Socially-acceptable Extended Reality Models and Systems

D5.2 SERMAS Dialogue Management

Date 31/03/2023

Funded by the European Union

DELIVERABLE INFORMATION	
Deliverable leader	TUDa
Document type	OTHER
Document code	D5.2
Document name	Dialogue Management
Work Package / Task	WP2 / T5.2 Dialogue Management
Delivery Date (DoA)	30/03/2024
Actual Delivery Date	
Reviewers	SUPSI

DELIVERABLE HISTORY			
Date	**Version**	**Author**	**Summary of main changes**
31.08.2023	0.1	T. Tran (TUDa)	Drafting the ToC
19.09.2023	0.2	T. Tran (TUDa)	Adding content and details for every section; minor changes in the outline;
23.01.2024	0.3	D. Petrak (TUDa)	Adding more details for experimental settings and results;
23.01.2024	0.4	T. Tran (TUDa)	Reviewing content
29.01.2024	0.5	D. Petrak (TUDa)	Update
21.02.2024	0.6	T. Tran (TUDa)	Updating the content according to comments
02.03.2024	0.7	TUDa	Collecting contributions from other partners
13.03.2024	0.8	TUDa	Further editing according to comments
23.03.2024	1.0	TUDa	Last major update
25.03.2024	1.1	TUDa	Final version

DISSEMINATION LEVEL		
PU	Public	x
PP	Restricted to other programme participants (including the EC services)	
RE	Restricted to a group specified by the consortium (including the EC services)	
CO	Confidential, only for the members of the consortium (including the EC)	

SERMAS partners

Disclaimer

This project has received funding from the Horizon Europe programme under the Grant Agreement 101070351.

Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or European Commission. Neither the European Union nor the European Commission can be held responsible for them.

Public Executive Summary

This document presents the overall dialogue management module developed in work package 5 of the SERMAS project, from a framework for synthetic dialogue data generation using large language models to several experiments of dialogue models, as well as our findings and discussion of future work.

The document starts with the introduction of the dialogue module within SERMAS Toolkit, its aspects connected to the concept of user acceptance, and the requirements of the SERMAS pilots that have been used in work package 5.

In this work, we consider real-world scenarios that a large number of dialogue data are hard to obtain for training and a small set of human-simulated dialogues can be collected for evaluation (as in D5.1). We emphasize the importance of three factors linked to user acceptance: (1) socio demographics, (2) user emotions, and (3) implicit user feedback. These factors are considered in our data generation framework as well as our experiments.

The module can generate synthetic dialogue data given task descriptions, reducing the cost of data collection. This is achieved by prompting a large language model with background information (Section 4). The module can also be used to annotate dialogue data given the descriptions of required information (namely intent, slot, emotion, implicit user feedback) (Section 4). The module may possibly make some incorrect annotations. However, our curation study shows that their number is relatively small. Also, human annotators can identify errors more quickly than come up with new dialogues or annotations from scratch. Both lead to lower cost of dialogue data creation and annotation when using our framework. Following the framework, the statistics and analysis of the generated data and annotations are presented.

After getting the data, we present the experiments carried out in this deliverable for developing the dialogue management module. The experiments are conducted using three representative state-of-the-art language generation models for comparison. We evaluate these models using both automatic and human evaluation metrics. We show that the three crucial factors of user acceptance are crucial for task completion and factual consistency of dialogue response generation models. Responses generated by the models trained with these factors also receive higher preference from human evaluators.

Table of contents

List of Figures

List of Tables

1. Introduction

A crucial aspect of the SERMAS XR agents is user acceptance, which has been studied a lot in human-computer interaction (Pelau et al., (2021), Araujo et al., (2018), Zamora, (2017), Ciechanowski et al., (2019)). There are three important factors link to user acceptance: (1) socio demographics, (2) user emotions, and (3) implicit user feedback. Sociodemographic information refers to the gender, age, occupation, etc of the user. User emotions refer to emotions expressed by user during the conversation, either via verbal or non-verbal signals. Implicit user feedback refers to user response to the preceding system utterance, such as a correction of the system utterance or a clarification question when the system utterance is unclear. While these factors are corelated with each other, they are considered separately in textual dialogue systems. In this work, we propose a dialogue management module to address this gap. The module is evaluated on the human created dataset from D5.1, which is related to the two SERMAS pilots: (II) Post Office Agent and (III) Receptionist Agent (D2.1). The dialogue module was also partially evaluated in the PoC Dialogue Management (D2.2).

To facilitate the training of dialogue models, we propose a framework for generating and annotating dialogue data. The annotations include the three crucial factors for user acceptance, namely demographic information, user emotions, and implicit user feedback. After getting the data, we train and evaluate different representatives of state-of-the-art language generation models. We evaluate these models using several metrics: (1) quality of the generated responses compared to references, (2) task completion rate, (3) factual consistency, and (4) toxicity of the generated responses.

The dialogue generation and annotation framework will be publicly released in a scientific publication as well as the generated dialogue data to facilitate future research in this direction. These resources can be accessed on our *GitHub repository*. The taxonomies of implicit user feedback have already been published (Petrak et al., (2023)) and can be accessed on the respective *GitHub repository*.

In the following sections, we first present the acronyms and terminology used in this report (Section 2). We briefly describe the requirements of the training data (Section 3) before present the data generation framework in Section 4. Section 5 gives the analysis of the generated data. Section 6 shows our experimental settings and results, which contains our training procedures and model hyperparameters. Finally, we conclude the deliverable with the achievements and follow-up work in Section 7.

The requirements in this deliverable follow the initial POSTE pilots, as the recent updates of the SERMAS pilots have been carried out after the data generation and annotation in this deliverable. The framework and code as part of the deliverable can be adapted to any further update of the dialogue module according to one's needs.

2. Terminology

To facilitate the reader in the following sections, we present the terms and their definitions used in this report (Table 1).

Term	Acronym	Definition
Application programming interface	API	The communication interface between components in the data collection platform.
Task-oriented dialogue	TOD	Task-oriented dialogues focus on supporting users to complete a particular goal.
Document-grounded dialogue	DocDial	Document-grounded dialogues focus on answering questions using information from an additional external knowledge source such as text documents.
Annotator		A human who plays the role of an agent or a user (of the agent).
User		The user who is going to interact with the agent.
Agent		The SERMAS agent who is going to support the user in completing service tasks, providing information, etc.
Intent		The goal of an input from the user, such as getting access to a building, seeking information of a product/service.
Slot		The attribute types or properties that are required to fulfill user intents, such as the name of the user for building access or the name of the internal host.

Slot value		The actual attribute value of a slot, such as "Paul" as the name of the user/speaker.
Turn		A pair of user and agent utterances.
Human-Computer Interaction	HCI	The field of research that focuses on understanding and optimizing how users and virtual agents interact.
Natural Language Processing	NLP	The field of research that uses and optimize machine learning to understand and produce natural language.

Table 1: Terminology.

3. Task Descriptions, Annotations and Formulation

This section provides a summary of the data requirements and annotations of our human-generated dialogue data (described in D5.1). The data requirements and annotations are refined and used in our proposed framework to generate synthetic dialogue data (Section 4). Section 3.1 and 3.2 present the refined descriptions. In addition to the annotations described in D5.1, we introduce social demographics as additional background information for dialogue generation in Section 3.2.1. Lastly, Section 3.3 describes the task formulation of dialogue response generation.

3.1. Task Descriptions

The SERMAS XR agents cover task-oriented document-grounded dialogues from three domains, including post office services, receptionist services and customer services in the insurance domain. For post office services, we consider dialogues about (1) customer support for parcel shipping, i.e., guiding a user through the process of sending a parcel with related information, and (2) topping up a prepaid SIM card. For reception services, we include the topic of building access control, i.e., the reception and registration of visitors in an office building. For customer services in the insurance domain, we include question answering dialogues about different types of insurance, such as pet, health, heritage, and finance. The question answering dialogues are additionally annotated with documents providing the knowledge required for response generation.

