Using Large Language Models to Shape Social Robots' Speech

Javier Sevilla-Salcedo*, Enrique Fernández-Rodicio, Laura Martín-Galván, Álvaro Castro-González, José C. Castillo, Miguel A. Salichs

Universidad Carlos III de Madrid, Madrid (Spain)

Received 14 February 2023 | Accepted 20 July 2023 | Published 27 July 2023

Abstract

Social robots are making their way into our lives in different scenarios in which humans and robots need to communicate. In these scenarios, verbal communication is an essential element of human-robot interaction. However, in most cases, social robots' utterances are based on predefined texts, which can cause users to perceive the robots as repetitive and boring. Achieving natural and friendly communication is important for avoiding this scenario. To this end, we propose to apply state-of- the-art natural language generation models to provide our social robots with more diverse speech. In particular, we have implemented and evaluated two mechanisms: a paraphrasing module that transforms the robot's utterances while keeping their original meaning, and a module to generate speech about a certain topic that adapts the content of this speech to the robot's conversation partner. The results show that these models have great potential when applied to our social robots, but several limitations must be considered. These include the computational cost of the solutions presented, the latency that some of these models can introduce in the interaction, the use of proprietary models, or the lack of a subjective evaluation that complements the results of the tests conducted.

Keywords

Human-Robot Interaction, Large Language Models, Social Robots.

DOI: 10.9781/ijimai.2023.07.008

I. Introduction

Intelligent conversational agents are increasingly being integrated into public-facing tasks such as customer service tasks, including troubleshooting and providing information; assistive chatbots have been used in multiple fields [1]. For interactions with the agent to be smooth, the agent must present itself naturally and intelligently to the user [2]. One of the main barriers is people's reluctance to interact with artificial systems due to unnatural text and, as a result, a conversation experience that is not very fluent, which leads to mistrust and uneasiness [3]. Social robots are embodied conversational agents, and to enable them to advance, we must focus on what makes a conversation natural and fluid, for example, by moving away from predefined and repetitive texts that lack naturalness [4].

One important limitation of current social robots is related to the robot's ability to convey information orally. It is common for robots to use the same expressions repeatedly, which leads to user boredom and, thus, a loss of interest in interacting with the robot. While it is desirable to give robots the ability to generate and adapt the content of their speech, there are situations in which the use of handcrafted texts might have advantages (e.g. when the robot is requesting information from the user). However, this can turn into a limitation if these tailored speeches become repetitive to the user. In these cases, a possible solution could be the development of strategies for phrasing these handcrafted texts differently but maintaining their original meaning.

* Corresponding author.

E-mail addresses: javier.sevilla@uc3m.es

Alternatively, in the field of social robotics, the robot is expected to be able to take the initiative in the interaction, so mechanisms for generating engaging topics and developing them are an interesting addition to this type of robot [5]. These mechanisms can be extended to allow the user to choose the topic to be discussed and to allow the robot to elaborate a discourse around the chosen subject. Furthermore, an interesting adaptation mechanism in human-human interaction is the ability of each speaker to adapt their speech to the other speaker, considering aspects such as age, familiarity, or background. In this sense, it would be desirable to endow the robot with the ability to perform a similar adaptation.

Both of the objectives presented above can be achieved through the use of Natural Language Processing (NLP). This field has attracted significant attention in recent years across multiple disciplines, including robotics. For example, in 2014, Woo et al. [6] combined different NLP techniques to create a conversational system for robotics. Their system uses predefined rules to construct sentences based on the user's input, with a set of fallback sentences for cases in which no sentences is constructed. That same year, Fujita et al. [7] presented an NLP-based model for the Todai robot with the goal of passing the entrance test for the University of Tokyo. The proposed model was trained by using previous exams as inputs. The results obtained were good for mathematics and physics, but the robot failed to pass the history and language portions of the exam. In 2016, Hammed [8] proposed a conversational system for social robots that uses a neural network to build a user profile with knowledge extracted from dialogues with the user and then uses this profile to adapt the conversation. A year later, Williams [9] proposed a text generation framework for robots that relied on a memory model distributed over

two levels. The long-term memory level stores information about locations, objects, and people, while the short-term memory level focuses on the context of the dialogue.

In 2018, Kahuttanaseth et al. [10] presented a system for guiding a mobile robot using natural language. This system uses a Recurrent Neural Network (RNN) encoder-decoder system that filters unnecessary words from the inputs received (e.g. 'please') and then extracts movement command sequences. Although in a limited context, this system can manage multiple commands in a single input. More recently, in 2020, Budiharo et al. [11] tested recurrent and convolutional neural networks as encoders in a question-answering model. The proposed system uses a bidirectional attention flow mechanism [12] to find the similarity between the question asked by the user and the questions in a dataset. Then, it uses this similarity to find the proper answer to the question. Their results showed that the RNN-based encoder performed relatively well. In 2023, Arroni et al. [13] proposed using the Transformer architecture for sentiment analysis. For this, they proposed a network consisting on a single transformer block with 11 attention heads, using learned embeddings of dimension 12 for positional and token encoding. Their results showed that this model outperformed a pre-trained DistilBERT when evaluating them using the Spearman correlation, although not in validation accuracy. That same year, Zhou et al. [14] evaluated the use of ChatGPT for conveying knowledge about gastric cancer (through a medical knowledge test), providing consultation recommendation to patients, and analysing endoscopy reports. Their results show that, while displaying high levels of appropriateness and consistency in its responses, ChatGPT may not always provide accurate responses and suggestions, which indicates that we should not over rely on this model for critical tasks like clinical diagnosis.

Similar to the works presented above, our research also aims to use NLP to allow conversational agents to provide a more natural conversational experience [15]. How the agent expresses itself is crucial for smooth interactions, and the user's perception of the robot can be affected by the responsiveness of its speech. We have used natural language models to tackle these problems, as they provide a new approach to generating non-prewritten texts. On top of that, given that our system has been integrated into a real social robot that interacts with Spanish people, one of the requirements is that it must work in Spanish, so we will consider two methods when exploring language models: on the one hand, we will evaluate multilingual models that include the Spanish language, and on the other hand, we will evaluate models in English, which a priori offer better performance, and use translation tools to generate the desired output.

Another constraint the system must consider is the computing power necessary to run large language models locally and in the cloud. In social robotics, this becomes even more important since these platforms often have limited resources that have to be shared by multiple modules within their software architectures. In particular, the *'two-second rule'* is used to set the maximum delay between interaction turns to two seconds [16], although other authors have reported that users prefer shorter times of around one second [17].

In this work, we seek to mitigate the issues introduced by relying only on predefined texts, which include the reduced fluidity and naturalness of the interactions with users [18], without losing the possibility of using handcrafted texts in situations in which they can be beneficial. In short, when it comes to speech, there are two possibilities: the robot can use predefined sentences or generate new text. We intend to provide solutions to both problems using NLP to obtain a fluent and spontaneous experience during conversations between humans and social robots. Therefore, the first contribution of this article is a methodology that allows the robot to generate pop-up conversation topics and information about them. In addition, the system can dynamically adapt the information to different user profiles. This makes the information more accessible and appealing to users. The idea behind this methodology is that, in social robotics, it is desirable for robots to have the ability to establish non-predefined conversations with users, even without a specific objective. A social robot must be able to interact even when it is not performing a specific task, and these interaction mechanisms must include speech. Therefore, it is important to have mechanisms capable of generating text on the fly on non-predefined topics.

The second contribution of this work is the integration of language models to enrich the verbal capabilities of a social robot, allowing it to paraphrase its repertoire of predefined sentences to achieve greater variability and reduce monotony. We expect that this paraphrasing will improve the quality of the interaction, preventing the robot from becoming monotonous and repetitive.

