Natural Language Generation in Dialogue Systems for Customer Care

(Article begins on next page)

06 October 2024

Natural Language Generation in Dialogue Systems for Customer Care

Mirko Di Lascio$^\heartsuit$, Manuela Sanguinetti$^{\heartsuit\diamond}$, Luca Anselma$^\heartsuit$, Dario Mana$^\clubsuit$,
Alessandro Mazzei$^\heartsuit$, Viviana Patti$^\heartsuit$, Rossana Simeoni$^\clubsuit$

$^\heartsuit$Dipartimento di Informatica, Università degli Studi di Torino, Italy
$^\diamond$Dipartimento di Matematica e Informatica, Università degli Studi di Cagliari, Italy
$^\clubsuit$TIM, Torino, Italy

$^\heartsuit${first.last}@unito.it, $^\diamond${first.last}@unica.it, $^\clubsuit${first.last}@telecomitalia.it

Abstract

English. In this paper we discuss the role of natural language generation (NLG) in modern dialogue systems (DSs). In particular, we will study the role that a linguistically sound NLG architecture can have in a DS. Using real examples from a new corpus of dialogue in customer-care domain, we will study how the non-linguistic contextual data can be exploited by using NLG.

1 Introduction

In this paper we present the first results of an ongoing project on the design of a dialogue system for customer care in the telco field. In most of the dialogue systems (DSs), the generation side of the communication is quite limited to the use of *templates* (Van Deemter et al., 2005). Templates are pre-compiled sentences with empty *slots* that can be filled with appropriate *fillers*. Most of commercial DSs, following the classical cascade architecture $NLUnderstanding \leftrightarrow DialogueManager \leftrightarrow NLGeneration$ (McTear et al., 2016), use machine learning-based Natural Language Understanding (NLU) techniques to identify important concepts (e.g., *intent* and *entities* in (Google, 2020)) that will be used by the dialogue manager (i) to update the state of the system and (ii) to produce the next dialogue act (Bobrow et al., 1977; Traum and Larsson, 2003), possibly filling the slots in the generation templates.

This classical, and quite common, information flow/architecture for dialogue processing has, as a working hypothesis, the assumption that most of *necessary* information is provided by the user's utterance: we call this information *linguistic channel* (L-channel). However, especially in the customer-care domain, this assumption is only partially true. For instance, in the sentence "*Scusami ma vorrei sapere come mai mi vengono fatti certi addebiti?*" ("Excuse me, I'd like to know why I'm charged certain fees?"), even a very advanced NLU module can produce only a vague information about the user's request to the DialogueManager. Indeed, in order to provide good enough responses, the DialogueManager resorts to other two sources of information: the *domain context channel* (DC-channel) and the *user model channel* (UM-channel). The DC-channel is fundamental to produce the *content* of the answer, while the UM-channel is necessary to give also the correct *form*.

It is worth noting that both channels, that are often neglected in the design of commercial DSs for customer-care domain, have central roles in the design of (linguistically sound) natural language generation (NLG) systems (Reiter and Dale, 2000). In particular, considering the standard architecture for data-to-text NLG systems (Reiter, 2007; Gatt and Krahmer, 2018), the analysis of the DC-channel exactly corresponds to the *content selection* task and the UM-channel influences both the sentence planning and sentence realization phases. In other words, the central claims of this paper are that in commercial DSs for customer care: (1) *L-channel is often not informative enough and one needs to use the DC-channel and the UM-channel for producing a sufficiently good answer*, (2) *DC-channel and UM-channel can be exploited by using standard symbolic[1] NLG techniques and methods*. The remainder of the paper supports both of these claims while presenting our ongoing project on the development of a rule-

---

[1]The well-known problem of *hallucinations* in neural networks deters their use in real-world NLG (Rohrbach et al., 2018).

based NLG prototype to be used in a customer care domain. Section 2 presents the corpus developed in the first stage of this project, consisting of real dialogues containing explanation requests in telco customer-care domain. Section 3 presents an NLG architecture for managing the L-DC-UM channels that can be adopted in a DS for customer care. Finally, Section 4 concludes the paper with few remarks on the current state of the project and on future work.

