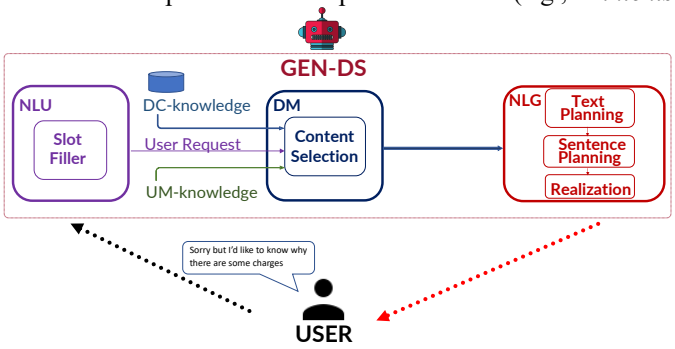


Fig. 2. The architecture of GEN-DS.

and *entities*, [11][19]) of the domain. Moreover, recent developments of neural NLG can be adopted also in some specific cases of generation in dialogues [20]. However, apart from systems devoted to *chit-chat*, also in modern task oriented DSs all the information must be coordinated by the DM to update the internal state of the DS and to produce the next dialogue act [5].

Many DSs assume that a relevant part of the necessary information is provided by the user's utterance, analyzed by the NLU module, and passed to the DM as *User Request* in Fig. 2 [5]. However, as outlined in Sections I and II.A, this assumption is only partially true in customer-care domain. Even a very advanced NLU module cannot make a detailed analysis in the case of a vague request as the one in Example 5 (Section II.A). Indeed, in this case very often commercial DSs apologize and ask users to repeat their request with more details [9]. Moreover, some user utterances are ungrammatical, as Example 3, and cannot be analyzed at all.

To provide better responses, in the case of vague or ungrammatical user requests, GEN-DS can resort to two other sources of information: the domain context knowledge (*DC-knowledge*) and the user model knowledge (*UM-knowledge*). In particular, the GEN-DS system depicted in Fig. 2 has been designed for overcoming the limitations of the *apologize-and-ask-to-repeat* strategy by using an NLG approach that exploits the DC-knowledge. In accordance with other systems developed for other domains [14], in GEN-DS the DC-knowledge plays a central role to produce the content of the answer: the basic idea is to produce responses that have exceptional evidence with respect to the specific DC-

knowledge. In this way we can anticipate the user intention by building interesting and concise answers to vague or ungrammatical requests. So, in GEN-DS, we specifically designed the NLU, the DM and the NLG modules for managing the explanation requests found in the corpus, but we believe that GEN-DS could be easily integrated in more general DSs designed for generic interactions. Note that the other source of information, that is the UM-knowledge, is not used for content selection but it plays a role in the realization sub-module deciding the linguistic details of the generated sentence. In the remaining part of this section, we describe the main features of the NLU, DM, and NLG modules of GEN-DS.

Natural Language Understanding module (NLU): it is based on regular expressions and is inspired by the NLU features of the COM-DS. In particular, the NLU distinguishes between the cases of generic or charge-related explanation requests. In some sense, the NLU must fill a single semantic slot related to the type of the request. For instance, in the case of ungrammatical utterances such as the one in Example 3, NLU returns a generic user request, while in the utterance from Example 2, the NLU returns a specific user request.

Dialogue Manager module (DM): it deals with the content selection task, that implements the notion of evidence formalized in the customer-care domain of a telecommunication company (see Section II.B), by providing the NLG module with all the transactions with their values of evidence. Moreover, in some specific cases, even the value of the user balance and the value of the total charges are provided to the NLG module. For instance, in the case of Example DC-K-1 in Table I, the content selection will pass the total amount of the charges, the transactions S_1, S_2, S_3 together with the information that S_2 and S_3 are both cases of exceptional evidence, and S_1 is normal evidence.

Natural Language Generation module (NLG): it includes, in turn, the sub-modules in charge of the three typical steps that characterize symbolic approaches to NLG, i.e., text planning, sentence planning and realization [21].