3.1.1. Post Office Services

For post office services, we include dialogues about parcel shipping and topping up a prepaid SIM card. In customer support dialogues for parcel shipping, the task is to help the user choose the right shipping box and delivery option for their needs (given the weight of the goods to be sent and the destination). Topping up a prepaid SIM card is less of an advisory

service but a task completion service, since customers usually know how much they want to recharge, their telephone number, and which telephone provider they are with. Table 2 lists the annotation slots for each task.

Slot Name	Description
Parcel Shipping	
Destination	The city and country of destination; national or international.
Weight	The weight of the item to be shipped, lightweight (up to 5kg), average (up to 20kg), heavy (up to 30kg).
Package Required	Whether or not a new shipping box is required.
Delivery Option	Express or standard delivery.
Country of Destination	The destination country.
Shipping Box Name	Name of the most suitable shipping box (small-sized, medium-sized, large-sized), based on the weight of the item to be sent.
Shipping Box Description	Brief description on why the suggested shipping box is a good choice.
Shipping Procedure	Description of the shipping procedure (e.g., take the box to the counter…).
Shipping Time	Expected delivery time, one to three days for national, four to six days for European, and 3-4 weeks for international deliveries.
Top Up Prepaid SIM Card	

Phone Number	Table or mobile phone number with country code, e.g., +39 XXX XXXXXXX.
Phone Provider	The phone provider, e.g., Vodafone, POSTE Mobile, …
Import Payment	The phone provider, e.g., 10€, 20€, 30€.
Outcome Operation	If all required information were provided, the system asks the user to insert the card for payment.
Request Ticket	
Type of Service	The type of service for which the user wants to request support, i.e., Parcel Shipping or Top Up Prepaid SIM Card.
Ticket Number	The ticket number generated for the request.

Table 2: Slot values for parcel shipping, top up prepaid SIM card and request ticket.

3.1.2. Receptionist Service

Another task we consider is Access Control as a receptionist service. This is an essential task in hotels, office buildings, or other facilities with restricted access. Visitors usually need to register at the reception desk before being allowed to enter the buildings. In our case, we focus on a scenario in which a visitor has an appointment with an employee in an office building. To access the building, the visitor needs to provide information about the appointment, e.g., the name of the host, date and time, and the room number. The access can be granted after the information can be validated or the host can confirm the visitor. The visitor can also request additional

safety information such as structural safety or fire safety. Table 3 shows the slots for this task.

Slot Name	Description
Guest Name	The name of the person who wants to access the building.
Host Name	The name of the person the guest wants to visit.
Host E-Mail	The e-mail address of the host.
Alternative Host Name	An alternative host, e.g., in case the host is not available today.
Alternative Host E-Mail	E-Mail address of the alternative host.
Meeting Date and Time	Date and time of the appointment.
Meeting Room Identifier	Room number in the building where the appointment takes place.
User Verification	The system can set up a verification call to let the host visually inspect the guest and authorize access.
Confirmation to open Turnstile	This is a signal to the system that controls the turnstile to let the guest enter.
Additional Safety Information	Any additional safety information, e.g., related to COVID-19.

Table 3: Slot values for access control.

3.1.3. Customer Service

For customer service, we focus on question answering in the context of the POSTE Italiane products (such as account conditions) and insurance

policies[1] (e.g., pet, health, or heritage insurance). Customers can often call their insurance agent or visited their local bank for questions related to such topics. In this work, we develop service agents to serve as first contact for customers. The agents can be always available around the clock and only certain cases will be redirected and handled by human employees. The slots for such customer service are presented in Table 4.

Slot Name	Descriptions
Question	A question related to one of the topics.
Type of Bills	If the user asks a question regarding a specific payment slip, they need to provide the type.
Evidence	The answer to the user's question.
Bill Form Description	Description of the specific payment form (if the question was about a payment form).
Bill Form Name	Name of the payment form (if the question was about a payment form).
Bill Form Payment Procedure	Information on how to fill in the payment form (if the question was about a payment form).

Table 4: Slot values for question answering.

3.2. Dialogue Annotations

3.2.1. Demographic Information

We consider gender, age, occupation, name, and language style as demographic information in this work. Overall, we use 1,000 common names collected from the Internet. We include 12 different language styles such as style matching Age and Job, Standard, Formal, Polite, Informal, Dialect, etc. Five generations are considered in this work, including Boomers

[1] *POSTE Italiane Service and Insurance Policies (English)* (last accessed 10 January 2024).

(born between 1952 and 1962), Generation X (born between 1962 and 1977), Millenials (born between 1977 and 1992), Generation Z (born between 1992 and 2007), and Generation Alpha (born between 2007 and 2016). Also, we include an extensive variety of occupations (overall 1,155), sampled from The Gazette[2], spanning various areas such as science and technology, education, arts and entertainment, healthcare, or manufacturing.

3.2.2. User Emotions

We use the emotion taxonomy from EmotionLines (Hsu et al., (2018)), which covers seven different emotions, including Neutral, Joy (which we refer to as Happiness), Sadness, Surprise, Fear, Anger, and Disgust. We extend the list with four other relevant emotion types (Kim et al., (2023); Rashkin et al., (2019)), including Confusion, Curiosity, Frustration, and Stress. Among these emotions, we consider Confusion, Frustration, Fear, Sadness, Disgust, Stress, and Anger as negative emotions.

3.2.3. Implicit User Feedback

For the generation and annotation of implicit user feedback, we use the user reaction type taxonomy proposed by Petrak et al., (2023), which distinguishes five user reaction types in response to generation errors in preceding system utterances (listed in Table 5).

User Reaction Type	Description
Ignore and Continue	The user ignores the error and continues the conversation, e.g., "Okay. Let's leave it like that."
Repeat and Rephrase	Instead of ignoring the error in the system utterance, the user repeats or rephrases their

[2] Available in *GitHub* (last accessed on 31 July 2023).

	original concern, e.g., "Actually, I wanted you to ...".
Make Aware with Correction	The user makes the system aware of its error and provides a correction or response alternative, e.g., "Partly. This doesn't take into account that ...".
Make Aware without Correction	Instead of providing a correction or response alternative, the user just makes the system aware of its error, e.g., "You're wrong.".
Ask for Clarification	In case of error, the user asks the system for clarification, e.g., "I'm not sure what you mean. Is it about ...".

Table 5. User Reaction Types by Petrak et al., (2023).

For generation errors in system utterances, we also propose an error taxonomy of ten types, nine of which are relevant for task-oriented document-grounded dialogues in Petrak et al. (2023) (Table 6).

Generation Error	Description
Ignore Question	This error occurs when the system fails to address the user's question. Instead of providing a relevant response or clarification, the system disregards the user's input.
Ignore Request	A situation where the system fails to act on a user's request. It can occur due to various reasons, such as misinterpretation of the request, technical limitations, or system glitches.
Ignore Expectation	This error happens when the system fails to fulfill the user's expectations in terms of understanding and addressing their needs or

	requests accurately withing the context of the task.
Attribute Error	If the system fails to correctly extract or understand the necessary slots or attributes from the user's utterance, this is called an attribute error.
Factually Incorrect	System responses that are factually wrong or inaccurate.
Topic Transition Error	A situation in which the system's response abruptly shifts to a different or previously discussed topic without a logical connection or adequate context.
Conversationality	Bad conversationality occurs when the system fails to maintain a coherent and natural conversation flow, e.g., it repeats previous responses or contradicts itself without recognizing or asking for new or missing information.
Unclear Intention	This error is characterized by the robot's failure to accurately address the user's intended objective.
Lack of Sociality	If a system's response doesn't adhere to social conventions, fails to include basic greetings, or exhibit toxic and disrespectful behavior or language, this is called Lack of Sociality.

Table 6: Generation errors in system utterances (Petrak et al., (2023))

3.2.4. Documents

For generating question answering dialogues, we use question-paragraph pairs extracted from the POSTE Italiane insurance and service policies

(English only), including pet, health, and heritage insurance, as well as bank and account conditions. Overall, we extracted 100 question-paragraph pairs for bank transactions and account conditions, 78 for health, 84 for heritage, and 39 for pet insurance from 316 Word documents. We first converted the Word documents into html files then excluded figures and tables. Next, we separated the html files by paragraphs. Most of the paragraphs have a corresponding heading in the form of a question such as "What is this insurance about?", we thus extracted them as question-paragraph pairs. In a final preprocessing step, we manually checked whether the questions matched the paragraphs and cleaned up the text. Hereinafter, we use the term "knowledge documents" when referring to the extracted paragraphs.