2 A Dialogue Corpus for Customer-care Domain

This study builds upon the analysis of a corpus of dialogues between customers and a DS for customer service developed by an Italian telecommunications company. The dialogues, which take place by means of a textual chat, mainly deal with requests for commercial assistance, both on landline and mobile phones. For the purpose of this study, the corpus was extracted by selecting, from a sample of dialogues held over 24 hours, a reduced subset that included requests for explanations from customers. The selection criteria were conceived so as to include all the dialogues where at least one message from the user contained a clearly stated request for explanation. The kind of requests identified in this collection basically reflects the problems typically encountered with a telecom service provider, such as undue or unfamiliar charges in the bill or in the phone credit (about 52% of the overall number of requests in this dataset).

The resulting corpus consists of 142 dialogues, with an average of 11 turns per dialogue, and an average length of 9 tokens in customer messages and 38 tokens in the bot messages. Such difference in the message length is due to the way the assistant's responses are currently structured, in that they usually include detailed information on invoice items or options available, while, on the other hand, customer's messages are most often quite concise. Also, the relatively high number of turns per dialogue might be explained with the high occurrence in the corpus of repeated or rephrased messages, both by the chatbot and by the customer, due to recurring misunderstandings on both sides.

As a matter of fact, the presence of such phenomena in the corpus, along with the overall goals set forth for the development of the NLG module

(a)

🤖 **Vuoi vedere il dettaglio della fattura?**
(Do you want to see the invoice details?)

👤 **La fattura di gennaio 2020 non è arrivata**
(I haven't received the invoice of January 2020 yet)
Ignoring question/feedback - Neutral

🤖 **Ecco il dettaglio delle voci di spesa presenti nella fattura `InvoiceNumber` del mese di gennaio per la linea `PhoneNumber:InvoiceDetails`. Vuoi avere il dettaglio di una specifica voce di spesa presente in fattura?**
(Here is the detail of the items on the invoice `InvoiceNumber` of January for the `PhoneNumber:InvoiceDetails`. Do you want the detail of a specific item in the invoice?)
Straight wrong response

👤 **Non mi interessa questa fattura, mi serve gennaio 2020**
(I don't need this invoice, I need the one of January 2020)
Repetition - Somewhat frustrated

🤖 **Vuoi cambiare argomento?**
(Do you want to change topic?)
Topic change

👤 **Ciao**
(Whatever)
Non-cooperativity - Somewhat frustrated

(b)
🤖 **Scusami, non ho capito bene la tua richiesta... Puoi ripeterla in modo più semplice?**
(I'm sorry, I didn't get your question... Could you rephrase it in a simpler way?)

👤 **No non hai capito niente**
(No you got it all wrong)
Non cooperativity - Somewhat angry

🤖 **Mi spiace non aver compreso correttamente. Ho la possibilità di farti chattare con un operatore. umano più esperto. Vuoi?**
(I'm sorry I haven't understood that correctly. I can put you in contact with a human representative. Is this what you want?)
Empathy

Figure 1: Excerpts from the annotated dataset. Annotation of errors is highlighted in red, that of customer's emotions in blue and the agent's empathy in orange.