The rest of this paper is structured as follows. Section II presents the tools and models used in this work. Next, Section VI covers the evaluation setup and metrics employed to assess the proposed methods. Next, Section III presents the first contribution of this work, which deals with the use of NLP models to create semantic descriptions of topics of interest that are adapted to users. Section IV discusses the second main contribution of this work, the paraphrase generator, which will provide the robot with more variability in its speech. Section V describes the integration of both contributions into a real social robot. Section VI presents the evaluation setup, the evaluation of our approach and the models used. In Section VII, the main results of the developed methods are presented. Finally, Sections VIII and IX discuss the main results and draw conclusions.

II. Materials

This section reviews the language models used during the development of this project. Since the inclusion of transformers in the world of deep learning, the development of language models has grown remarkably, along with their capabilities. In this paper, we used these models to give our robots the ability to generate text from scratch and paraphrase the handcrafted texts that they use automatically.

A. Transformers in Language Modelling

A transformer is a deep learning model built on self-attention mechanisms. These mechanisms assess the input data by weighting the significance of each component. While self-attention mechanisms were first included in RNN structures, transformers are built on self-attention alone, and they provide a better performance than RNNs [19]. Transformers have an encoder-decoder architecture in which the encoder layer consists of modules that sequentially handle the input sequence one module at a time. On the other hand, the decoder layer is composed of modules that handle the encoder's outputs. Each encoder layer generates new encoding vectors. These vectors contain information about the components of the inputs that are relevant to each other. Each decoder layer extracts the generated encodings and builds an output sequence out of the decoding and the encoded contextual information [20]. To achieve this, each encoder and decoder module uses the self-attention mechanism [21]. The use of transformers as a new base architecture for language models has brought about a significant change in the capabilities of language models [22]. At the language model level, an increasing number of transformer-based networks have emerged and continue to emerge [23].

B. From Fine-Tuning to Prompt Learning

The way in which models are trained for specific tasks has evolved over time. Traditionally, models have been trained from scratch, starting from random weights and adjusting these weights using a

dataset relevant to the task to be performed; as the size of these models has grown, this has become a long and tedious process. Over the years, to improve the results and reduce training times, researchers have adopted new methodologies, such as transfer learning, which uses the knowledge (weights) of a model trained for a known task as initial weights to train a new model for a new task [24]. Subsequently, fine-tuning was introduced; it allows pre-trained models to be taught new specific tasks. In this context, fine-tuning refers to a technique in deep learning in which the weights of a pre-trained model are readjusted by training without losing their initial settings. The main layers of the model are frozen so that it is better adapted to the tasks for which it is trained [25]. One of the advantages of this is that fine-tuning does not involve training the entire model but rather involves updating its gradient, which is a significantly faster process. Later, more advanced models, such as Generative Pre-trained Transformer 3 (GPT-3) and Text-to-Text Transfer Transformer (T5), went a step further and evolved from fine-tuning to prompt learning [26]. In contrast to fine-tuning, prompt learning or in-context learning refers to a technique in which examples are added within the model's input prompt so that the model can understand the expected inference behaviour without the need to be fine-tuned or specifically trained. Occasionally, prompt learning is referred to as n-shot learning; for example, there can be 'few-shot', 'one-shot', or 'zero-shot' learning. These terms refer to the number of examples of the task given to the model: there can be few, one, or zero examples, respectively.

Fig. 1 shows examples of different types of prompt learning. One of the advantages of n-shot learning is the inclusion of encodings that help the model understand the task it has to perform. For example, in the example shown in Fig. 1, we find the encoding '=>'. This pattern indicates that the model has to look at the sequence before the encoding and then add its result afterwards. In this case, the examples added to the prompt will help the model understand the task of adding integers. When designing a prompt, the number of examples required for correct functioning will depend on the task to be performed, the complexity of the text, and even the sequence format. There is a trade-off between the number of examples and the length of the input sequence, as a longer input sequence means that the model must process more data; hence, more resources must be used, and the inference times will be longer.

```
- Zero-shot -
input_Prompt:
Add two integers:  #task description
2 + 5 =>           #prompt
```

```
- One-shot -
input_Prompt:
Add two integers:  #task description
8 + 4 => 12        #example
2 + 5 =>           #prompt
```

```
- Few-shot -
input_Prompt:
Add two integers:  #task description
8 + 4 => 12        #example
1 + 4 => 5         #example
2 + 9 => 11        #example
3 + 5 => 8         #example
2 + 5 =>           #prompt
```

Fig. 1. Prompt learning examples. Top: Zero-shot learning; Middle: One-shot learning; Bottom: Few-shot learning.

C. Models Used

For the development of the functionalities proposed in this work, several models have been considered, and all of them are listed in Table I. These models are well-known for their performance and are extensively used in different applications. One constraint that has to be kept in mind is the language in which these models have been trained. As stated in the introduction, our robots have been specifically designed to interact with older adults who only speak Spanish. This means that all the predefined texts used by our platforms, which are the texts that we need to paraphrase, are written in this language. For the development of the specific modules for user-adapted semantic description generation presented in Section III, GPT-3 has been used exclusively due to its great adaptability to all the objectives of the application [27] [28]. While GPT-3 has been trained with a corpus of text written in English (which means that we have to translate the results obtained into Spanish), we decided that the level of performance of this model justified the need for this translation step. While the user-adapted semantic description generation pipeline generates text from scratch, the deep learning-based paraphrase generation module does need to work with texts that are prewritten in Spanish, which is a limitation that has to be considered when selecting the model to use for this task. We tested two approaches: (i) using models to paraphrase Spanish sentences directly, and (ii) translating the sentences into English, using models for paraphrasing English sentences, and finally, translating the results back into Spanish. For this task, in addition to testing the performance of GPT-3, we also tested T5, *multilingual T5* (mT5), *Pre-training with Extracted Gap-sentences for Abstractive Summarisation Sequence-to-sequence* (PEGASUS), and BERT2BERT.

GPT-3 is an auto-regressive language model capable of producing human-like text [29] from an input sequence. It was trained with about 45 TB of text data, which led to the refined learning of the language domain and allowed the model to learn new mnemonic rules online. It has been implemented in sentiment analysis [30], used to generate programming code [31], and used for text summarisation [32], among other things. Within GPT-3, there are several versions of the model depending on the size of its architecture, and thus, it has varying capabilities. For our work, we used the Babbage and Davinci models, which are described in Table I.

T5 [33] is a model designed for NLP tasks like translation, summarisation, and question answering, which are all reframed as text-to-text problems. This makes it possible to reuse models, hyperparameters, and loss functions during training for different tasks. This method explores the advantages of scaling the model and the corpus size by using 11 billion parameters during training and the Colossal Clean Crawled Corpus (C4) [33], which includes hundreds of GB of natural-language text. In this work, we tested two checkpoints of the T5 model from HuggingFace [34]. We will refer to them as *PMO-T5*[1] and *Parrot*[2].

A variation of the T5 model used in this work is multilingual T5 (mT5) [35]. It has a similar architecture but has been trained to work in languages other than English using a multilingual version of the C4 dataset. A second difference is that mT5 only uses non-supervised learning, which means that it has to be fine-tuned for any task it will be used for, as shown in Table I. We have fine-tuned an mT5 model[3] with the Spanish instances of the PAWS-X multilingual dataset [36], using the process recommended by HuggingFace[4] (because of this, we will refer to it as HFT5 in the evaluation section).