3.3. Task Formulation

We define a dialogue as a set of multiple turns T. Each turn consists of two utterances, a user utterance U_t and a system utterance S_t. Given a dialogue context $C = [T_0, \ldots, T_{t-1}]$, and additional background information K, the task is to predict user intent I_t, dialogue belief state B_t and system utterance S_t:

$$(I_t, B_t, S_t) = generate(K, C, U_t)$$

Equation 1:Task Formulation

Additional background information $K = \{D_t, DI, E_t, GE_t, F_t\}$ can be a (knowledge) document D_t, user demographic information DI, user emotion E_t, generation error GE_t, or implicit user feedback F_t. Belief state B_t consists of slot values extracted from the dialogue context C, which may be used to query knowledge from a database or a set of documents (Chen et al., (2022)), such as a document D_t containing information about insurance.

4. Framework for Generating and Annotating Dialogues Using Large Language Models

Dialogue data collection and annotation is usually resource-intensive and even more demanding when user feedback is collected. This process requires prolonged interactions between humans and agents. In previous work, dialogue datasets were often collected through crowdsourcing, which can result in poor quality due to methodological artefacts or annotator biases. Recent works suggest synthetic data generation with large language models as a more cost-efficient alternative, which can also lead to diverse and high-quality data. However, synthetically generated data usually comes with the risks of (1) hallucinated facts or (2) harmful content such as disrespectful and toxic.

We propose a synthetic approach to generate and annotate dialogue data for training the SERMAS XR agents, including demographic information, user emotions and implicit user feedback. Due to the potential limitations of synthetic data, we recruited human annotators for quality assessment, curation, and used the data collected for WP5.1 as test set in our experiments. To identify the best possible LLM for generating synthetic data, we conducted a preliminary study with various available models to generate and annotate different dialogues and asked human annotators to evaluate their quality.

4.1. Preliminary Study

In this study, we compare the utterances of 50 generated dialogues from two different models, namely GPT-3.5-Turbo from OpenAI and LLaMA-30B from Meta AI, for (1) human-likeness (Naturalness), (2) relevancy in the dialogue context (Coherence), (3) Engagement, (4) Task Coverage, (5) Length. The models differ in terms of size (GPT-3.5-Turbo has 175B parameters and LLaMA 30B parameters) and length of context window (4k

tokens in the case of GPT-3.5-Turbo and 2k tokens in the case of LLaMA-30B). In addition, LLaMA-30B is open-sourced, thus available for local deployment, while GPT-3.5-Turbo is only accessible through a commercial API. Table 7 shows the average results over all dialogues.

Model	Natural-ness	Coherence	Engagement	Task Coverage	Length
LLaMA-30B	3.12	3.52	0.8	3.52	3,24
GPT-3.5-Turbo	4.40	4.92	1.0	4.68	7,12

Table 7: Results of our preliminary study to assess the dialogue generation capabilities of LLaMA-30B and GPT-3.5-Turbo.

Naturalness, Coherence and Task Coverage are measured in a Likert-Scale from 1 (lowest rating) to 5 (highest rating). Length measures the average number of turns per dialogue. In general, annotators rated the GPT-3.5-Turbo dialogues higher. Task Coverage is the most important aspect in our work since incomplete data could often lead to low model performance and thus weaken user experience. Based on these results, we decided to use GPT-3.5-Turbo for the generation and annotation of synthetic dialogue data.

4.2. Synthetic Data Construction

Figure 1 gives an overview of our framework for generating and annotating dialogues. We distinguish feedback dialogues that contain annotations for implicit user feedback and feedback-free dialogues. However, the procedure for dialogue and annotation generation is the same for both types of dialogues.

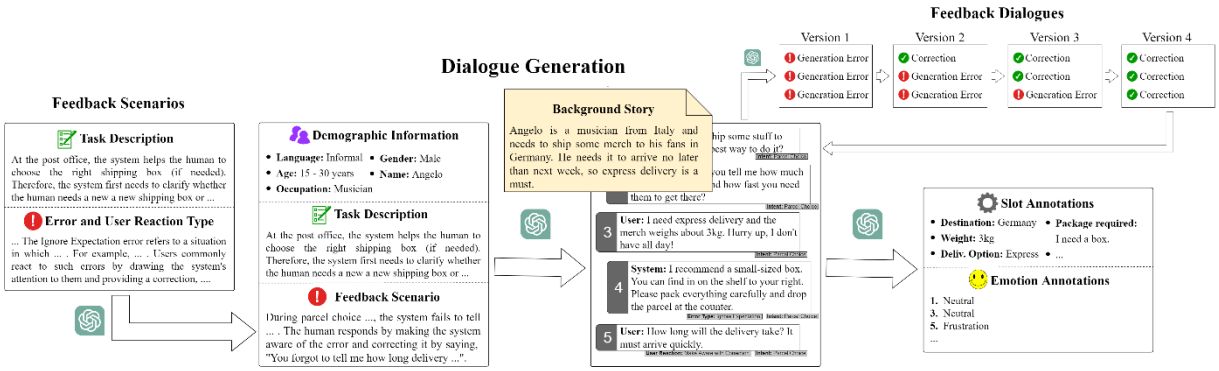

Figure 1: Overview of our framework for the generation and annotation of dialogues. In general, we distinguish feedback and feedback-free dialogues. Feedback scenarios are specific to feedback dialogues.

For each step that involves GPT-3.5-Turbo, we require the model to return the results in a predefined JSON scheme. The JSON scheme depends on the generation step, i.e., dialogue or annotation generation, and ensures that the returned values contain all required fields and are processable without human intervention. If a certain step in the generation process does not match this requirement, the whole dialogue is discarded.

4.2.1. General Approach to Dialogue Generation

For dialogue generation, we provide GPT-3.5-Turbo with randomly sampled demographic information for the user, a task description, and the role of the starting actor, i.e., user or system. As indicated by the boxes on the left side of Figure 1, a task description describes the flow of events and information that needs to be conveyed by each speaker to fulfill the task. In the case of question answering, the description also includes a randomly sampled list of documents from the respective topic. We then instruct the model to use the task description and the demographic information to generate a background story for the conversation, depicted in the center of Figure 1. We also instruct the model to return the utterance-level annotations for intents and limit the dialogue to 13 turns, since we found that longer dialogues tend to deviate from the task description. For

background stories, we limit the length to five sentences to avoid them becoming a distraction.

Figure 2 shows the prompt used for generating feedback-free dialogues. The generation of feedback dialogues is discussed in Section 4.2.2.

> Generate a dialogue (max. 13 turns) between a human and a dialogue system in the following task: {name of the task}. For the human, imagine a person ({occupation}, between {age} years old) called {name} that uses {language} language style with a short emotional and task-related background story of max. 5 sentences (including the human's country of residence). Generate the dialogue in a role-play manner. The dialogue system is empathetic and replies and interacts with the human according to their persona and background story. Do not include personal information (e.g., the person's name) in the dialogue. The {role of the starting actor} starts. The conversation begins and ends with a greeting.
> {task description}
> For each utterance, include the intent (the task addressed) in the json output.

Figure 2: Instruction for the generation of (feedback-free) dialogues.

4.2.1.1. Annotation Generation

For slot annotations, we provide GPT-3.5-Turbo with the generated dialogue and a list of all slots defined in the task description, possible values, and examples. We also tried to reduce the number of API calls by generating dialogue and annotation in one step, but this shortcut does not produce reliable results. We also instruct the model to only assign and copy values from the dialogue (to prevent hallucinations) and to return the annotations on utterance-level. Figure 3 presents the prompt used for slot annotation.

Given is the following dialogue between a dialogue system and a person:
{dialogue}
Identify and copy the corresponding sequences for each of the following slots in the person utterances: {list of slots in person utterances with examples}. Identify and copy the corresponding sequences for each of the following slots in the system utterances: {list of slots in system utterances with examples}.