In general, text planning for NLG concerns both the selection of the salient information and its organization in a causal and temporal structure [21]. Indeed, since content selection is managed by the DM, the role of text planning in GEN-DS is to order the information provided the DM. We designed a very simple text planning schema to sort the content in a specific ordered list, that will be used in the realization for ordering the sentences: (1) information on user balance, (2) information on total charges, (3) the information on the transactions with exceptional evidence, and (4) the information on the transactions with normal evidence.

The sentence planner of GEN-DS is a rule-based module that defines the number and the types (e.g., passive, declarative) of the sentences in the final message, defines which sentences need to be merged for fluency, and defines which lexical elements to use for each sentence. GEN-DS uses a sentence planner previously adopted in several applicative projects of data-to-text generation [22]. The syntactic information on the sentences is encoded in a few predefined syntactic templates: Fig. 3 shows the syntactic template used to generate the first sentence in Example 8 (see below). The syntactic template is an unordered tree encoding notions from both constituency and dependency theories of syntax: it adopts both *phrases* from constituency theory (e.g., Noun Phrase, NP, Verbal Phrase, VP) and *relations* from dependency theory (such as subject and object, abbreviated to *subj* and *obj* in Fig. 3). Note that the trees do not fully specify the word inflection and the word order. The leaves in Fig. 3 (starting with #) indicate lexical items that will be specified in the realization by using the corresponding numeric values and the realizer domain dictionary.

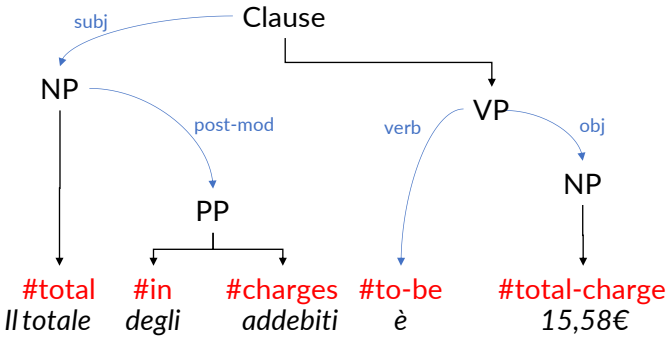


Fig. 3. A syntactic template for a declarative sentence. The leaves of the tree (in red, starting with #) contain lexical items that will be instantiated by the realizer.

The sentence planner decides which templates to use following two principles. The first principle regards visual readability: the sentences will be read in a textual chat; thus, shorter sentences are preferable. The second principle regards linguistic fluency: it prescribes to aggregate information on transactions which have the same value of evidence. For instance, in the case of Example DC-K-1 in Table I, S_1 will be communicated in one single sentence, and S_2 together with S_3 in another single sentence.

Finally, the realization process is implemented by using the SimpleNLG-IT library [23], which completes the syntactic templates with the necessary morpho-syntactic and orthographic information of the Italian language and the correct numeric values. SimpleNLG-IT is a rule-based realizer that formalizes the Italian grammar by producing morphologically correct word inflections and word orders. In the current implementation of GEN-DS, SimpleNLG-IT is the only module that uses information provided by the user model: for young users (less than twenty years), the pronoun *tu* is used (a colloquial second person pronoun), in contrast to the pronoun *lei* used for older people (a more formal second person pronoun).

Given Example DC-K-1 reported in Table I, the final output produced by SimpleNLG-IT is the one shown in Example 8.

(8) *Il totale degli addebiti è 15,58€. Recentemente hai pagato 4,00€ (2x€2,00) per l'Offerta Base Mobile e 1,59 € per l'Opzione ChiChiama e Richiama. Infine, come al solito, hai pagato il rinnovo dell'Offerta 20 GB Mobile (€9.99).*
(The total charge is €15.58. Recently, you paid €4.00 (2x€2.00) for the Basic Mobile Plan and €1.59 for the WhoCalled and CallMeBack Options. In addition, as usual, you paid for the renewal of the 20 GB Mobile Plan (€9.99)).

D. Building Experimental Scenarios

In this section and in the next, we present the first experimental human-based evaluation of GEN-DS.