in this project, led us to the design of an annotation process that involved different dimensions, such as errors in conversation and emotions. By er-

ror, in this context, we mean any event that might have a negative impact on the flow of the interaction, and more generally on its quality, potentially resulting in breakdowns (i.e. whenever one party leaves the conversation without completing the given task (Martinovsky and Traum, 2003)). The error tagset used in this corpus is partially inspired by three of the popular Gricean maxims, i.e. those of *Quantity*, *Relation* and *Manner* (Grice, 1989) (each one including further sub-types, not described here), and it has been conceived so as to include error categories that may apply to both conversation parties. The second dimension, instead, is meant to include, on the one hand, customers' emotions (as perceived in their messages), and, on the other hand, the chatbot's empathic responses (if any). In particular, as regards customers' emotions, besides two generic labels for neutral and positive messages, we mostly focused on negative emotions, especially anger and frustration, also introducing for these ones two finer-grained labels that define their lower or higher intensity. While a full description of the annotation scheme is beyond the scope of this paper, Figure 1 shows two brief examples of how we applied this scheme to the sample dataset[2]. An overview of the scheme with a discussion on the main findings and annotation issues can be found in Sanguinetti et al. (2020).

Due to privacy concerns and the related anonymization issues that may arise (as further discussed in Section 4), the corpus cannot yet be publicly released. However, in an attempt to provide a qualitative analysis of the annotated data, we collected some basic statistics on the distribution of errors and emotions labeled in this sample set. Overall, we report an amount of 326 errors (about 21% of the total number of turns) from both parties; among them, the error class that includes violations of the maxim of Relevance is by far the most frequent one (65% of the errors). Such violations may take different forms, also depending on whether they come from the customer or the chatbot. As regards the customer, errors of such kind typically take place when the user does not take into account the previous message from the chatbot, thus providing irrelevant responses that do not allow to move forward with the conversation and make any progress; these cases cover

approximately 21% of customers' errors. On the chatbot side, the most frequent error type is represented by those cases in which the agent misinterprets a previous customer's message and proposes to move on to another topic rather than providing a proper response (30% of cases). As for the second annotation dimension, i.e. the one regarding customers' emotions, most of the messages have a neutral tone (about 86% of user turns), but, among non-neutral messages, the two main negative emotions defined in this scheme, namely anger and frustration, are the ones most frequently encountered in user messages (both with a frequency of 41%), while the cases of messages with a positive emotion constitute less than 1%, and usually translate into some form of gratitude, appreciation, or simple politeness.

All these dimensions are functional to a further development of the NLG module, in that they provide, through different perspectives, useful signals of how, and at which point in the conversation, the template response currently used by the chatbot might be improved using the NLG module. Broadly speaking, framing the error taxonomy within the Grice's cooperative principle provides a useful support for the generation module to understand, in case an error is reported, how to structure the chatbot response so as to improve the interaction quality in terms of informativeness and relevance (as also discussed in Section 3).

3 Balancing information sources in NLG for DS

In this Section, we illustrate a DS architecture that explicitly accounts for the L-DC-UM information channels. In particular, we point out that DC and UM channels can be managed by using standard NLG methods.

A commonly adopted architecture for NLG in data-to-text systems is a pipeline composed of four modules: data analyzer, text planner, sentence planner and surface realizer (Reiter, 2007; Pauws et al., 2019). Each module tackles a specific issue: (1) the data analyzer determines what can be said, i.e. a domain-specific analysis of input data; (2) the text planner determines what to say, i.e. which information will be communicated; (3) the sentence planner determines how to communicate, with particular attention to the design of the features related to the given content and language (e.g. lexical choices, verb tense, etc.); (4)

---

[2]For further details on the scheme and the definition of all tags, the annotation guidelines are available in this document: `https://cutt.ly/cdMcnyM`
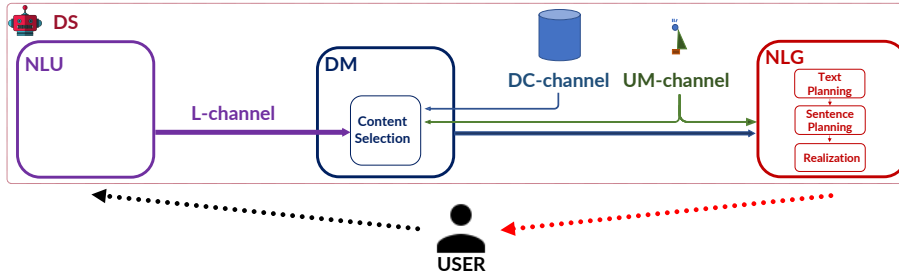
Figure 2: A dialogue system architecture accounting for L-DC-UM channels.