Figure 3: Instruction for the generation of slot annotations.

For emotion annotations, we instruct the model to predict the emotion for each user utterance in the dialogue, given the dialogue and our emotion taxonomy.

4.2.2. Generation of Feedback Dialogues

4.2.2.1. Feedback Scenarios

For each feedback dialogue, we first generate three feedback scenarios that are then used as an additional source for dialogue generation (left side of Figure 1). A feedback scenario describes a generation error and the following implicit user feedback. We generate all feedback scenarios for a dialogue at once, using the same API call. For this, we provide GPT-3.5-Turbo with the task description and a list of randomly sampled error and user reaction types. To ensure coherence, we add the constraint that feedback scenarios must be non-mutually exclusive and together form a story in the context of the task description. We limit the number of feedback scenarios per dialogue to not exceed the length of 13 turns per dialogue and to leave room for the start and end of the conversation. Figure 4 shows the prompt used for generating feedback scenarios.

{names of error types} are common errors in dialogues.
{list of error type definitions}
Users commonly react to such errors by {user reaction types}.
Combine each of these user reaction types with an error type.
Then generate a feedback situation (up to 4 sentences, including why and how it reflects the respective error type) for 3 of these combinations in the following task:
{task description}
It is important that the feedback situations are different but not mutually exclusive and together make a story. For each feedback situation, provide a precise description as continuous text (no dialogues), including the user's reaction and why and how the situation reflects the respective error type.

Figure 4: Instruction for the generation of feedback scenarios.

4.2.2.2. Feedback Dialogues

For feedback dialogue generation, we instruct GPT-3.5-Turbo to consider each feedback scenario in three utterances in the generated dialogue, in addition to the constraints described in Section 4.2.1: The system utterance with the generation error, a subsequent user utterance that reflects the user reaction, and a following system utterance that addresses the user reaction. We consider the generated dialogue as Version 1 and generate three additional versions of the same dialogue, each resolving one of the feedback scenarios (upper right side of Figure 1). For each version, we first mask the affected system utterance and generate a replacement using the task description and the preceding dialogue context. Next, we drop the following two utterances since they are directly related to the generation error. This way, the conversation continues with the next regular user utterance. We continue the process until all feedback turns have been resolved as in Version 4. For slot values, we only regenerate the annotations for the replaced system utterances in Version 2 to 4 and retain the other

annotations from Version 1. Figure 5 presents the prompt for generating feedback dialogues.

Generate an erroneous long and in-depth dialogue (at least 13 utterances) between a human and a dialogue system. For the human, imagine a person ({occupation}, between {age} years old) called {name} that uses {language} language style with a short emotional and task-related background story of max. 5 sentences (including the human's country of residence). Generate the dialogue in a role-play manner. Play the dialogue system as not helpful and inattentive. Do not include personal information (e.g., the person's name) in the dialog. The {role of the starting actor} starts. The conversation begins and ends with a greeting.
{task description}
An feedback situation consists of a system utterance, in which the dialogue system makes an erroneous statement, and a subsequent human utterance, in which the human reacts to the error in the system utterance in the predefined way. Next, the system responds considering the reaction of the person. Then the situation is done. Generate the dialogue using the following {number} error situations (all must be included): {error situations}
Highlight the erroneous system utterance by adding the respective scenario identifier to the error field of the utterance and to the error field of the following person utterance. Errors always originate from system utterances. Each scenario can only occur twice, once in a system utterance and once in the subsequent human utterance.

Figure 5:Instruction for the generation of feedback dialogues.

5. Analysis of the Generated Dialogues

Overall, we generated 8,526 dialogues, divided into 1,662 feedback-free and 6,864 feedback dialogues (1,716 in four versions, each with one feedback scenario less per dialogue). Table 8 shows the data distribution across different tasks.

Task	Feedback-Free	Feedback Dialogues			
		Version 1	Version 2	Version 3	Version 4
Parcel Shipping	206	214	214	214	214
Top Up SIM Card	207	214	214	214	214
Access Control	203	238	238	238	238
Question Answering	1,046	1,050	1,050	1,050	1,050
Total	1,662				6,864

Table 8: Distribution across tasks for generated dialogues.

In the following, we focus on analyzing the completeness of generated slot and intent annotations, the distribution of demographic information, user emotions and feedback scenarios represented in the dialogues.

5.1. Slot and Intent Annotations

Table 9 shows the ratio of dialogues for which intent and slot annotations were successful, i.e., dialogues that provide all annotations for intent and required slot values.

Task	Feedback-Free	Feedback Dialogues			
		Version 1	Version 2	Version 3	Version 4
Parcel Shipping	0.87	0.74	0.72	0.70	0.70
Top Up SIM Card	0.87	0.74	0.72	0.71	0.69
Access Control	0.86	0.82	0.83	0.84	0.84
Question Answering	0.99	0.73	0.99	0.99	0.99

Table 9: The ratio of dialogues that are complete in the sense that they are annotated with all intent and slot values. Hallucinated slot values, i.e., slot annotations that do not occur in the corresponding utterance, are considered as missing.

We observe large differences between question answering and the other tasks. We found that this is mostly due to variations in the slot annotations. While the slot annotation scheme for question answering is rather simple (Section 3.1), this is different for other tasks where slots often depend on the background story. For example, in the case of parcel shipping, if the user already has a shipping box and just requires information on the shipping procedure, details about available shipping box types are negligible. For feedback dialogues, we observe that the generated corrections do not always address the missing information required by the task description.

5.2. Demographic Annotations

Figure 6 shows the distribution of language styles, age ranges and occupations randomly sampled for background story generation.