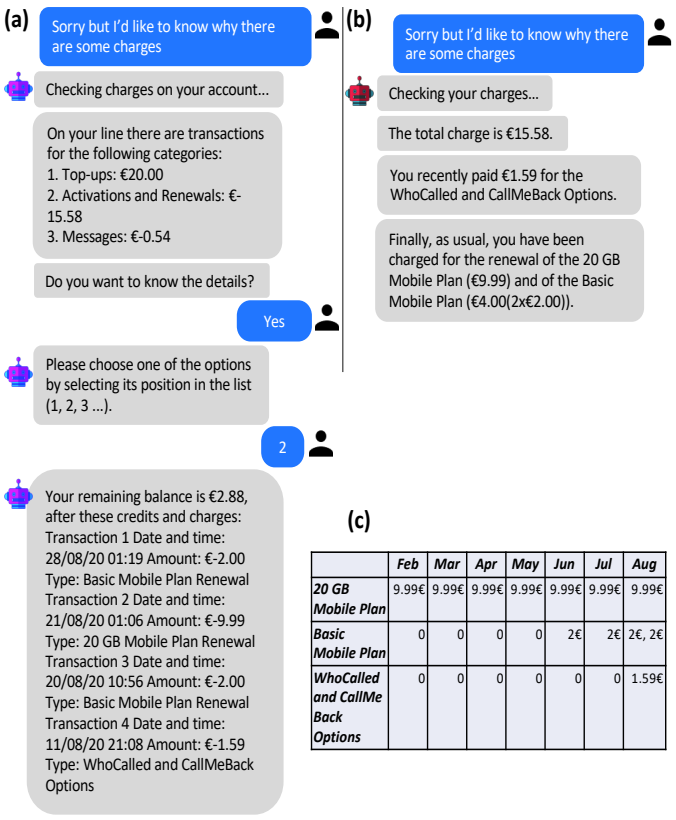
Our main goal was to design a realistic experimentation for GEN-DS. Different methodologies and frameworks have been proposed over the years to evaluate DSs [24][25][26]. However, we observed in our sample corpus that, among the possible situations where a customer may ask for assistance, just a small percentage (5%, see Section II.A) include explanation requests, which are the focus of our research. Therefore, it would not be trivial to collect enough samples in an interactive unconstrained experimentation conducted with live users. Thus, we decided to follow one of the experimental protocols defined in [27], where six prototypical dialogues and contexts, called *scenarios*, were generated off-line and evaluated by users along several properties. Notice also that the corpus analysis (Section II.A) showed that the whole range of possible customers' requests falls into three main categories. Using these three categories, we designed four distinct scenarios (reported in the supplementary material). In our study, a scenario is a prototypical situation consisting of a request for explanation, contained in the user utterance, and a specific DC-knowledge. It is worth pointing out that the four scenarios we devised ensure the coverage of all the categories extracted from the corpus and provide a good statistical power at the same time (see Section II.F). Moreover, the number of scenarios is comparable to the number of scenarios used in [27]. We built two scenarios (Scenario 1 and Scenario 3) using a linguistic input from Category 1 (the largest category), one scenario (Scenario 2) using a linguistic input from Category 2, and finally one scenario (Scenario 4) using a linguistic input from Category 3. For each scenario, we randomly extracted from the corpus one dialogue of the corresponding category. We then have used the user explanation request that opens the dialogue and the DC-knowledge of that dialogue to produce an answer with the GEN-DS system (see Fig. 4, with translated dialogues).

We recovered the DC-knowledge for the specific dialogue from the commercial system database: it consists of the user transactions of the last two months. As explained in Section II.B, we formalized the evidence-effect-importance over a period of seven months. We thus augmented the actual DC-knowledge obtained by the commercial system with ad-hoc realistic additional knowledge on the previous five months. With the aim to evaluate the utility of our formalization of DC-knowledge in realistic but similar situations, we designed Scenarios 1 and 3 keeping the same user's request though varying the transactions in the augmented knowledge.

The scenarios generated using COM-DS have an average of about 6 turns per dialogue (with an average length of 6 tokens in customer turns and 46 tokens in the agent turns). In contrast, the responses generated by GEN-DS, with 2 turns per dialogue (but an average length of 55 tokens in customer turns and 15 tokens in the agent turns) help reduce the number of turns overall, as the system provides the users with the required information right after the explanation request.