[1] https://huggingface.co/ceshine/t5-paraphrase-paws-msrp-opinosis

[2] https://huggingface.co/prithivida/parrot_ paraphraser_on_T5

[3] https://huggingface.co/seduerr/mt5-paraphrases-espanol

[4] https://huggingface.co/docs/transformers/training

TABLE I. Table That Summarises the Different Models Used in This Work. The Columns Are, in Order: (I) the Name of Each Model; (Ii) the Number of Parameters That the Model Uses; (Iii) the Architecture of the Model; (Iv) What Task the Models Were First Trained for; (V) If the Model Allows or Not for Multiple Languages; (Vi) How Accessible Is Each of the Models; and (Vii) the Process Required for Adapting the Model to a new Task

Model		Parameters	Architecture	Original Task Training	Multilingual	Access	Task-Specific Training
GPT-3	Babbage	1.3B	Decoder	Next Word Prediction	Yes (Fine-Tuning)	Limited (Paid-Propietary API)	Prompt Learning & Fine-Tuning
	Davinci	175B					
T5		11B	Encoder-Decoder	Masked Language Modeling "span-corruption"	No	Open Source	Prompt Learning & Fine-Tuning
Multilingual T5		300M	Encoder-Decoder	Masked Language Modeling "span-corruption"	Yes	Open Source	Fine-Tuning
PEGASUS		568M	Encoder-Decoder	Gap-Sentence-Generation	No	Open Source	Prompt Learning & Fine-Tuning
BERT2BERT		110M + 110M	Encoder-Decoder	Masked Language Modelling & Next Sentence Prediction	Yes (Fine-Tuning)	Open Source	Fine-Tuning

TABLE II. Inference Parameters Used for Each Pipeline Module. Engine Indicates the Type of Model Used; Temperature Controls the Degree of Randomness of the Response. Response Length Limits the Maximum Number of tokens to Be Generated; Top P Controls to Some Extent the Randomness and Creativity of the Response. Frequency Penalty Penalises the Repetition 1 of Tokens in the Output, While Presence Penalty Penalises the Generation of 2 New tokens Already Present in the Input Text. Finally, Stop Sequences Are Encodings to Stop Generation

Parameter	*Random Topic Generation*	*Semantic Description Generation*	*User-adapted text modification*
Engine	*Babbage*	*Babbage*	*Davinci*
Temperature	0.64	0	0.6
Response length	54	500	200
Top P	1	1	1
Frequency penalty	2	1.92	0.4
Presence penalty	2	0	0.2
Stop Sequences	\n,subject:	\n	“””

The PEGASUS model [37] was designed for summarising texts with different words (abstractive summarisation). It was first pretrained to predict missing sentences in an input text, as this is similar to abstract summarisation. This was done through self-supervised training using a corpus of documents extracted from the web, like the C4 or HugeNews datasets, and then the model was tuned using 12 datasets for abstractive summarisation. Compared with the T5 model, PEGASUS presents similar results with 5% of the parameters. In this work, we have fine-tuned the PEGASUS model for paraphrase generation[5].

The BERT2BERT model has an encoder-decoder architecture in which both components are modelled as *Bidirectional Encoder Representations from Transformers* (BERTs) [38]. The goal is to achieve a larger size with lower resource usage compared to the T5 and PEGASUS models. BERT2BERT's language model has a deeper knowledge of a language's context thanks to bidirectional training. It was trained for *masked language modelling* (predicting missing words in a text) and *next sentence prediction* (predicting the sentence that follows another sentence), as shown in Table I. During training, the weights for the layers present in the BERT model are initialised with the original values from this model, while the layers specific to the BERT2BERT model are initialised to random values. In this work, we used a version of BERT2BERT that uses an encoder and decoder trained in Spanish [39], and we fine-tuned it for paraphrase generation[6].

[5] https://huggingface.co/tuner007/pegasus_paraphrase

[6] https://huggingface.co/mrm8488/bert2bert_shared-spanish-finetuned-paus-x-paraphrasing

III. User-Adapted Semantic Description Generation

This section presents the application of natural language generation techniques to generate user-adapted semantic descriptions. We have proposed a modular design; the pipeline is divided into three main modules, and each one can work independently. Fig. 2 shows a diagram of the developed pipeline, where the random topic generator module generates a topic that is later fed to the semantic description generator module, which creates a description from the given topic. Finally, the user-adapted text modifier module takes the generated description and adapts it according to the type of user with whom the robot is interacting. For conciseness, in the rest of the text, the entire pipeline will be referred to as user-adapted semantic description generation (UASDG). Each module performs an independent inference in the pipeline with individually tuned parameters for optimal performance. Table II lists the parameters used in each module.

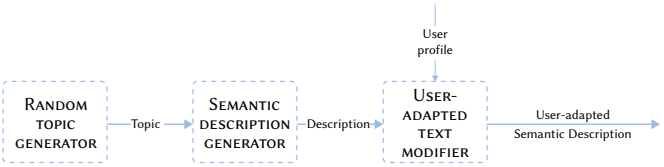

Fig. 2. User-Adapted Semantic Description Generator Diagram.

A. Random Topic Generation

This module can randomly generate a topic, which gives the social

robot a certain spontaneity that enriches human-robot interactions [40]. The idea was to make inferences with the model and obtain a different theme with each execution. As mentioned in Section II, one of the significant advantages of using language models, such as GPT-3, is the use of prompt learning to adjust the model to the needs of a specific task without the need for application-specific fine-tuning training. With this in mind, several input prompts were designed and tested for the model to generate random topics. Fig. 3 (lines 1 to 10) shows how using prompt learning, the model is given several examples with the desired behaviour. The header, together with the examples, helps the model to understand what we are aiming for. Additionally, the inference parameters *Temperature* and *Top P* are adjusted to always favour creativity and not repetitiveness, as shown in Table II. In addition, using the encoding '*subject: -topic-*', the model understands the expected behaviour in a refined way. As seen in the figure, the input prompt shows the desired pattern with eight examples so that the model can understand the pattern to follow. In this case, the topic generated was 'Trees'.

```
1  input_Prompt: The following is a list of random
                 subjects for generating knowledge
                 about the world:
2  subject: Pencils
3  subject: Radiology
4  subject: Fishing
5  subject: Napoleon
6  subject: Dolphins
7  subject: Keys
8  subject: Pigeons
9  subject: Newspapers
10 subject:
11 model Output: Trees
```

Fig. 3. Random Topic Generator example.

B. Theme-Based Semantic Description Generator

This module is designed as a tool to generate a semantic description given an input topic. To this end, as with the topic generation module, numerous input prompts were designed and implemented to test the functionality and exploit its potential using GPT-3. Initially, among other prompts, we tested *'Let me give relevant semantic information about -any topic-'*; however, although the model generated sufficient text, the model generated numerous facts surrounding the theme but failed to provide useful information. Finally, after numerous iterations, the final prompt was *'I have been doing deep research about -any topic-, let me give you brief, relevant semantic information about it'*, which is shown in Fig. 4. This prompt generated a sufficient amount of information in a precise way without too much extra text. In this case, unlike in the previous module, the inference parameters, which are shown in Table II, have been adjusted to favour the text's veracity and thus compromise its creativity.

The adjusted model provides promising results, as it fits the initial purpose correctly, even with zero-shot learning (see Section II). As for the implementation, like the previous module, it is developed as a stand-alone component. Fig. 4 shows the results for the input prompt with the input topic *ladders*.

```
input_Prompt:   I have been doing deep research about
                ladders,  let me give you brief, relevant
                semantic information about it:

model Output:   Ladder is a tool used to climb up or down.
                It can be made of wood, metal, plastic and
                other materials. Ladders are also called as
                steps or rungs in the ladder system. They
                are usually placed on walls and floors for
                easy access to different levels of buildings
                such as apartments, offices etc.
```

Fig. 4. Semantic Description Generator example.

C. User-Adapted Text Modification

Social robots interact with very different and diverse types of people; for example, they can go from talking to a child to talking to an older adult. Therefore, when talking about specific or complex topics, these interactions could be enhanced by a mechanism capable of adapting the text we want to transmit and paraphrasing it according to the user. To this end, after previous experience working with generative models, we have searched for a prompt with a natural language structure that is capable of generating this type of adaptation using zero-shot learning with GPT-3.