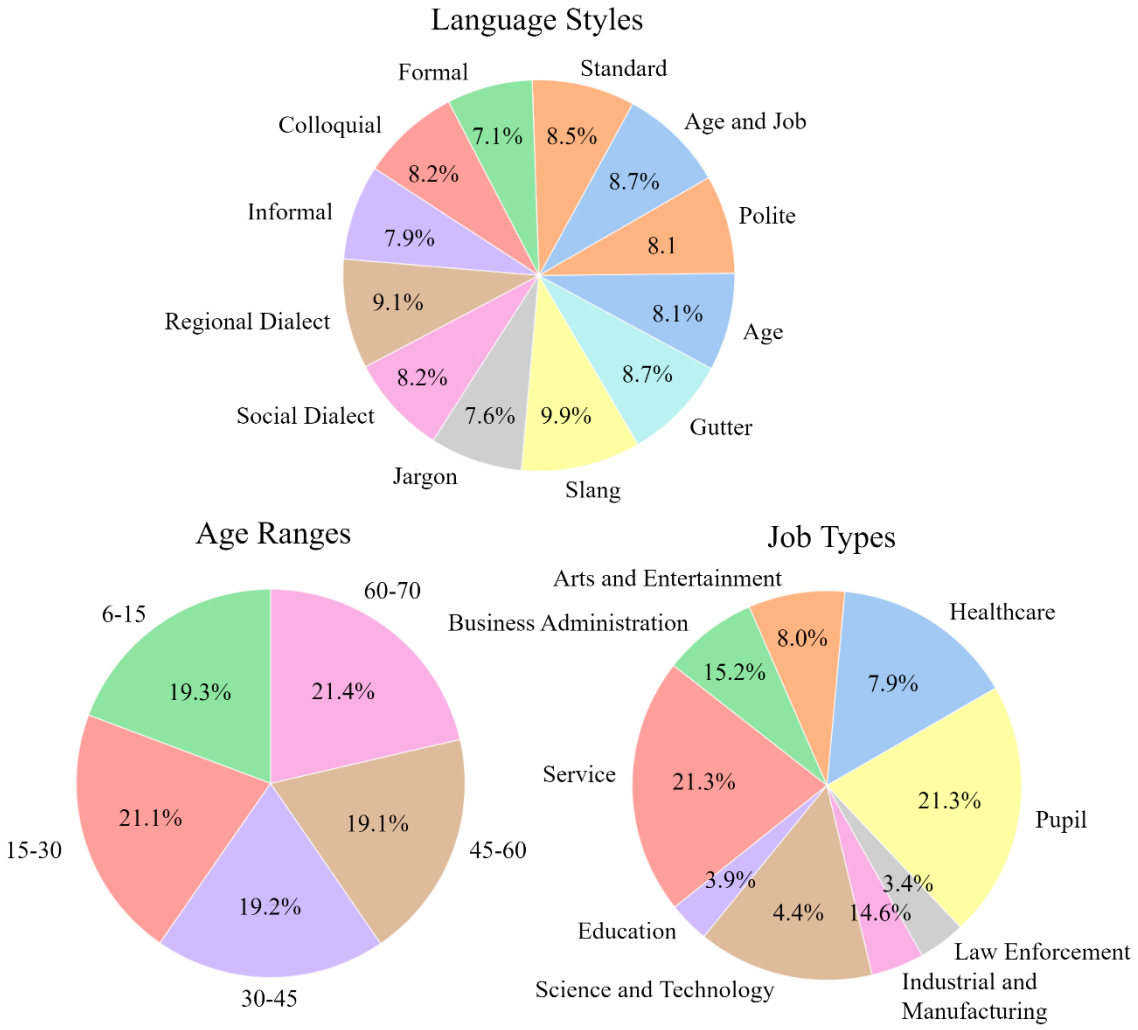

Figure 6: The distribution of language styles, occupations, and age ranges in the generated dialogues.

Language styles are almost equally weighted. For occupations (jobs), the figure shows that jobs from the categories of business administration, service, industrial and manufacturing, and pupil largely outweigh the other categories, which makes sense in the context of the tasks and topics represented in the generated dialogues. Overall, we observe 693 unique job titles. The figure do not show the distribution of names due to the large number of them. There are 1,496 unique names in total included in the generated dialogues. 638 (42%) occur only once and 712 (47.59%) occur two to three times. The remaining 146 names occur four or more times throughout the entire dataset.

5.3. Emotion Annotations

The chart in Figure 7 shows the distribution of emotions in the dialogues. With 40.5%, Neutral is the most common emotion, followed by Curiosity (27.5%). Frustration and Confusion are relatively rare. As expected, we observe them mostly in feedback dialogues. The category Others refers to emotions that present less than or equal to 5%, including Anger, Disgust, Fear, Surprise, and Stress.

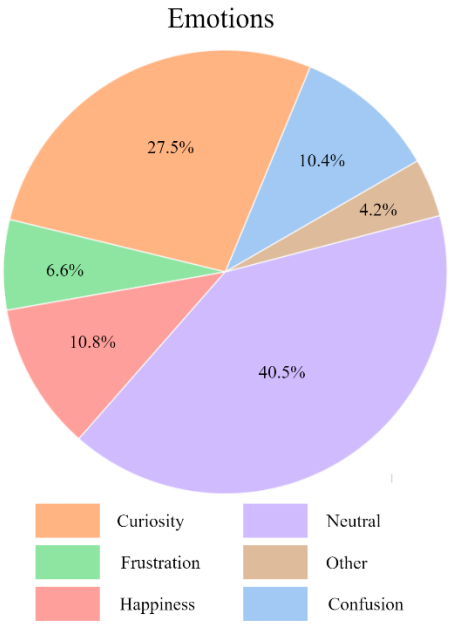

Figure 7: Distribution of emotions in the generated dialogues.

Figure 8 illustrates the distribution of the five most common emotions observed in user utterances from both the feedback-free and feedback dialogues (excluding Neutral). As expected, negative emotions are more common in feedback dialogues. For Curiosity, we found that the polarity depends on the dialogue context, for example, whether the previous system utterance successfully addressed the user's request. Curiosity is an emotion that can be either positive or negative, thus it is frequently observed in both dialogue types. Happiness in feedback dialogues is mostly observed as a reaction to system utterances that implement user feedback.

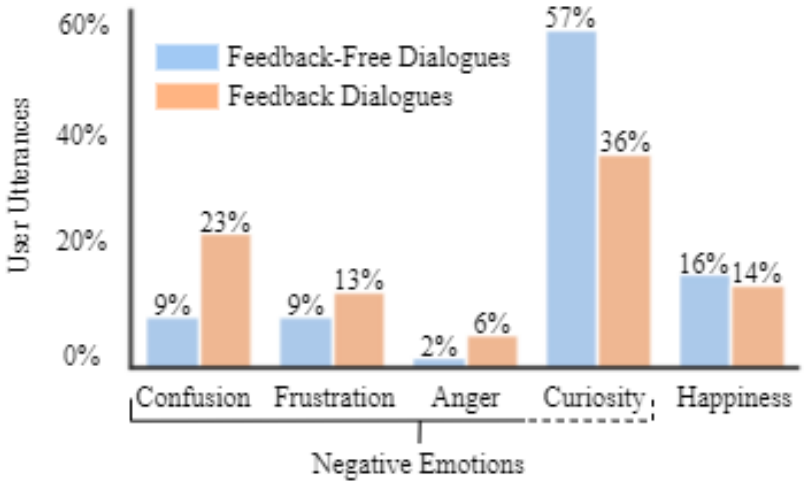

Figure 8: Ratio of the most observed user emotions in feedback and feedback-free dialogues.

5.4. Feedback Scenarios

Overall, we generated 4,714 feedback scenarios included in the 1,716 feedback dialogues of Version 1. Figure 9 shows the distribution of error and user reaction types.

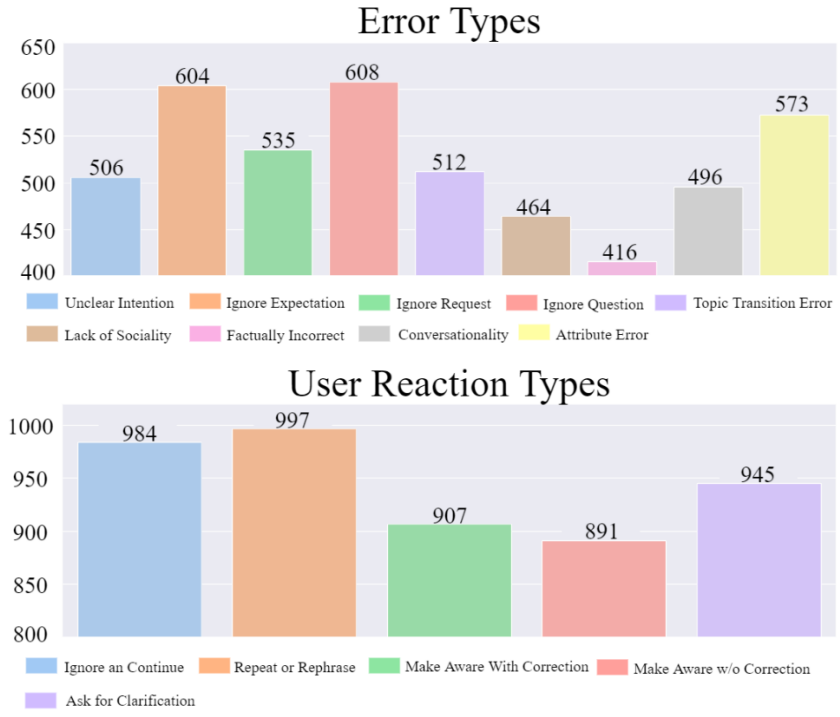

Figure 9: Distribution of error and user reaction types in the feedback dialogues.

Given that most of the dialogues are about question answering (Table 8), it is not surprising that Ignore Question is the most frequent error type. Figure 10 shows the distribution of user reactions in relation to error types represented in the feedback scenarios of the feedback dialogues.