E. Participants and Experimental Procedure

To validate the experimental hypothesis that is that users prefer dialogue systems where the answers are generated (selected



	Feb	Mar	Apr	May	Jun	Jul	Aug
20 GB Mobile Plan	9.99€	9.99€	9.99€	9.99€	9.99€	9.99€	9.99€
Basic Mobile Plan	0	0	0	0	2€	2€	2€, 2€
WhoCalled and CallMe Back Options	0	0	0	0	0	0	1.59€

Fig. 4: Scenario 1 used in the experimentation translated in English. (a) shows the original dialogue selected from the corpus between a user and COM-DS, (b) shows the dialogue generated by GEN-DS, and (c) shows the common DC-knowledge. Due to space constraints, the original dialogue excerpt in Italian is provided in the Supplementary Material.

and/or ordered) based on the evidence of the transactions, we prepared an online questionnaire. In line with previous work on similar tasks (such as [11][20][24][27]), a pairwise comparison was carried out, in that users were asked to evaluate two different DSs: the original commercial system COM-DS, used to build the dialogue corpus (see Section II.A), and an implementation of GEN-DS (see Section II.C). We invited several colleagues, students, and acquaintances by email, asking for friendly participation without rewards. Around one hundred people have been invited and fifty-four users participated in our experiment. Thirty users (55.6%) were students, 23 users (42.6%) were employees, and one user (1.9%) was a teacher. Most of the users (29 users, 53.7%) were 18-30 years old, 11 users (20.4%) were 31-45 years old, 13 users (24.1%) were 46-60 years old and only one user (1.9%) was less than 18 years old. Finally, all the participants were Italian native speakers, and ten of them (18.5%) had no experience with DSs before this experimentation. Prior to participation we informed users that the survey concerned DSs, that no sensible data would be collected and that we ensured anonymity. As an introduction to the questionnaire, we informed users that they would be presented with different pairs of dialogues produced by different customer care DSs and that they would be asked to evaluate them over four different scenarios. In each pair of dialogues (Fig. 4), the systems were

simply tagged as System A and System B and arranged as Latin squares. We released the questionnaire as an online form, built with Google Form, composed of 43 questions, where 36 questions concerned dialogue scenarios, six questions concerned user profile information (age, educational qualification, occupation, technological skill, and their previous experience with chatbots), and one question was an open question for free comments.

We followed the experimental protocol defined by Demberg et al. ([27]): in the questionnaire, for each scenario, we presented both the original dialogue extracted from the corpus and a dialogue generated by GEN-DS. Both dialogues have the same user explanation request and the same DC-knowledge. For each dialogue the users were asked to rate four specific properties of the DSs, that are *usefulness, necessity, understandability, quickness*. Users were presented with a statement regarding such qualities and were asked to specify their agreement with a 7-point Likert scale where 1 corresponds to "I completely disagree" and 7 to "I completely agree" (the supplementary material contains the complete questionnaire). The statements are, respectively:

Usefulness: "*All the information provided by the system is USEFUL to respond to your request*" (Italian: "Le informazioni fornite dal sistema sono tutte UTILI per rispondere alla tua richiesta"). The rationale of this question is to check if all the information provided by the DS concerns the fulfilment of the explanation request, that is there is no useless information provided in relation to the explanation request. To answer this question, the user needs to consider the DS sentences in the dialogue together with the transaction table associated with the scenario, that is the specific DC-knowledge. In a sense, usefulness pertains to the notion of *precision* used in Information Extraction.

Necessity: *"All the information NECESSARY to answer your request has been presented by the system*". (Italian: "Tutte le informazioni NECESSARIE per rispondere alla tua richiesta sono state presentate dal sistema"). The rationale of this statement is to check whether the information contained in the specific scenario DC-knowledge provided by the DS concerns the fulfilment of the explanation request, that is whether there is no unnecessary information in relation to the explanation request. Also in this case the user needs to consider the DS sentences in the dialogue together with the specific scenario DC-knowledge. In a sense, necessity pertains to the notion of *recall* used in Information Extraction.