Within the input prompt, we briefly describe a person, indicating the type of user (an older person, a child, etc.) so that the model can adapt the paraphrase. We used the encoding (""") within the prompt; this indicates a change from the description to the text to paraphrase. Before the definition, we inform the model that a person has asked us to tell them what the text means. Fig. 5 shows two examples of inference in which we copy the definition and history of gravity from an encyclopaedia[7] and ask the model to adapt it in the first example for an *older person* and in the second example for a *child*. Line *1* of the figure shows the input prompt used for this application. Although only two not-very-descriptive types are shown in the examples, the user information obtained from the robot's perception system can describe the user in a considerable amount of detail to help the model fit the requirements of different users.

As with the applications described above, this application has been implemented modularly as a stand-alone function. In this case, the input prompt has kept the main structure shown in Figure 5; however, making use of the information about the user, we modify the first and last sentences of the prompt, and the text to be modified is introduced into the input prompt following the defined structure, as shown in line 3 of Fig. 5. The output of the module will be the modified text. As shown in the figure, both examples provide adapted descriptions. The output for an older adult explains gravitation as an attractive force among objects using less technical and lighter language. On the other hand, the output for a child simplifies the description even further, leaving behind the 'attraction' concept and focusing instead on the concepts of 'falling' and 'pulling down', which are more relatable to a child.

IV. Deep Learning-Based Paraphrase Generation

While giving social robots the ability to generate texts from scratch is a desirable feature that can improve the naturalness of interactions with users, there are still situations in which having the ability to handcraft the speech of the robot can provide some advantages. However, it is important that these predefined interactions have some variability so that they do not become repetitive. A solution for this is adding methods for paraphrasing these prewritten texts. The proposed method receives the original sentence and returns a

[7] https://www.britannica.com/science/gravity-physics

```
input_Prompt:  A friend, who is really old, asked me
               what this passage means:
"""
Gravity, also called gravitation, in mechanics, the
    universal force of attraction acting between all
    matter. It is by far the weakest known force in
    nature and thus plays no role in determining the
    internal properties of everyday matter. On the
    other hand, through its long reach and universal
    action, it controls the trajectories of bodies in
    the solar system and elsewhere in the universe and
    the structures and evolution of stars, galaxies, and
    the whole cosmos. On Earth all bodies have a weight,
    or downward force of gravity, proportional to their
    mass, which Earth's mass exerts on them. Gravity is
    measured by the acceleration that it gives to freely
    falling objects. At Earth's surface the acceleration
    of gravity is about 9.8 metres (32 feet) per second
    per second. Thus, for every second an object is in
    free fall, Its speed increases by about 9.8 metres per
    second. At the surface of the Moon the acceleration of
    a freely falling body is about 1.6 metres per second
    per second.
The works of Isaac Newton and Albert Einstein dominate
    the development of gravitational theory. Newton's
    classical theory of gravitational force held sway from
    his Principia, published in 1687, until Einstein's
    work in the early 20th century. Newton's theory is
    sufficient even today for all but the most precise
    applications. Einstein's theory of general relativity
    predicts only minute quantitative differences from the
    Newtonian theory except in a few special cases. The
    major significance of Einstein's theory is its radical
    conceptual departure from classical theory and its
    implications for further growth in physical thought.
"""
I rephrased it for him in plain language an elder can
understand:
"""
model Output:  Gravity is a force that exists between
               two objects. It causes the objects to
               have an attractive force on each other.
               The more massive the object is, the
               stronger this attractive force becomes.
```

```
input_Prompt:  A friend, who is a child, asked me what
               this passage means:
"""
- Text about Gravity -
"""
I rephrased it for him in plain language a child can
understand:
"""
model Output:  Gravity is the force that makes things
               fall down. It is the
               weakest force in nature, but it is very
               powerful because it affects
               everything in the universe. Gravity is
               measured by how fast something
               falls. The faster it falls, the more
               gravity is pulling it down. Gravity is
               what makes the Earth's surface curved.
```

Fig. 5. Examples of text modification adapted to the user. The example shown above is adapted for an elderly person, while below is the same text adjusted for a child.

paraphrased version of the input text. The platform into which this method has been integrated has Spanish as its default language. However, because the body of work that focuses on solving NLP tasks in English is larger than that in Spanish, we have decided to test two different approaches to perform this task: (i) we generate the paraphrase directly in Spanish, or (ii) we use a model trained on texts in English, translate the texts used by the robot from Spanish to English before passing them to the model, and translate them back into Spanish after they have been paraphrased. This last approach can be seen in Fig. 6.

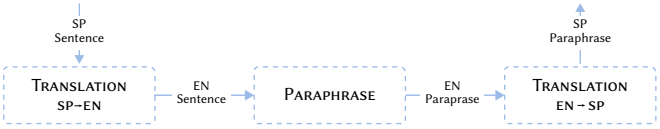

Fig. 6. Diagram of the exploitation process for models that paraphrase sentences, translating them to English before passing the text through the model, and translating the output of the model back to Spanish.

When considering which model to use for our application, one of the key factors was the similarity between the meanings of the original and paraphrased sentences. Our goal was to develop a model that is able to add variability to the robot's speech while maintaining the meaning conveyed by the original utterances. To paraphrase sentences in Spanish, we used the mT5, HFT5, and BERT2BERT models, and fine-tuned them using a Spanish paraphrase dataset, the Paws-x [36]. This task proved to be difficult due to the limited amount of available pretrained models and paraphrase datasets in Spanish. For models that paraphrase sentences in English, we were able to find models that had already been pretrained and fine-tuned on paraphrasing tasks. In particular, we tested PMO-T5, Parrot, PEGASUS, and GPT-3. When these models were used, the original text in Spanish was translated before passing it through the model, and the output was translated back into Spanish.

When we run our paraphrase generator, we can specify which of the tested models will be used to generate paraphrases. If we want to select a model that paraphrases text in English, we can also choose which translator the pipeline will use to convert the texts into English and back into Spanish. The translators that have been tested are the Google Translate[8], DeepL[9], and Argos translators[10].

As stated before, resource usage can be a limitation when working with deep learning models. This is particularly concerning for the task we are trying to perform, as the paraphrase generation module will be involved in the majority of the interactions between the user and the robot, and thus, it must abide by the time constraints that exist in any conversation. While this limitation can be mitigated by deploying these models on specialised hardware, there might be situations in which this is not an option (for example, if the robot is in a location with bad internet). To obtain as much flexibility as possible, our paraphrase module allows both the local and remote execution of the language model. When the module is running externally, the robot sends the utterance that has to be transformed to the server, paraphrase generation is performed there (along with the required translations if needed), and then the server sends the resulting utterance back to the robot.

[8] https://cloud.google.com/translate/

[9] https://www.deepl.com

[10] https://pypi.org/project/argostranslate/

Fig. 7. Mini, a social robot developed for interacting with older adults suffering from mild cognitive impairment.

V. Integrating Our NLP Applications Into the Mini Robot

The models presented in Sections III and IV have been integrated into the social robot Mini [41], which is shown in Fig. 7. Mini is a tabletop robot with a soft appearance that is designed to assist older adults with mild cognitive impairment. This robot has five degrees of freedom (one per shoulder, another on the waist, and two more on the neck and head), OLED screens placed on the face to act as eyes, and coloured LEDs on its cheeks and on its chest. Regarding its perception capabilities, Mini is equipped with touch sensors on the shoulders and belly, a microphone and loudspeaker for speech-based interactions, and a touch screen that can be used both for interacting with users through menus and for displaying multimedia content.

Mini's architecture has been designed following a modular approach, as shown in Fig. 8. At the top of the architecture, a decision-making system (DMS) controls what the robot does at any given time based on stimuli coming from the environment, the inputs given by the user, and the knowledge the robot possesses. Below the DMS, there is a series of modules that allow Mini to perform different tasks: the skills. Examples of these skills include playing cognitive stimulation games, showing the user pictures, videos, music, and other multimedia content, and reading the news to the user, among other things. Here, we find the UASDG pipeline presented in Section III. It has been integrated as an individual skill that can be activated and deactivated by the DMS.