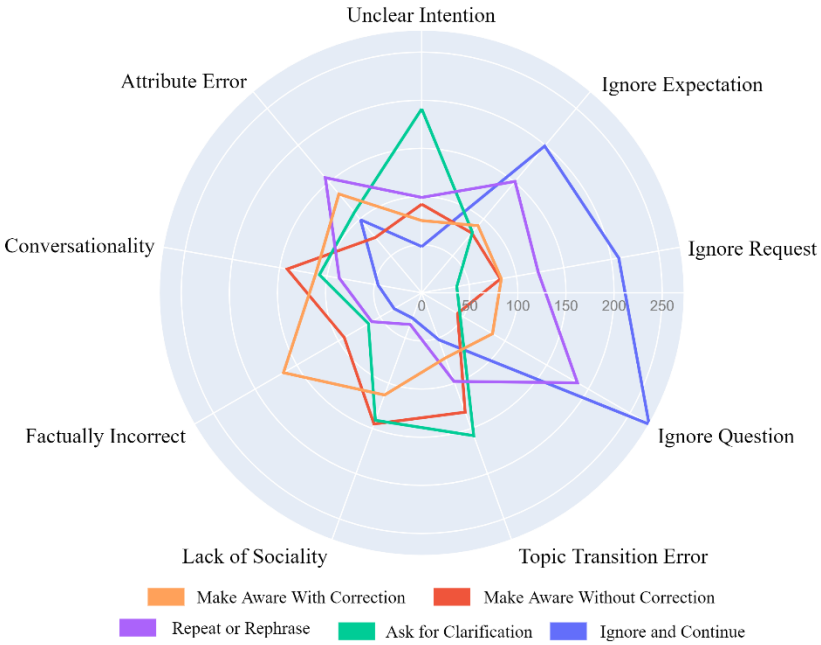

Figure 10: Distribution of user reactions in relation to error types represented in feedback scenarios.

The figure shows that our approach for generating feedback scenarios resulted in meaningful combinations of error and user reaction types. For example, Factually Incorrect is mostly addressed by Make Aware with Correction. The same applies to Unclear Intention and Attribute Error, which are mostly addressed by Ask for Clarification and Repeat or Rephrase. The latter one is also frequently observed in combination with Ignore Question and Ignore Expectation errors. Table 10 shows the ten most common error and user reaction type combinations.

	Error Type	Feedback Type	Frequency
1	Ignore Question	Ignore and Continue	273
2	Ignore Request	Ignore and Continue	208

3	Ignore Expectation	Ignore and Continue	199
4	Unclear Intention	Ask for Clarification	191
5	Ignore Question	Repeat or Rephrase	187
6	Factually Incorrect	Make Aware with Correction	166
7	Topic Transition Error	Ask for Clarification	158
8	Attribute Error	Repeat or Rephrase	156
9	Ignore Expectation	Repeat or Rephrase	151
10	Lack of Sociality	Make Aware without Correction	141

Table 10:The most common error and user reaction type combinations included in the feedback dialogues.

Ignore Question and Ignore Request are two of the most frequent error types. While we observe the first one more common in question answering dialogues, the second one is more common in the other tasks. For both we observe that Ignore and Continue is the most frequent user reaction type, followed by Repeat or Rephrase. Unclear Intention is an error type mostly observed in parcel shipping, top up prepaid SIM card, and access control. The most frequently observed user reaction to this is Ask for Clarification. Factually Incorrect is the rarest error type, which is mostly seen in question answering and in combination with Make Aware With Correction.

5.5. Human Evaluation

We asked two student assistants from our lab to assess and curate the intent, slot and emotion annotations in 480 feedback-free dialogues and the error and user reaction type annotations in 380 feedback dialogues. We used INCEpTION (Klie et al., (2018)) as the data platform for this study. We calculated the agreement between the annotators using Krippendorff's Alpha (Krippendorff, (2004) as provided in the INCEpTION platform. Table 11 shows the results.

	Annotation Type	Missing	Changed	IAA
Feedback-Free Dialogues	Intent	6%	35%	0.90
	Slots	56%	19%	0.83
	User Emotions	2%	81%	0.91
Feedback Dialogues	Error Type	16%	36%	0.97
	User Reaction Type	16%	34%	0.89

Table 11: The ratio of dialogues with at least on missing or changed annotation in our human evaluation study.

26 dialogues were reported as off-topic and are not considered in these results. Overall, the ratio of dialogues with at least one missing annotation is low, except for slot annotations. We found that most of them are parcel shipping dialogues, which have a comparatively complex annotation scheme (see Section 3.1.1). A detailed analysis revealed that an average of 1.8 annotations were added to the dialogues, most of them (36%) were slot annotations. For the dialogues with at least one changed annotation, we found that in many of these cases placeholders such as the slot name put in brackets ([shipping_box_name]) were used instead of the slot values from the dialogues (reported by the students). Emotion is the most frequently changed annotation type (on average 2.09 times per affected dialogue), with the originally annotated emotion often being Neutral.

6. Experiments

We implement three common state-of-the-art language generation models of different architectures and pretraining approaches and train them on the data generated in Section 4. These models include FLAN-T5 (780M parameters) (Chung et al., (2022)), GPT-2 (780M parameters) (Radford et al., (2019)) and LLaMA-2 (7B parameters) (Touvron et al., (2023)). FLAN-T5 is an instruction-tuned version of T5 (Raffel et al., (2020)), which is an encoder-decoder model. GPT-2 and LLaMA-2 are decoder-only models. While FLAN-T5 was pretrained in the task of sequence-to-sequence generation, GPT-2 was pretrained to predict the next token and LLaMA-2's objective was to follow instructions (like those used for dialogue generation in Section 4). The pretrained model weights for FLAN-T5 and GPT-2 are available in the Huggingface Model Hub[3]. Access to the weights for LLaMA-2 must be requested from Meta AI[4]. While we fully finetune FLAN-T5 and GPT-2, we only finetune the LoRA (Hu et al., (2021)) weights for LLaMA-2.

6.1. Experimental Settings

We first finetune the pretrained models to our scenario using the generated feedback-free dialogues (Feedback-Free in Table 12) and included the demographic information (+Demographics) and user emotions (+Emotions) as part of the input sequences. We then use the best performing feedback-free models (the bold ones) for experiments using the feedback dialogues (Feedback). For testing, we evaluate on the human-collected data from WP 5.1. Accordingly, we only report the results on this data.

[3] Pretrained model weights for *FLAN-T5* and *GPT-2* in the Huggingface Model Hub (last accessed 09 January 2024).

[4] *Form* for requesting access to the LLaMA-2 model weights (lase accessed 11 January 2024).

6.1.1. Input Sequences

Each model used in this work requires different formats of the input sequence. In general, the components of the input sequence depend on the features used (e.g., user emotions or demographic information). Figure 11 shows the input sequence used for training and inference using FLAN-T5 (Chung et al., (2020)).

<knowledge> {knowledge} <user_persona> {demographic information} <user_emotion> {emotion} <error_text> {error text} <user_reaction> {user reaction} <dialog> {context} </s>

Figure 11:Input sequence used for FLAN-T5. Additionally added source data is highlighted.

The target sequence includes the intent, slot values, and generated system utterance. It is basically the same as the last part of the input sequence for GPT-2 (Radford et al., (2019)), which is shown in Figure 12 (starting from <intent>, but without the special token).

<knowledge> {knowledge} <user_persona> {demographic information} <user_emotion> {emotion} <error_text> {error text} <user_reaction> {user reaction} <dialog> {context} <intent> {intent} <slots> {slots} <system> {target} <|endoftext|>

Figure 12: Input sequence used for training with GPT-2.

For inference with GPT-2, we use the same input sequence as for FLAN-T5 (Figure 11). For LLaMA-2 (Touvron et al., (2023)), Figure 13 shows the input sequence.

Given is the following task-oriented knowledge-grounded dialog (<dialog>) between a human user (<user>) and a virtual agent (<system>). Previously, this conversation went wrong because the virtual agent made a statement that was contextually incorrect ({error text}). The human user reacted accordingly ({user reaction}). Generate the user's intent (<intent>), extract the slot values (<slots>) and generate the next system utterance by considering the user's emotion ({emotion}), persona ({demographic information}) and the following document: {knowledge} <dialog> {context} <intent> {intent} <slots> {slots} <system> {target}

Figure 13: Input sequence for LLaMA-2.