the surface realizer produces the sentences by using the results of the previous modules and considering language-specific constraints as well. Note that by definition NLG does not account for linguistic input (that is, L-channel), all the modules account for the context of the communication. In other words, data analysis and text planning explicitly process the information about the input data (the DC-channel), and text planning and sentence planning process the information about the audience (the UM-channel). Moreover, by using the nomenclature defined in (Reiter and Dale, 2000), the specific task of *content selection* decides *what to say*, that is the atomic nucleus of information that will be communicated.

In our project, we adopt a complete NLG architecture in the design of the DS (Figure 2). In Figure 2, we show the contributions of the L-DC-UM channels in the interaction flow. It is worth noting that we assigned the *content selection* task to the DM module rather than to the text planning of the NLG module. Indeed, the content selection task is crucially the point where all the three information channels need to be merged in order to decide the content of the DS answer to the user question.

In order to understand the contribution of the three information channels to the final message construction, we describe below the main steps of the module design using the following customer's message, retrieved from the corpus, as an example:

Scusami ma vorrei sapere come mai mi vengono fatti alcuni addebiti?. ("Excuse me, I'd like to know why I'm charged certain fees?")

Here, the customer requests for an explanation about some (unspecified) charges on her/his bill, making the whole message not informative enough. In this case, the DS can deduce from the L-channel only a generic request of information on transactions. However, using the architecture

shown in Figure 2, a more informative answer can be produced considering the UM-channel and the DC-channel.

As a working hypothesis, we assume that the user model consists uniquely in the age of the user. By assuming that the user is 18 years old, we can say that the DS should use an informal register, i.e. the Italian second person singular (*tu*) rather than the more formal third person singular (*lei*). It is worth noting that the current accounting of the user model is too simple and there is room for improvement both in the formalization of the model, and in the effect of the user model on the generated text. Taking into account the classification of the user model acquisition given by (Reiter et al., 2003), it is interesting to note that the dialogic nature of the system allow for the possibility to explicitly ask users about their knowledge and preferences on the specific domain.

Moreover, we assume that the DC-channel consists of all the transactions of the last 7 months, for example: T1, with an amount of 9.99€ (M1-M7); T2 with an amount of 2€ (M5-M7, appearing twice in M7); and T3 with an amount of 1.59€ (M7) (see Table 1).

	M1	M2	M3	M4	M5	M6	M7
T1	9.99	9.99	9.99	9.99	9.99	9.99	9.99
T2	0	0	0	0	2	2	2, 2
T3	0	0	0	0	0	0	1.59

Table 1: A possible transactions history.

Looking at the data in Table 1, different forms of automatic reasoning could be applied in order to evaluate the relevance of each singular transaction of the user. At this stage of the project, we aim to adapt the theory of importance-effect from (Biran and McKeown, 2017) to our specific domain, where the relevant information is in the form of relational database entries. The idea is to

consider the time evolution of a specific transaction category, giving more emphasis to information contents that can be classified as *exceptional evidences*. Informally, we can say that the transactions T2 and T3 have a more irregular evolution in time with respect to T1, therefore they should be mentioned with more emphasis in the final message.