Understandability: "*The system provided the information in a way that it is easy to understand*". (Italian: "Il sistema ha fornito le informazioni in un modo facile da comprendere"). The rationale of this statement is to evaluate the comprehensibility of the language used by the DS in the conversation.

Quickness: *"The system was quick in allowing you to find the salient information*". (Italian: "Il sistema è stato rapido nel permetterti di trovare le informazioni salienti"). The rationale of this statement is to ask users to evaluate the "efficiency" of the text generated by the DSs concerning the requested explanation in quickly obtaining the desired information. Notice that this property does not concern computational performance (e.g., response time) of the system, but just "textual" features such as conciseness. We provide this question since this notion seems to be particularly important for the user satisfaction in DS interactions [27].

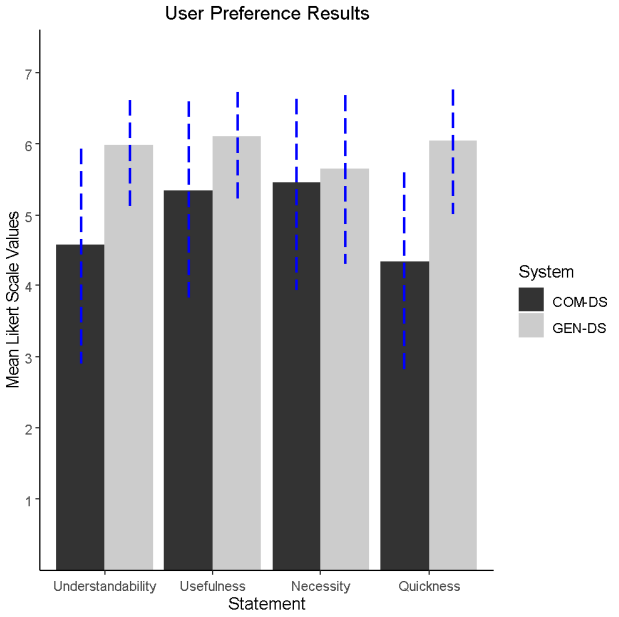


Fig. 5: Mean values for usefulness, necessity, understandability, and quickness for the two systems.

Satisfaction: "*Which system would you recommend to a friend?*" (Italian: "Quale dei due sistemi consiglieresti ad un amico?"). Following the evaluation schema proposed by Demberg et al. [27], to assess user satisfaction, in the questionnaire we include also a binary question asking whether the user prefers one system or the other.

F. Results

For each property presented above, we discuss the results (see Fig. 5–the supplementary material contains the answers to the questionnaire given by the 54 users).

Usefulness. This question assessed the user's confidence that all the information mentioned by the systems is relevant. GEN-DS (M=6.10, SD=0.95) had a higher mean compared to COM-DS (M=5.35, SD=1.58) and the difference was significant ($t(215)=6.16$, $p<0.001$) according to a two-tailed paired t-test.

Necessity. This question assessed the user's confidence that all the relevant information in the DC-knowledge has been mentioned by the system in the dialogue. The evaluation seems to show a slight preference for the GEN-DS system (M=5.65, SD=1.35) with respect to COM-DS (M=5.45, SD=1.51). However, this preference is not statistically significant ($t(215) = 1.52$, $p=0.07$).

Understandability. This question assessed the user's confidence that all the information mentioned by the systems is comprehensible. The mean of the GEN-DS system was rated significantly higher (M=5.98, SD=0.96) on this statement in comparison to COM-DS (M=4.58, SD=1.68, $t(215)=11.02$, $p<0.001$ according to a two-tailed paired t-test).

Quickness. This question assessed the user's confidence that the system presents information quickly. The mean of the GEN-DS system was rated significantly higher (M=6.04, SD=1.04) on this statement in comparison to COM-DS (M=4.35,

SD=1.56, t(215)=13.04, p<0.001 according to a two-tailed paired t-test).

Satisfaction. A significant preference for the GEN-DS system was observed. From a total of 216 choices in the experiment (54 participants x 4 dialogue pairs), GEN-DS was preferred 157 times (72.7%), whereas the dialogue from the corpus was preferred only 59 times (27.3%). This difference is significant according to a two-tailed binomial test (p<0.001). Thus, the null hypothesis that the corpus-based system is preferred at least as GEN-DS can be rejected with high confidence. We conducted a post-hoc power analysis to assess whether we had sufficient subjects, and we obtained a good power value (power=1) for usefulness, understandability, quickness, and satisfaction.