6.1.2. Hyperparameters

For the experiments with feedback-free dialogues, we trained all models for five epochs, except for LLaMA-2 (Touvron et al., (2023)) since it took already five epochs to adapt the pretrained model to our prompting mechanism. For the experiment with feedback dialogues, we subsequently trained the best performing feedback-free models for ten epochs using the feedback data (ten epochs, since we have seen further improvements after the fifth epoch). We used a batch size of 32 and a learning rate of 5e-5 with no warmup steps. As optimizer, we used the implementation of AdamW (Loshchilov, (2019)) available in PyTorch[5]. Except for LLaMA-2, we fully-finetuned all models. For LLaMA-2, we only finetuned the LoRA (Hu et al., (2021)) weights, using a rank of 8, an alpha of 16, and a dropout rate of 0.05.

6.1.3. Evaluation Metrics

We use F1-Score (based on overlapping tokens in target and prediction), BLEU(-n) (Papineni et al., (2002)) and BertScore (Zhang et al., (2019)) to evaluate the concordance (Matching) of the generated system utterances with the target values. For BLEU and BertScore, we use the implementation from the HuggingFace Evaluation Library[6] and with $n = 4$ for BLEU.

[5] *PyTorch*, a Python package for machine learning (last accessed 10 January 2024).

[6] *Huggingface Evaluation Library* (last accessed 14 January 2024).

To evaluate for task completion, we use Inform and Success as proposed by Budzianowski et al., (2018), and measure the correctness of the predicted intents (Intent Accuracy) and slot values (Slot Accuracy). For Inform and Success, we use the implementation from Nekvinda et al., (2021), as a reference.

To measure the toxicity in the generated responses, we use Perspective API[7]. Perspective API is a free-to-use service provided by Google and Jigsaw. To measure the factual consistency in the case of question answering, we use Q^2 (Honovich et al., (2021)). For Q^2, we use reference implementation which is available in GitHub[8].

6.2. Results

Table 12 shows the results of our experiments. In general, we find that including user emotions (+Emotions) has a positive impact. This is particularly significant in the case of LLaMA-2 (Touvron et al., (2023)), where we used instructions for finetuning that provide additional context, i.e., a brief description of how the user emotion should be considered. In the feedback experiments, we observe great improvements in task completion and factual consistency of generated responses (Q^2 metric), which are both crucial for task-oriented document-grounded dialogues such as in the case of the SERMAS XR agents. For example, by including the generation error in the case of LLaMA-2 or by combining the generation error with the user reaction in the case of FLAN-T5 (Chung et al., (2022)) and GPT-2 (Radford et al., (2019)). In general, we attribute these improvements to the additional context provided by the generation error and the user reaction, which can be interpreted as a negative example for a response in the specific dialogue context.

[7] *Perspective API*. Model and training details can be found *here* (last accessed 16 January 2024).

[8] *Reference Implementation of Q^2* (last accessed 14 January 2024).

Experiment		Matching			Task Completion				Quality	
		F1	BLEU	Bert-Score	Inform	Success	Intent Acc	Slot Acc	Toxicity	Q²
FLAN-T5	FLAN-T5	45.0	20.0	88.3	86.7	85.9	54.8	60.9	0.02	52.7
Feedback-Free	+Emotion	46.7	21.0	88.9	83.9	83.2	61.2	58.3	0.02	57.5
	+Demo.	43.2	18.4	87.7	87.0	86.0	33.5	29.3	0.03	54.5
	+Emotion +Demo.	44.2	19.1	88.1	85.3	85.1	43.9	36.7	0.02	56.4
Feedback	+Gen.Err.	41.4	19.8	87.8	96.8	92.7	72.5	76.7	0.02	56.9
	+User Re.	41.3	19.3	87.6	96.6	94.1	69.0	76.2	0.02	56.3
	+Gen.Err. +User Re.	43.4	22.1	88.2	96.9	95.3	83.5	77.2	0.02	60.2
GPT-2	GPT-2	34.9	10.4	87.1	88.3	81.6	78.7	69.6	0.02	28.1
Feedback-Free	+Emotion	35.1	10.4	87.1	84.1	83.8	75.4	67.3	0.02	26.7
	+Demo.	34.6	10.4	87.1	80.2	80.2	69.3	57.5	0.02	26.3
	+Emotion +Demo.	36.0	11.4	87.3	85.1	84.8	71.6	66.7	0.02	29.2
Feedback	+Gen.Err.	29.2	8.0	86.2	92.4	91.7	84.3	79.3	0.02	30.9
	+User Re.	30.0	8.3	86.3	98.9	96.5	83.0	80.3	0.02	32.3
	+Gen.Err. +User Re.	30.3	9.7	86.4	94.7	93.3	88.0	80.8	0.01	35.5
LLaMA-2	LLaMA-2	29.3	7.1	86.1	85.9	81.2	37.6	39.2	0.02	28.3
Feedback-Free	+Emotion	36.3	14.9	85.4	89.3	85.3	40.2	41.3	0.01	18.7
	+Demo.	33.8	4.5	86.5	85.6	82.5	37.1	40.1	0.02	21.3
	+Emotion +Demo.	28.8	5.6	81.3	86.7	87.9	41.4	39.6	0.03	20.6
Feedback	+Gen.Err.	24.1	7.9	77.4	93.1	95.7	54.8	59.6	0.01	29.1
	+User Re.	24.1	7.9	77.4	93.1	95.7	54.8	59.6	0.02	27.1
	+Gen.Err. +User Re.	25.0	9.2	80.1	82.4	83.6	46.3	47.2	0.03	33.5

Table 12: Results of our experiments. We use the baseline models as deltas, i.e., the pretrained models finetuned on the generated feedback-free dialogues. The models with the greatest improvements are underlined. In general, improvements are highlighted in green. Deteriorations in red.

6.3. Human Evaluation

As part of D5.4, we conduct human evaluation on the fine-tuned models with the contributions from TUDa and POSTE. More details regarding the evaluation activities will be reported in D5.4. In this section, we present the results of the human evaluation.

We use the two best feedback-trained models from Table 12 (FLAN-T5 and GPT-2 with generation error and user reaction) and their feedback-free counterparts (FLAN-T5 with emotions and GPT-2 with user emotions and demographic information) to generate responses for 50 randomly chosen samples from the human-collected test set in WP 5.1. We then asked two participants from our lab (who participated during their working hours) to rate the generated responses for human-likeness (naturalness), relevancy in the dialogue context (coherence), social acceptability (safety), factual consistency (with the target document in the case of answering questions), and engagement (whether they would use this model in practice). We use the Likert scale from 1 (lowest rating) to 5 (highest rating) for each attribute and provide the annotators with the knowledge document, dialogue context, and generated response for this evaluation. Table 13 shows the results.

Experiment	Natural.	Cohere.	Safety	Engage.	Factual Consistency
FLAN-T5					
Feedback-Free	4.14	4.15	4.55	3.75	2.20
Feedback	4.30	4.25	4.59	3.89	2.24
GPT-2					
Feedback-Free	4.24	3.82	4.45	3.44	1.55
Feedback	4.42	4.05	4.46	3.76	1.58

Table 13: Results of our human evaluation. Improvements are highlighted in green.

The responses generated by the feedback-trained FLAN-T5 model are mostly rated higher by the participants. They also reported that the generated responses encourage more user interaction, e.g., by requesting additional information or paying more attention to the user and their situation and are in general more factual consistent. In the case of question answering, the generated responses are mostly summaries of the respective documents. According to the authors, the GPT-2 model trained only with emotions already produced very engaging answers, although they are not as coherent as the responses generated by FLAN-T5. This also affects factual consistency in the case of question answering, which is much lower for both GPT-2 models than for FLAN-T5.

7. Conclusion and Future Work

The goal of this deliverable is to develop the dialogue management component of the SERMAS XR agents. User acceptance and trust are important criteria for the success of SERMAS agents. We propose a framework that considers the social demographics of users, user emotions and implicit user feedback to generate synthetic dialogues. Our framework makes use of a large language model to generate and annotate dialogues for the SERMAS pilots. Our framework can be employed and extended to novel pilots with a significant reduction in data collection and annotation cost compared to recruiting human annotators (as compared to the collected data in D5.1). The generated data can be effectively used for training dialogue models as shown in the experiments.