The current implementation of the DS is based on a trivial NLU (regular-expressions), a symbolic sentence planner and realizer (for Italian) (Anselma and Mazzei, 2018; Mazzei et al., 2016). By considering all the three L-UM-DC channels, the answer generated by the DS is:

Il totale degli addebiti è €15, 58. Hai pagato €4, 00 (2×€2, 00) per l'Offerta Base Mobile e €1, 59 per l'Opzione ChiChiama e RiChiama. Infine, hai pagato il rinnovo dell'offerta 20 GB mobile. ("The total charge is €15.58. You have been charged €4.00(2×€2.00) for the Mobile Base Offer and €1.59 for the Who'sCalling and CallNow options. Finally, you have been charged for the renewal of the 20 GB mobile offer.")

4 Conclusion and Future Work

In this paper we have discussed the main features of the design of a DS system for telco customer care. In particular, we outlined the peculiarities of this domain, describing the construction of a specifically-designed dialogue corpus and discussing a possible integration of standard DS and NLG architectures in order to manage these peculiarities. This is an ongoing project and we are considering various enhancements: (1) we will integrate emoji prediction capabilities into the proposed architecture in order to allow the DS to automatically attach an appropriate emoji at the end of the generated response, relying on previous work for Italian (Ronzano et al., 2018); we would also take into account the current user emotions, while generating an appropriate emoji – it may be the case that an emoji that is adequate when the conversation is characterized by a neutral tone, suddenly becomes inappropriate if the user is frustrated or angry (Pamungkas, 2019; Cercas Curry and Rieser, 2019); (2) we would like to enhance the system so as to adapt the generated responses to other aspects of the users, such as their mental models, levels of domain expertise, and personality traits; (3) we want to evaluate the DS following the user-based comparative schema adopted in

(Demberg et al., 2011).

Finally, we add some closing remarks on the corpus availability and its anonymization. The publication of a dataset of conversations between customers and a company virtual assistant is a great opportunity for the company and for its surrounding communities of academics, designers, and developers. However, it entails a number of obstacles to overcome. Rules and laws by regulating bodies must be strictly followed – see, for example, the GDPR regulation[3]. This means, first of all, including within the to-be-published dataset only those conversations made by customers who have given their consent to this type of treatment of their data. Moreover, it is mandatory to obscure both personal and sensitive customer data. Such obfuscation activities are particularly difficult in the world of chatbots, where customers are free to input unrestricted text in the conversations. Regular expressions can be used in order to recognize the pieces of data to be obscured, such as email addresses, telephone numbers, social security numbers, bank account identifiers, dates of birth, etc. More sophisticated techniques needed be adopted to identify and obscure, within the text entered by customers, names, surnames, home and work addresses. Even more complex and open is the problem of anonymizing sensitive customer data. For example, consider the case of a disabled customer who reveals his/her sanitary condition to the virtual assistant, in order to obtain a legitimate better treatment from the company: the text revealing the health condition of the customer must be obscured. Other relevant sensitive data include racial or ethnic origins, religious or philosophical beliefs, political opinions, etc. Some of these techniques, used for identifying certain types of data to be obscured, have a certain degree of precision that may even be far, given the current state of the art, from what a trained human analyst could do. Therefore, it is also necessary to consider the need for the dataset being published to be reviewed and edited by specialized personnel before the actual publication. With this in mind, the techniques of data recognition mentioned above - regular expressions, Named Entity Recognition, etc. - could also be exploited to develop tools that can speed up the task of completing and verifying the accurate anonymization of the dataset.

---

[3] `https://eur-lex.europa.eu/eli/reg/2016/679/oj`

References

Luca Anselma and Alessandro Mazzei. 2018. Designing and testing the messages produced by a virtual dietitian. In *Proceedings of the 11th International Conference on Natural Language Generation, Tilburg University, The Netherlands, November 5-8, 2018*, pages 244–253.

Or Biran and Kathleen McKeown. 2017. Human-centric justification of machine learning predictions. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, pages 1461–1467.

Daniel G. Bobrow, Ronald M. Kaplan, Martin Kay, Donald A. Norman, Henry Thompson, and Terry Winograd. 1977. Gus, a frame-driven dialog system. *Artif. Intell.*, 8(2):155–173, April.