G. Subgroup analysis

As post-hoc analysis, to search for relations between the features characterizing the users and the scores that the users gave to the system properties in the questionnaire, we computed correlations between subgroups of users and how these users rated system properties. In Table II, we report the Pearson correlation coefficients for numerical features and the Spearman's rank correlation coefficients for categorical features. The only feature that showed significant correlation with users' scores is the age of the users. We report the correlation between the users' ages and the scores assigned by the users to the four system properties. It is interesting to notice that all the four properties show highly significant negative correlations with age when aggregating COM-DS and GEN-DS scores: in other words, this means that, as age grows, users tend to give lower scores regardless of the system. When the correlation is computed separately on COM-DS and GEN-DS, we have negative significant correlation values only for specific systems/properties. While the understandability of COM-DS decreases as age grows, this does not impact on GEN-DS. Indeed, we found that people in the range 18-30 gives 5.35/7 for understandability whilst people in the range 45-60 gives 3.98/7. Moreover, older users deem GEN-DS slower with respect to young users. Indeed, for the quickness property, users in the range 18-30 give 6.5/7 whilst those in the range 45-60 give 5.71/7. However, given the limited number of users in our experimentation, and the fact that more than half of the users are between 18 and 30 years old, these results should be replicated in a specifically designed experimentation.

III. Discussion

In this section we review the experiment results by considering the main research question underlying this paper, that is whether we can improve dialogue systems in the case of low-quality linguistic input by using content selection techniques that predict users' intentions.

As observed in our corpus, users often ask for explanations without providing enough linguistic information. Most commercial DSs, as the COM-DS considered in this study, cannot properly manage these dialogues. We built the GEN-DS system to properly address this issue. In particular, by using the formalization of evidence, GEN-DS is able to provide a tentative answer to unclear questions. The experiment described in Section II.E asks users to compare, along several properties, the responses provided by COM-DS in real conversations with the ones generated by GEN-DS based on the same DC-knowledge.

The results reported in Fig. 5 show that users deem GEN-DS superior to COM-DS with respect to the properties of usefulness, understandability, and quickness. The users report that GEN-DS presents the same DC-knowledge in a way that is more useful, understandable, and quick with respect to COM-DS. All these three properties are related to the way in which the relevant information is organized in the dialogue. The higher values reported for usefulness and quickness confirm that GEN-DS provides more relevant information and more concisely. As regards usefulness, such results can be attributed to the shorter dialogue length produced by the sentence planning module, which relies on two main principles (see Section II.C): one that promotes shorter sentences for the sake of readability, and one that aggregates the sentences based on the evidence model described in Section II.B, to improve linguistic fluency. The higher results reported for understandability seem to confirm that combining these principles does not come at the expense of linguistic clarity, on the contrary, it enhances it. In contrast, GEN-DS and COM-DS are not deemed statistically different with respect to necessity, that is the property of a dialogue to present all the necessary contextual information. This was an expected result as COM-DS offers a detailed account of all the transactions of the last two months, and the information it provides is a superset of the information provided by GEN-DS. The overall preference of the users toward GEN-DS is confirmed by the satisfaction question, where we asked them to explicitly compare COM-DS and GEN-DS. In short, based on these findings, we can conclude that *we can improve DSs in the case of low-quality linguistic input by predicting the users' intentions.*

IV. Conclusions

In this paper we proved that the quality of a DS in the domain of customer care can be improved by using the notion of evidence. Based on a preliminary corpus analysis, we observed that several explanation requests, that are often used by users to start their conversations with DSs, are vague or ungrammatical or, more in general, hardly understandable. In such cases the dialogue manager does not have sufficient linguistic information to produce a meaningful answer. Most commercial DSs deal with this situation with a simple apologize-and-ask-to-repeat strategy. However, a possible handling strategy may