Although a relatively low number of off-topic dialogues have been reported, the generated data may still contain unintended biases. A smaller human-generated dataset remains irreplaceable for evaluation. We adopted the human-generated dataset from WP5.1 as the test set for evaluation. Besides, since automatic evaluation cannot fully reflect the human judgment for text generation, we design and develop a web-based platform for human evaluation as presented in ToC Dialogue Management. This is also part of our work for D5.4 Validation.

For experiments, we employed three state-of-the-art language generation models, namely FLAN-T5, GPT-2 and LLaMA-2. Our results show that including demographic information and user emotions in general leads to better results than only fine-tuning the pretrained models on dialogue dataset. In addition, feedback data significantly improves task completion and factual consistency of the generated responses, which is crucial for task-oriented document-grounded dialogue systems. Our human evaluation also shows that responses generated by the feedback-trained models are in

general rated higher compared to responses generated by the feedback-free models.

Currently, the dialogue management component considers only verbal communication signals. User emotions have been currently considered as given textual descriptions, one can also detect the emotions from other modalities such as speech or vision. As discussed in D4.1 and D4.3, the messages between verbal and non-verbal signals can be exchanged as textual descriptions. While we have considered the potential non-verbal signals as input, generating non-verbal signals as responses to a user will be considered in future work (D5.3).

8. References

Pelau, C., Dabija, D. C., & Ene, I. (2021). *What makes an AI device human-like? The role of interaction quality, empathy and perceived psychological anthropomorphic characteristics in the acceptance of artificial intelligence in the service industry.* *Computers in Human Behavior*, *122*, 106855.

Jennifer Zamora. 2017. *I'm Sorry, Dave, I'm Afraid I Can't Do That: Chatbot Perception and Expectations.* In Proceedings of the 5th International Conference on Human Agent Interaction (HAI '17). Association for Computing Machinery, New York, NY, USA, 253–260. https://doi.org/10.1145/3125739.3125766

Ciechanowski, L., Przegalinska, A., Magnuski, M., & Gloor, P. (2019). *In the shades of the uncanny valley: An experimental study of human–chatbot interaction.* *Future Generation Computer Systems*, *92*, 539-548.

Araujo, T. (2018). *Living up to the chatbot hype: The influence of anthropomorphic design cues and communicative agency framing on conversational agent and company perceptions.* *Computers in Human Behavior*, *85*, 183-189.

Dominic Petrak, Nafise Moosavi, Ye Tian, Nikolai Rozanov, and Iryna Gurevych. 2023. *Learning From Free-Text Human Feedback – Collect New Datasets Or Extend Existing Ones?*. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 16259–16279, Singapore. Association for Computational Linguistics.

Chao-Chun Hsu, Sheng-Yeh Chen, Chuan-Chun Kuo, Ting-Hao Huang, and Lun-Wei Ku. 2018. *EmotionLines: An Emotion Corpus of Multi-Party Conversations*. In *Proceedings of the Eleventh International Conference on*

Language Resources and Evaluation (LREC 2018), Miyazaki, Japan. European Language Resources Association (ELRA).

Hyunwoo Kim, Jack Hessel, Liwei Jiang, Peter West, Ximing Lu, Youngjae Yu, Pei Zhou, Ronan Bras, Malihe Alikhani, Gunhee Kim, Maarten Sap, and Yejin Choi. 2023. *SODA: Million-scale Dialogue Distillation with Social Commonsense Contextualization*. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12930–12949, Singapore. Association for Computational Linguistics.

Hannah Rashkin, Eric Michael Smith, Margaret Li, and Y-Lan Boureau. 2019. *Towards Empathetic Open-domain Conversation Models: A New Benchmark and Dataset*. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5370–5381, Florence, Italy. Association for Computational Linguistics.

Zhiyu Chen, Bing Liu, Seungwhan Moon, Chinnadhurai Sankar, Paul Crook, and William Yang Wang. 2022. *KETOD: Knowledge-Enriched Task-Oriented Dialogue*. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 2581–2593, Seattle, United States. Association for Computational Linguistics.

Jan-Christoph Klie, Michael Bugert, Beto Boullosa, Richard Eckart de Castilho, and Iryna Gurevych. 2018. *The INCEpTION Platform: Machine-Assisted and Knowledge-Oriented Interactive Annotation*. In *Proceedings of the 27th International Conference on Computational Linguistics: System Demonstrations*, pages 5–9, Santa Fe, New Mexico. Association for Computational Linguistics.

Klaus Krippendorff, Reliability in Content Analysis: Some Common Misconceptions and Recommendations, *Human Communication Research*, Volume 30, Issue 3, July 2004, Pages 411–433, *https://doi.org/10.1111/j.1468-2958.2004.tb00738.x*

Chung, H.W., Hou, L., Longpre, S., Zoph, B., Tay, Y., Fedus, W., Li, E., Wang, X., Dehghani, M., Brahma, S., Webson, A., Gu, S.S., Dai, Z., Suzgun, M., Chen, X., Chowdhery, A., Valter, D., Narang, S., Mishra, G., Yu, A.W., Zhao, V., Huang, Y., Dai, A.M., Yu, H., Petrov, S., Chi, E.H., Dean, J., Devlin, J., Roberts, A., Zhou, D., Le, Q.V., & Wei, J. (2022). *Scaling Instruction-Finetuned Language Models*. *ArXiv, abs/2210.11416*.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). *Language models are unsupervised multitask learners.* *OpenAI blog*, *1*(8), 9.

Touvron, H., Martin, L., Stone, K.R., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., Bikel, D.M., Blecher, L., Ferrer, C.C., Chen, M., Cucurull, G., Esiobu, D., Fernandes, J., Fu, J., Fu, W., Fuller, B., Gao, C., Goswami, V., Goyal, N., Hartshorn, A.S., Hosseini, S., Hou, R., Inan, H., Kardas, M., Kerkez, V., Khabsa, M., Kloumann, I.M., Korenev, A.V., Koura, P.S., Lachaux, M., Lavril, T., Lee, J., Liskovich, D., Lu, Y., Mao, Y., Martinet, X., Mihaylov, T., Mishra, P., Molybog, I., Nie, Y., Poulton, A., Reizenstein, J., Rungta, R., Saladi, K., Schelten, A., Silva, R., Smith, E.M., Subramanian, R., Tan, X., Tang, B., Taylor, R., Williams, A., Kuan, J.X., Xu, P., Yan, Z., Zarov, I., Zhang, Y., Fan, A., Kambadur, M., Narang, S., Rodriguez, A., Stojnic, R., Edunov, S., & Scialom, T. (2023). *Llama 2: Open Foundation and Fine-Tuned Chat Models*. *ArXiv, abs/2307.09288*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. *Exploring the limits of transfer learning with a unified text-to-text transformer*. J. Mach. Learn. Res. 21, 1, Article 140 (January 2020), 67 pages.

Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., ... & Chen, W. (2021). *Lora: Low-rank adaptation of large language models.* *arXiv preprint arXiv:2106.09685*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. *Bleu: a Method for Automatic Evaluation of Machine Translation*. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q., & Artzi, Y. (2019, September). *BERTScore: Evaluating Text Generation with BERT.* In *International Conference on Learning Representations*.

Paweł Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gašić. 2018. *MultiWOZ - A Large-Scale Multi-Domain Wizard-of-Oz Dataset for Task-Oriented Dialogue Modelling*. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5016–5026, Brussels, Belgium. Association for Computational Linguistics.

Or Honovich, Leshem Choshen, Roee Aharoni, Ella Neeman, Idan Szpektor, and Omri Abend. 2021. *Q2: Evaluating Factual Consistency in Knowledge-Grounded Dialogues via Question Generation and Question Answering*. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7856–7870, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Tomáš Nekvinda and Ondřej Dušek. 2021. *Shades of BLEU, Flavours of Success: The Case of MultiWOZ*. In *Proceedings of the 1st Workshop on Natural Language Generation, Evaluation, and Metrics (GEM 2021)*, pages 34–46, Online. Association for Computational Linguistics.

Loshchilov, I., & Hutter, F. (2017). *Decoupled Weight Decay Regularization. International Conference on Learning Representations*.