While the DMS and the skills control what task the robot performs at a given moment and how these tasks are performed, a second set of modules in Mini's architecture provides a series of transversal features for any task that Mini needs to complete. The liveliness module generates random behaviours (e.g. motions for all the joints or changes in gaze) to give Mini a lively appearance. The Perception Manager controls the modules capturing information from the environment and the user.

The Human-Robot Interaction (HRI) Manager is the module that controls any interactions between Mini and the robot. Whenever

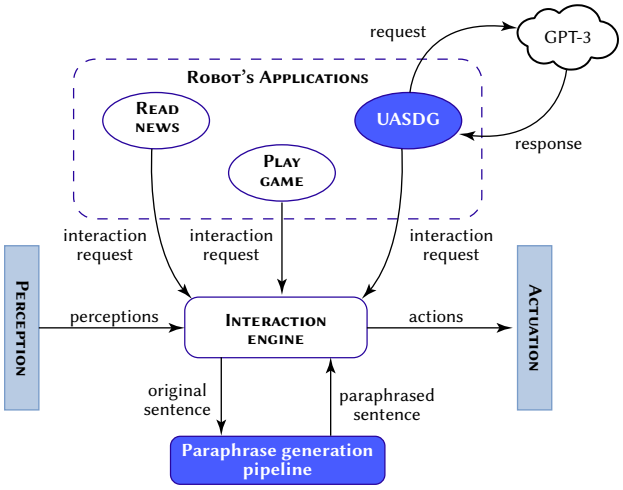

Fig. 8. Schematic view of Mini's architecture. The work presented in this manuscript has been integrated in the blocks in dark blue.

one of the skills needs to start an interaction or respond to a user command, it sends all the information necessary to the HRI Manager, which in turn ensures that the interaction is completed successfully. For example, whenever the UASDG pipeline generates a new text that has to be conveyed to the user, it is sent to the HRI Manager, which in turn ensures that the message is uttered properly and that there are no conflicts with any other interaction requests coming from other skills.

The Expression Manager controls how the robot's messages are conveyed and ensures no conflicts between them. It uses state machine-like structures to model multi-modal expressions. Among the elements in this module, the ones that are relevant for this work are the Interface Players. These Players receive each of the uni-modal actions that make up the expression (e.g. lifting an arm, saying a sentence, etc.) and send commands to the modules controlling the output interfaces (e.g. the drivers for the motors, the text-to-speech module, etc.). The Emotional Text-To-Speech Player is one of the Players and receives sentences that the robot has to utter, prepares them, and sends them to the text-to-speech module. The paraphrasing method described in Section IV has been integrated into this player.

During the startup stage of the software architecture, the ETTS Player loads a YAML configuration file for the paraphrase module. This file, shown in Fig. 9, specifies the model to be loaded and the translator that will be used (if the model parameters contain the name of one of the models trained to paraphrase sentences in English), a parameter that indicates whether the robot's utterances have to be paraphrased or not (this allows us to bypass the paraphrase module if we do not need it), and the deployment mode for the paraphrase module.

This is the case, for example, for the paraphrase generation module, as it is integrated into the Expression Manager, a key element in the interactions between the user and the robot. On the other hand, if the architecture presented in this section is deployed on a platform with enough resources, we might prefer to run the models locally to avoid potential problems caused by communication with external machines. Because of this, the tools presented in this work can be deployed in one of three manners. GPT-3 models are only accessible through the API provided by OpenAI, and thus they run the inferences on their servers. For those models that are accessible to researchers, we provide two possible solutions: running the models directly on Mini or deploying them on our external server. This server has been specifically designed to run machine learning models, which are too computationally demanding for Mini's hardware. It has an Intel Core i9-10900K CPU that runs at 3.7 GHz, an NVIDIA GeForce RTX 3090 GPU, and 64 GB of RAM.

```
paraphrase_config:{
    'model': 't5',
    'translator': 'deepl',
    'mode': 'local',
    'paraphrase': 'on',
    'pauses': 'on'
}
```

Fig. 9. Example of the YAML file used for configuring the paraphrase module.

VI. Evaluation Methods

This section outlines the evaluation setup for assessing the quality and efectiveness of paraphrased sentences, the evaluation of the user-adapted semantic description generation approach and the models used for paraphrase generation in Spanish and English. First, we describe the metrics that will allow us to compare the models used for user-adapted text modification and paraphrase generation. In this last case, separate comparisons have been performed for models trained in Spanish and models trained in English. Next, we describe the evaluations that have been conducted to test the two contributions presented in this manuscript.

A. Metrics

There are three main factors that we need to keep in mind when evaluating the quality of a paraphrased sentence: (i) the inference time should be as low as possible so that it does not hinder the interaction with the user; (ii) the meanings of the original and paraphrased sentences should be as close as possible; and (iii) the original and paraphrased sentences should be as different as possible. The inference time is a critical factor when using AI models in human-robot interactions, as studies have shown that responses in a conversation can lose their meaning if they are delivered too late. Times over two seconds make it impractical to achieve an optimal interaction. Moreover, to measure the similarity between two sentences' meanings and how they are written, we use two metrics widely used in NLP: the BiLingual Evaluation Understudy (BLEU) and BERT scores.

1. BLEU

The *BiLingual Evaluation Understudy* score [42] is used to evaluate the quality of an automatic translation; that is, it indicates the similarity between the translation generated by the model and a translation made by a human being. The main advantage of this metric is that it is easy to calculate and interpret, is language-independent, and tends to match human evaluations. Since its inception, it has spread from automatic translation to other NLP tasks, such as paraphrase generation. Using BLEU in our evaluation, we will be able to assess if the paraphrasing process generates sentences that are different enough to add real variability to the robot's speech.

BLEU compares matching words in both sentences, known as *n-grams*, where *n* indicates the number of words compared simultaneously. This metric also penalises the candidate sentence based on the lengths of the original and candidate sentences. Once the metrics for the individual n-grams have been computed, we can calculate the cumulative BLEU score. This value can go from zero, i.e. two sentences are completely different, to one , i.e. both sentences are identical. We will attempt to obtain the lowest BLEU score possible because we want to obtain a sentence distinct from the original. In our research, we used the BLEU-2 and BLEU-3 metrics. The former computes the geometrical average of the 1-gram and 2-gram precisions, while the latter computes the geometrical average of the 1-gram, 2-gram, and 3-gram precisions. These metrics have been initialised with the weights shown in Table III.

TABLE III. Weights Used for the BLEU-2 Y BLEU-3 Metrics

	1-gram	*2-grams*	*3-grams*
BLEU-2	0.25	0.25	0
BLEU-3	0.33	0.33	0.33

2. BERT

The BERT score [43] evaluates the semantic similarity between sentences. To do this, contextual embeddings are generated using BERT to represent the tokens in both the original and candidate sentences. Tokens are then compared using the cosine similarity. In the BERT score computation, *precision* and *recall* are calculated based on this comparison. Precision is determined by the proportion of tokens in the candidate sentence with a high cosine similarity with any token in the original sentence. It measures the relevancy of the generated tokens to the original sentence. On the other hand, recall represents the proportion of tokens in the original sentence with a high cosine similarity with any token in the candidate sentence. It measures the coverage of the generated tokens compared to the original sentence.

Using both values, F1-score is computed. The F1-score provides an overall measure of the similarity between the meanings of the original and generated sentences. Its value is between 0 and 1, with 1 indicating the highest possible similarity. Thanks to this metric, we will be able to ensure that the paraphrased sentences maintain the meaning of the original utterance while adding variability to the robot's speech, which could hinder the interaction.

B. Evaluation of the User-adapted Semantic Description Generation Approach

In this evaluation, we measured the response times of the three modules of the user-adapted semantic description generation pipeline: the random topic generator, the semantic description generator, and the user-adapted text modification module. The evaluation process involved running the system 200 times. Each run started with a random topic generated by the topic generator. The semantic description generator then produced an arbitrary description related to the topic. This description was then adapted to the user using the text modification module. The response times of the pipeline were analyzed throughout the iterations.