Property	**Understandability**		**Usefulness**		**Necessity**		**Quickness**	
System	COM-DS	GEN-DS	COM-DS	GEN-DS	COM-DS	GEN-DS	COM-DS	GEN-DS
Age	-.441**		-.305**		-.436**		-.371**	
	-.406**	-.193	-.295*	-.224	-.327*	-.344*	-.228	-.426**

Table II. Correlation between system properties and user age. * indicates a significant correlation at the 0.05 level (2-tailed), ** indicates a significant correlation at the 0.01 level (2-tailed).

consist in using the extra-linguistic knowledge related to the dialogue, such as the UM-knowledge (about the user) and the DC-knowledge (about the domain). In the specific context of customer care, the DC-knowledge consists of the commercial transactions between the user and the company. For many companies this kind of information is encoded in relational structures.

In this paper, we provided a formal definition of *evidence* for relational data that can be used to select content from the DC-knowledge, and we implemented this notion on a data-to-text generation system that we called GEN-DS.

The experimentation in Section II.E compared real dialogues taken from a corpus of human/COM-DS conversations for the customer care service of a telecommunication company with synthetic dialogues generated by the GEN-DS system. The questionnaire results showed preferences for GEN-DS dialogues by measuring different properties: usefulness, necessity, understandability, quickness, general satisfaction. Thus, we showed that this formalization can improve the quality of the dialogues in the customer-care domain.

To test GEN-DS in the specific case of a conversation that starts with hardly understandable user sentences, we designed the experimentation as a simulated dialogue rather than a real one. We are aware of the limit of this kind of experiment, but the specificity of our research goal does not allow to design a natural interaction with users to judge the contribution of the notion of evidence for content selection.

We believe that the results of this study can be easily extended to other application domains. Indeed, the core idea of our research is the formalization of the notion of evidence for relational knowledge and its application to NLG in the customer care domain. This notion was originally defined by Biran and McKewon in the field of machine learning and, inspired by their formalization of evidence in terms of importance and effect, we proposed in this work (1) to encode the *past* behavior of the system into *importance*, (2) to encode the *recent* behavior of the system into *effect*. So, to apply our approach of generation based on evidence in DC-knowledge, one needs to reformulate the notions of relevance, importance and effect for the specific task related to the DS.

Finally, a central question that arises from our research concerns the possibility of mixing together the DC-knowledge, the UM-knowledge, and the user requests. As a future work, it could be interesting to investigate how the notion of evidence can be used also in case of *understandable* linguistic input and how the user model can contribute to this process.

REFERENCES

[1] J. Grudin, R. Jacques, R. "Chatbots, humbots, and the quest for artificial general intelligence" in *Proc. of the Conference on Human Factors in Computing Systems*, pp. 1-11, ACM, 2019.

[2] M. Chung, E. Ko, H. Joung. S. J. Kim. "Chatbot e-service and customer satisfaction regarding luxury brands". *Journal of Business Research*, Vol. 117, pp. 587-595, 2018.

[3] E. Kimani, K. Rowan, D. McDuff, M. Czerwinski, G. "A Conversational Agent in Support of Productivity and Wellbeing at Work" in *Proc. of ACII,* pp. 1-7, 2019.

[4] A. Rapp, L. Curti, A. Boldi, "The human side of human-chatbot interaction: A systematic literature review of ten years of research on text-based chatbots". *Int. Journal of Human-Computer Studies*, Vol. 151, July 2021.

[5] M. McTear, Z. Callejas, and D. Griol. *The Conversational Interface: Talking to Smart Devices*. Springer, 1st edition, 2016.

[6] M. Jain, P. Kumar, R. Kota, S.N. Patel, "Evaluating and informing the design of chatbots" in *Proc. of Designing Interactive Systems Conference.* ACM, pp. 895–906, 2018.

[7] A. Følstad, M. Skjuve, "Chatbots for customer service: user experience and motivation", in *Proc of CUI*, ACM, pp. 1–9, 2019.

[8] A. Følstad, M. Skjuve, P.B. Brandtzaeg, "Different Chatbots for Different Purposes: Towards a Typology of Chatbots to Understand Interaction Design" in *Proc. of INSCI*, pp. 145–156, 2018.