Amanda Cercas Curry and Verena Rieser. 2019. A crowd-based evaluation of abuse response strategies in conversational agents. In *Proceedings of the 20th Annual SIGdial Meeting on Discourse and Dialogue*, pages 361–366, Stockholm, Sweden, September. Association for Computational Linguistics.

Vera Demberg, Andi Winterboer, and Johanna D. Moore. 2011. A strategy for information presentation in spoken dialog systems. *Computational Linguistics*, 37(3):489–539.

Albert Gatt and Emiel Krahmer. 2018. Survey of the state of the art in natural language generation: Core tasks, applications and evaluation. *J. Artif. Intell. Res.*, 61:65–170.

Google. 2020. Dialogflow documentation. https://dialogflow.com. Online; accessed 2020-08-10 11:24:07 +0200.

Paul Grice. 1989. *Studies in the Way of Words*. Harvard University Press, Cambridge, Massachussets.

Bilyana Martinovsky and David Traum. 2003. The error is the clue: Breakdown in human-machine interaction. In *In Proceedings of the ISCA Workshop on Error Handling in Dialogue Systems*.

Alessandro Mazzei, Cristina Battaglino, and Cristina Bosco. 2016. SimpleNLG-IT: adapting SimpleNLG to Italian. In *Proceedings of the 9th International Natural Language Generation conference*, pages 184–192, Edinburgh, UK, September 5-8. Association for Computational Linguistics.

Michael McTear, Zoraida Callejas, and David Griol. 2016. *The Conversational Interface: Talking to Smart Devices*. Springer Publishing Company, Incorporated, 1st edition.

Endang Wahyu Pamungkas. 2019. Emotionally-aware chatbots: A survey. *CoRR*, abs/1906.09774.

Steffen Pauws, Albert Gatt, Emiel Krahmer, and Ehud Reiter. 2019. Making effective use of healthcare data using data-to-text technology. In *Data Science for Healthcare*, pages 119–145. Springer.

Ehud Reiter and Robert Dale. 2000. *Building Natural Language Generation Systems*. Cambridge University Press, New York, NY, USA.

Ehud Reiter, Somayajulu Sripada, and Sandra Williams. 2003. Acquiring and using limited user models in NLG. In *Proceedings of the 9th European Workshop on Natural Language Generation (ENLG-2003) at EACL 2003*.

Ehud Reiter. 2007. An architecture for data-to-text systems. In *Proc. of the 11th European Workshop on Natural Language Generation*, ENLG '07, pages 97–104, Stroudsburg, PA, USA. Association for Computational Linguistics.

Anna Rohrbach, Lisa Anne Hendricks, Kaylee Burns, Trevor Darrell, and Kate Saenko. 2018. Object hallucination in image captioning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4035–4045, Brussels, Belgium, Nov. Association for Computational Linguistics.

Francesco Ronzano, Francesco Barbieri, Endang Wahyu Pamungkas, Viviana Patti, and Francesca Chiusaroli. 2018. Overview of the EVALITA 2018 Italian Emoji Prediction (ITAMoji) Task. In *Proceedings of the Sixth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2018)*, volume 2263 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Manuela Sanguinetti, Alessandro Mazzei, Viviana Patti, Marco Scalerandi, Dario Mana, and Rossana Simeoni. 2020. Annotating Errors and Emotions in Human-Chatbot Interactions in Italian. In *Proceedings of the 14th Linguistic Annotation Workshop (LAW@COLING 2020)*. Association for Computational Linguistics.

David Traum and Staffan Larsson. 2003. The Information State Approach to Dialogue Management. In *Current and New Directions in Discourse and Dialogue*, pages 325–353. Springer.

Kees Van Deemter, Emiel Krahmer, and Mariët Theune. 2005. Real versus template-based natural language generation: A false opposition? *Comput. Linguist.*, 31(1):15–24, March.