C. Evaluation of the Models Used For Paraphrase Generation

This evaluation compared different models for their effectiveness in paraphrasing sentences. The evaluation was conducted separately for models trained in Spanish and English. A set of 539 sentences was used for the evaluation. These can range from having one word to 80. Table VIII in Appendix A shows an example of sentences extracted from the set used for evaluating our solution. When evaluating the models trained in English, the sentences have been translated first from Spanish and then back to this language after being paraphrased.

We tested the mT5 and BERT2BERT models to paraphrase sentences directly in Spanish. We fine-tuned the mT5 model ourselves (the HFT5 model). During the evaluation, we passed every sentence through both models and compared the paraphrased sentences generated by the models with the original sentences using the BLEU and BERT scores. We also measured the time required to obtain the paraphrased sentences. For this, the paraphrase pipeline returns, alongside the paraphrasing result, the timestamp at four points in the process: (i) when the paraphrase request is received; (ii) when the translation from Spanish to English is completed; (iii) when the model has returned the paraphrasing result; and (iv) after the paraphrased sentences have been translated back into Spanish. For this first test, there was no translation, so we only used the timestamps for points (ii) and (iii). In this evaluation, the models were deployed locally. Once

all the sentences were paraphrased, we calculated the average values of the metrics and the response time.

Regarding the models trained to perform paraphrasing in English, we tested the PMO-T5, Parrot, PEGASUS, and GPT-3 models. We evaluated these models using the same sentences used to evaluate the models trained in Spanish. This means that, in this case, the sentences had to be translated from Spanish to English, and the paraphrase results had to be translated back into Spanish. For this evaluation, we used the DeepL translator web service. Because of this, we present two separate sets of measurements: (i) the BERT and BLEU scores for the original sentences after translating them into English and the sentences generated by the models before translating them back into Spanish (which demonstrates the performance of the models themselves, without the translation from Spanish to English and from English to Spanish); and (ii) the BERT and BLEU scores for all four models calculated by comparing the original sentence in Spanish and the generated sentence after translating it back into Spanish (the performance of the entire pipeline).

VII. Results

In this section, we discuss the results obtained for the evaluation of the User-adapted Semantic Description Generation approach and the Paraphrase generation models. These results include both the ones obtained using the metrics described in Section VI, as well as the inference time for each of the two contributions presented.

A. Results of the User-adapted Semantic Description Generation Approach

This section will cover the quantitative results of user-adapted semantic description generation. By analysing the response times of our pipeline, we evaluated the system iteratively to validate its performance. The results are shown in Table IV. Appendix B shows various user-adapted semantic description generation examples from the set used to evaluate our pipeline. We can see that the median response time of the entire pipeline is 4.87 seconds. Within the pipeline, the topic generation module is relatively fast, with a median time of 0.23 seconds, followed by the description generation module. Finally, the text adaptation module is the slowest, with a median time of 3.29 seconds.

TABLE IV. Inference Time Response Statistic Analysis for the User-adapted Semantic Description Generation Pipeline

	Random Topic Generation	Semantic Description Generation	User-adapted text modification	Complete Pipeline
Min (s)	0.20	0.36	0.82	1.46
Max (s)	12.64	13.16	8.42	16.89
Median (s)	**0.23**	**1.02**	**3.29**	**4.87**

Regarding the user-adapted text modification module, we used the metrics described in subsection VI.A to evaluate the ability of the model to maintain the original content of the text; however, it should be noted that our aim in developing the user adaptation module is not to remain faithful to the text itself but rather to ensure that the end user understands the text. As in evaluating response times, we performed 200 iterations with the module configured to interact with an older adult using the prompts shown in Section III to analyse its performance. The average value of the BERT score is 0.76; this value suggests that we are not losing the main ideas and intentions of the original texts. On the other hand, for the BLEU scores, we obtain averages of 0.33 for BLEU-2 and 0.28 for BLEU-3; these are low scores overall, which may mean that the text has different sentence structures and words in more complete adaptations.

Several videos in which the application is used have been recorded to demonstrate its use. To make the part of the text being adapted to the user easier to perceive, the topic that was chosen is wine; in the first video, the generated text is shown without adaptation[11]. The second video and the third video show, respectively, the adapted text for an elderly person[12] and a child[13].

B. Results of the Models Used For Paraphrase Generation

As covered in section VI, we have compared the different models in Spanish and English. Regarding the Spanish models, the results, shown in Table V show that the mT5 and HFT5 models obtained similar BLEU-2/3 and BERT values (0.76/0.74 and 0.77, respectively), while the difference between these values is higher for the BERT2BERT model (0.62 and 0.50/0.43). Regarding the inference time, mT5 and BERT2BERT were able to generate new sentences in under two seconds (0.88 s for mT5 and 1.42 s for BERT2BERT). For HFT5, the time required to obtain a prediction averaged 4.04 seconds, which is significantly slower than the inference time of the pretrained mT5 model, although their BERT and BLEU-2/3 scores were similar.

TABLE V. Evaluation Results for the Models Fine-Tuned for Paraphrasing Sentences in Spanish

	BERT	BLEU-2	BLEU-3	local t (s)
mT5	0.77	0.76	0.74	0.88
HFT5	0.79	0.78	0.76	4.04
BERT2BERT	0.62	0.50	0.43	1.42

When we manually evaluated the paraphrase generation results in Spanish, we observed that the sentences generated by the mT5 model were either identical to the original sentences or lost their original meaning. The latter problem was also observed in the sentences generated by the BERT2BERT model. We also observed that the mT5 model truncated sentences greater than a certain length. These issues also appeared in the sentences generated with t he HFT5 model. Finally, during this manual review, we observed that there were cases in which the paraphrased sentences present objectively good results, as they maintain the meaning of the original sentence while changing how it is written, but they might not make complete sense or might be phrased in a way that will sound weird to users.

As far of the models trained to perform paraphrasing in English, the results, shown in Table VI, give an idea of how good the selected paraphrasing models are and how good the proposed translator-paraphrase-translator architecture is for our application (paraphrasing sentences in Spanish).

When we analyse just the paraphrase step, the results obtained are very similar for the Parrot and PEGASUS models, which have BERT scores that are higher than their BLEU-2/3 scores (0.69 and 0.44/0.38 for the former, 0.66 and 0.32/0.37 for the latter). On the other hand, we saw an increase in all the metrics for the PMO-T5 model, which has high BERT and BLEU-2/3 scores (0.88 and 0.76/0.7, respectively). Finally, the GPT-3 model scores (a BERT score of 0.74 and a BLEU-2/3 score of 0.55/0.48) were between those obtained for the PMO-T5 model and those obtained for the Parrot and PEGASUS models. When we add the translation steps before and after paraphrase generation, we can see a similar increase in all metrics. Finally, we compare the times required

[11] https://youtube.com/shorts/E7azQgY4HD8?feature=share

[12] https://youtube.com/shorts/mRUOn1MBzuQ

[13] https://youtube.com/shorts/Vy3_n-VBITM

TABLE VII. Examples of Failed Paraphrases When Using Models Trained in Spanish

	Spanish sentence	English translation
Original	¿Quieres continuar con el juego?	Do you want to continue with the game?
Paraphrased	En el juego, ¿Quieres continuar con el juego?	In the game, do you want to continue with the game?
Original	Me llamo Mini y soy un robot social.	My name is Mini, and I am a social robot.
Paraphrased	Me llamo Mini y soy un robot social.	My name is Mini, and I am a social robot.
Original	¿Cómo te llamas?	What is your name?
Paraphrased	¿Cómo se llama la llama?	What is the flame's name?
Original	Me llamo Mini y soy un robot social.	My name is Mini, and I am a social robot.
Paraphrased	Me llamaron Mini y somos una mente social.	They called me Mini, and we are a social mind.

to obtain a paraphrased sentence. We see that the PMO-T5 and GPT-3 models show the best results, both when they are run locally (2.04 s for PMO-T5, 2.01 s for GPT-3) and when they are run on the external server (1.61 s for PMO-T5, 1.59 s for GPT-3), compared with the Parrot (3.16 s when run locally, 2.27 s when run on the server) and PEGASUS (3.46 s when run locally, 2.05 s when run on the server) models. If we evaluate the models individually (without taking the translation steps into account), we see that the PMO-T5 model ran faster than GPT-3 (0.85 s/0.61 s for PMO-T5 and 1.06 s/0.74 s for GPT-3 when they are run locally / on the server), while Parrot proved to be the slowest (2.76 s when run locally and 1.2 s when run on the server).