[9] M. Aliannejadi, H. Zamani, F. Crestani, and W. B. Croft. "Asking clarifying questions in open-domain information-seeking conversations" in *Proc. of ACM SIGIR*, pp. 475—484, 2019.

[10] K. Fang, Q. Zhang, Z. Zhuang and Z. Zhang. "Making, Recommendations Better: The Role of User Online Purchase Intention Identification" in *Proc. of ICSN*, 2016.

[11] X. Li, Y.-N. Chen, J. Gao and A. Celikyilmaz. "End-to-end task-completion neural dialogue systems" in *Proc. of IJCNLP*, 2017.

[12] M. Di Lascio, M. Sanguinetti, L. Anselma, D. Mana, A. Mazzei, V. Patti, and R. Simeoni, "Natural language generation in dialogue systems for customer care" in *Proc. of CLiC-it,* pp. 1–6, CEUR, 2020.

[13] M. Sanguinetti, A. Mazzei, V. Patti, M. Scalerandi, D. Mana, and R. Simeoni, "Annotating errors and emotions in human-chatbot interactions in Italian" in *Proc. of LAW*, pp. 1–12, ACL, 2020.

[14] G. Stoilos, S. Wartak, D. Juric, J. Moore, and M. Khodadadi. "An Ontology-Based Interactive System for Understanding User Queries" in *Proc. of ESWC, 2019.*

[15] E. Reiter, "Natural language generation challenges for explainable AI" in *Proc. of NL4XAI*, pp. 3–7, ACL, 2019.

[16] R. Puduppully and M. Lapata. "Data-to-text Generation with Macro Planning". *TACL*, Vol. 9, pp. 510–527, ACL, 2021.

[17] O. Biran and K. McKeown, "Human-centric justification of machine learning predictions" in *Proc of.* IJCAI, pp. 1461–1467, 2017.

[18] L. Anselma, M. Di Lascio, D. Mana, A. Mazzei, and M. Sanguinetti, "Content selection for explanation requests in customer-care domain' in *Proc. of INLT4XAI*, pp. 5–10, ACL, 2020.

[19] A. M. Preininger, B. South, J. Heiland, A. Buchold, M. Baca, S. Wang, R. Nipper, N. Kutub, B. Bohanan, and G. Purcell Jackson. "Artificial intelligence-based conversational agent to support medication prescribing". *JAMIA* open Vol. 3, No. 2, pp. 225-232, 2020.

[20] T. Zhao, A. Lu, K. Lee and M. Eskenazi. "Generative Encoder-Decoder Models for Task-Oriented Spoken Dialog Systems with Chatting Capability" in *Proc. of SIGDIAL*, pp. 27-36, ACL, 2017.

[21] A. Gatt, E. Krahmer, "Survey of the state of the art in natural language generation: core tasks, applications and evaluation*". J. Artif. Int. Res.* 61, 1, pp. 65–170, 2018.

[22] L. Anselma and A. Mazzei, "Building a Persuasive Virtual Dietitian*". Informatics*. Vol. 7. No. 3. MDPI, 2020.

[23] A. Mazzei, C. Battaglino, and C. Bosco. "SimpleNLG-IT: adapting SimpleNLG to Italian" in *Proc. of INLG*, pp. 184–192, ACL, 2016

[24] M. Danieli and E. Gerbino. "Metrics for evaluating dialogue strategies in a spoken language system" in *Proc. of AAAI symposium on Empirical Methods in Discourse Interpretation and Generation.* Vol. 16, 1995.

[25] M. Walker, C. Kamm and D. Litman. "Towards developing general models of usability with PARADISE". *Natural Language Engineering* Vol. 6, No 3-4, pp. 363-377, 2000.

[26] J. Deriu, A. Rodrigo, A. Otegi, G. Echegoyen, S. Rosset, E. Agirre, M. Cieliebak. "Survey on evaluation methods for dialogue systems". Art. Int. Rev., Vol. 54, pp. 755-810, Springer, 2021.

[27] V. Demberg, A. Winterboer, and J. D. Moore, "A strategy for information presentation in spoken dialog systems*". Computational Linguistics*, Vol. 37, No 3, pp. 489–539, ACL, 2011.