TABLE VI. BERT and BLEU Scores, and the Inference Time for Local and Remote Execution, for the Models Trained in English When Evaluating Only the Paraphrase (Paraph), and When Evaluating the Entire Pipeline (Trans-Paraph-Trans). The Highest BERT Score and the Lowest BLEU Score and Inference Times Have Been Highlighted in Bold for Evaluations That only Consider the Paraphrase and for Evaluations That Consider the Entire Pipeline

	BERT	BLEU-2	BLEU-3	local t (s)	remote t (s)
PMO-T5 paraph	0.88	0.76	0.7	0.85	0.61
PMO-T5 trans-paraph-trans	0.77	0.56	0.46	2.04	1.61
Parrot paraph	0.69	0.44	0.38	2.76	1.2
Parrot trans-paraph-trans	0.5	0.31	0.25	3.16	2.27
PEGASUS paraph	0.66	0.44	0.37	2.57	1.12
PEGASUS trans-paraph-trans	0.51	0.32	0.25	3.46	2.05
GPT-3 paraph	0.74	0.55	0.48	1.06	0.74
GPT-3 trans-paraph-trans	0.56	0.39	0.31	2.01	1.59

Finally, as a proof of concept for the paraphrase module, we used one of Mini's applications: telling stories to the user. We chose one of the stories and recorded one video in which the robot tells the story as is[14] and another in which the robot paraphrases the story before telling it[15]. This is done by passing the sentences in the story through the paraphrase pipeline one by one.

[14] https://youtube.com/shorts/rERpBROzhtw?feature=share

[15] https://youtube.com/shorts/WGWZ4NN6fz8?feature=share

VIII. Discussion

The results of the Spanish paraphrasing models indicated that mT5 had high BLEU and BERT scores, meaning similar meanings to the originals but with limited wording variation. In contrast, BERT2BERT produced different sentences that lost some original meaning. In the case of the English paraphrasing models, PMO-T5 had the highest scores for both metrics, sacrificing some original meaning for more diverse sentences. There was a trade-off between semantic and text similarity, and the Spanish paraphrased sentences were generally of lower quality than English, as shown in Table VII. Therefore, we decided to focus on those modules finetuned for paraphrasing sentences in English.

If we focus on the adaptation that the UASDG pipeline performs, on top of the text and semantic similarities, there is a third factor that also plays a role: how well the paraphrased text takes into account the profile of the user interacting with Mini (if the user is a child or an older adult). The analysis of the results obtained by computing the BERT and BLEU-2/3 scores indicates that the method maintains semantic relevance in the context of both texts. In contrast, the modified text does not bear much resemblance to the original, which reflects the effort made by the model to adapt the text so that it can be better understood by the user. A possible reason for these results, compared to those observed for the paraphrase module, is that the text has to be adapted to different audiences, and this introduces a certain level of variability. User-adapted text modification tends to highlight the main concepts found in the original text, leading to the omission of things that may be too complex for the user or not important for understanding the original topic, as shown in Table IX, Appendix B. There must be a trade-off between omitting complex elements and not undermining the understanding of the text.

Conversely, focusing on the entire UASDG pipeline, the ability to adapt to the user's profile and the possibility of autonomously selecting conversational topics may enhance the perceived intelligence and naturalness of the robot, thus improving its interactions. Nevertheless, there are still some challenges to overcome in full integration. In general, integrating generative language models in this scope gives us flexibility and provides creativity to some extent without losing naturalness. The correct design of the prompt in each module allowed us to correctly match the expected performance in the preliminary results. Although the initial objectives have been met, despite the fact that we made several templates adjusted to the profiles shown in the examples, a dynamic adaptation should also be created to not restrict the possible user profiles [24]. Our pipeline can generate semantically rich text efficiently. Topic generation has proven to be a useful tool providing the system with spontaneity and creativity. On the other hand, the text adapted to the user shows high grammatical malleability without losing the semantics of the original text, which helps the message reach the user in optimal shape.

Even though the quality of the text generated by the models we use in our applications is a key aspect that can be used to assess the usability of the modules presented in this manuscript, another factor must be taken into account. These models will be integrated into a robot designed for human-robot interaction. As we mentioned in Section VII, messages conveyed by a participant in a conversation can lose their meaning if they are delivered with an extreme delay. Some studies set the maximum delay between interaction turns to 2 seconds [16], although other works contend that this time should be lower (around 1 second) [17]. Because of this, it is important to consider the inference time when deciding whether a model can be integrated into our architecture. The HFT5 model was unfit for real interactions using the paraphrase module, as its inference time is above two seconds. While the BERT2BERT model does meet the two-second threshold, its inference time is still too close to this threshold, meaning that the rest of the robot's modules involved in conveying responses to the user would have to perform their tasks in under 0.6 seconds for the total response time to be under 2 seconds. Finally, the pretrained version of mT5 is the only model trained in Spanish that could perform at the speed required in real interactions. For the models trained in English, only the PMO-T5 and GPT-3 models can perform below the selected threshold, and they can do this only when they are deployed on the external server. However, in both cases, the mT5 model in Spanish has the same issue (the total time is too close to the limit). Here, it is important to mention two things. First, the measured times do not consider the delay introduced by the communication between the robot and the server. Second, the inference time was obtained by averaging the time required to paraphrase the entire list of sentences used by Mini. However, some of the sentences used were significantly longer than the rest, increasing this average value. Most of the sentences used by Mini in common situations are shorter, and thus the time required to paraphrase them will be lower.

When it comes to UASDG response times, on the other hand, we found relatively longer overall times for the entire pipeline. The topic generation module is faster than the rest of the modules because it generates a single term and has a short prompt, which means that the model can work with a smaller amount of text. On the other hand, the slowest module in the pipeline is the module that has to handle the largest amount of text, which is the user-adapted modification module; however, there is a key difference between the UASDG and paraphrase modules. The UASDG functionality will be part of a robot's skills, which means it will not be part of every interaction between Mini and the user. Additionally, this module is not used to respond to the user's inputs, which softens the time requirements. Additionally, in the case of excessively long waiting times during the execution of the pipelines, we have deterrent techniques with utterances for the robot to use to fill these gaps without affecting the interaction. For these reasons, making UASDG follow the two-second rule is not as critical as ensuring that the paraphrase module follows this rule.

Finally, while the results observed are encouraging, a series of limitations must be addressed. One of the main limitations, which is due to the large sizes of these models, is the computational capacity required for training and inference and its related costs. In the case of the text generation module in the UASDG pipeline, because it is a large decoder-based model, its use leads to a higher latency in the inference that, when implemented with social robots, can affect its immediacy and thus the naturalness and fluidity of the interaction. We were able to mitigate this limitation by training and deploying our models on an external server, but these tasks can still be challenging. A second limitation connected to the selected models is that some of them (like GPT-3) are proprietary models, which limits the level of access that we have to them. Regarding the evaluation of the proposed modules, we decided to focus on objective evaluations, as they can help us determine if a particular model can or cannot be integrated into our architecture, and they give us a good idea of how these models are going to perform. However, sometimes the perception that the user has of a robot does not coincide with the results provided by objective metrics. For example, while the BERT score might indicate that Mini's dialogues are losing part of their meaning after going through the paraphrase module, this may not be an issue for the user, and the interaction might still be satisfactory. Thus, conducting a subjective evaluation of the modules presented in this manuscript would be useful. Also, another limitation of the evaluation of the paraphrase generation module is that the sentences generated by the models fine-tuned for paraphrasing sentences in English were evaluated by Spanish native speakers, which could have affected their perception of the appropriateness of these sentences. Finally, some technical limitations related to the paraphrase module must be mentioned. The first one is connected to the format that the paraphrase module expects the input sentence to have. Text-to-speech modules used in robotics can provide special commands for modifying how a sentence is uttered (for example, introducing pauses into the speech or altering the prosodic features of the voice) or for introducing non-verbal sounds (like a laugh or a yawn). However, the proposed method for paraphrasing sentences does not allow these commands. Thus, if this module has to be used with a TTS module that allows these commands, it would be necessary to remove the commands before sending the sentence to the paraphrase module and then put them back once the output sentence is received. Finally, one last limitation that has to be considered is that paraphrasing sentences has a chance of resulting in text that makes no sense, which could hinder interactions (although our results show that this is not common). Regardless of these limitations, the results obtained by evaluating the integration of the NLP applications presented in this manuscript into Mini's architecture indicate that our work was completed successfully.

IX. Conclusions

In this work, we have presented how language models can enhance human-robot interactions. In particular, we have addressed two problems. First, we implemented a mechanism that allows robots to talk about topics that have not been considered beforehand. To this end, we used the GPT-3 model to generate an appropriate topic of conversation and then to obtain relevant information about this topic. Moreover, the received information needs to be adapted to the person the robot is talking to. Thus, the robot adapts the conversation to the profile of the user. With this mechanism, when the robot is, for example, interacting with a child, it uses language that is not technical so that the child can understand it.

Second, when interacting with robots that use predefined utterances, the user might perceive the robots as repetitive and monotonous. To mitigate this issue, we have integrated different language models for paraphrasing predefined texts written in Spanish. The results have shown a trade-off between the variety we can introduce in the text and the amount of meaning that is lost in the process. Additionally, when English-based models are used, English-Spanish translations produce significantly more variability than the direct use of Spanish-based models.

Both mechanisms have been integrated into our social robot, Mini, considering the fact that additional interaction delays might reduce the interaction quality. While the results obtained are encouraging, there are still some limitations that should be tackled in future work. These limitations include the computational power required to run some of the larger language models, the latency that these modules introduce in interactions, and the lack of control over proprietary language models (like GPT-3). Regardless, the results point towards the advantages that integrating transformer-based NLP solutions can provide for the interaction capabilities of social robots.

Appendix

A. Examples of Sentences Used for the Evaluation of the Paraphrase Pipeline

TABLE VIII. Examples of Paraphrases When Using Models Trained in Spanish, With Their Translation to English

Spanish original sentence	English translation
¿Cúanto es 9 menos 4?	What is the result of 9 minus 4?
¿En qué ciudad se encuentra esta torre?	In which city is this tower located?
¡Empezamos!	Let's start!
Muéstrame una tarjeta con un objeto verde.	Show me a card with a green object.
Vamos con una fácil para empezar.	Let's start with an easy one.
Claro, a mi las noticias a veces me aburren.	Sure, I sometimes get bored of the news.
¿Quieres elegir otro cartón?	Do you want to choose another card?
Podemos repetirlo en otro momento.	We can repeat this another time.
Los árboles eran el pino, el abeto, el roble y el sauce.	The trees were the pine, the fir, the oak, and the willow.
¡Muy bien!	Very good!
Para saber la respuesta dividimos 12 entre 3. El resultado es 4 lápices por persona.	In order to find the answer, we divide 12 by 3. The result is 4 pencils per person.
Se trata de la catedral de Zamora. Pero qué bonita es esta ciudad.	It is the cathedral in Zamora. Oh, how beautiful this city is.
¿Cuáles de estas palabras son deportes?	Which of these words are sports?
Esta palabra es algo que se encuentra en un baño.	This word is something that you can find in a bathroom.
¿Cuánto da si resto 8 a 20?	What is the result of subtracting 8 to 20?
¿Estás seguro de que no tienes línea?, mira bien tu cartón. Seguimos para bingo.	Are you sure that you don't have a line? Check your card again.
Por favor, muéstrame un hexágono.	Please, show me an hexagon.
Juan va al mercado. La carne le cuesta 12 euros, y paga con un billete de 20. ¿Cuánto le tienen que devolver?	Juan goes to the market. Meat costs 12 euros, and he pays with a 20 euro bill. How much change is he getting?
Acuérdate de contestar usando el micrófono.	Remember to answer using the microphone.
¿De qué animal se trata?	Which animal is this?
¿Qué palabra de la pantalla está relacionada con agua?	Which word in the screen is related to water?
La respuesta era casa, mochila, alfombra y pelota.	The answer was house, backpack, carpet, and ball.
En este juego, te voy a ir enseñando objetos, y luego tú, tienes que pulsar en la persona de la pantalla que utiliza ese objeto para su profesión.	In this game, I will show different objects to you, and you have to select among the people in the screen the one that uses that object in their work.
La solución es plátano.	The answer is bannana.
Este ejercicio es para ver cómo de bien conoces la ciudad de Zamora. Yo te voy a ir enseñando edificios conocidos de la ciudad y tú me tienes que decir cómo se llaman.	This exercise aims at evaluating how well do you know the city of Zamora. I will show you known buildings in the city, and you have to tell me their name.
La solución era mochila.	The answer was backpack.
En la imagen había 1 euro y 14 céntimos.	The image showed 1 euro and 14 cents.
En este ejercicio, voy a ir mostrando colores por el corazón y tú tienes que seleccionar en la pantalla, el objeto que sea del mismo color.	In this exercise, I will light my heart in different colours, and you have to select on the screen the object that is the same colour.
¿Qué edificio es el que muestro ahora?	Which building am I showing now?
En este juego, voy a ir poniendo diferentes fotos de comida y me tienes que decir a qué zona de España pertenece. Para que no sea my difícil, te voy a dar a elegir entre tres opciones. A ver qué tal se te da.	In this game, I will show you different pictures of food, and you have to tell the region in Spain that food belongs to. Just so it is not that hard, I will give you three options to choose from. Let's see how you do.
¿Cuáles son las palabras que te he dicho? Dímelas por el micrófono de una en una.	What are the words that I just said? Say them through the microphone one by one.
¡Bravo! Cómo me gusta esta canción. Espero que a ti también. Bueno, vamos a seguir con más ejercicios.	Bravo! Oh, how I love this song. I hope you like it too. Well, let's continue with more exercises.
Contesta cuando escuches el pitido.	Answer after you hear the beep
Parece que no nos estamos entendiendo, debe de haber algún problema.	It seems that we are not understanding each other, there must be a problem.

B. Examples of Generations Used for the Evaluation of the UASDG Pipeline

TABLE IX. Examples of Semantics Description Generations Used for Evaluation. It Is Worth Noting How User Adaptation Is Able to Synthesise Long and Detailed Texts to Improve the End-user's Understanding

Topic	Semantic Description	Elder Adaptation	Child Adaptation
Glasses	Glasses are a type of optical device that is worn on the eyes to correct vision. They can be used for reading, watching television, or computer use. Glasses are also called spectacles and they come in many different types and styles. The most common types of glasses include bifocals (which have two lenses), trifocals (which have three lenses) and progressives (which allow you to see objects at progressively closer distances)	Glasses are something you wear on your eyes to help you see better. They come in many different kinds and styles, and can be used for different things like reading, watching TV, or using a computer. The most common types are called bifocals (with two lenses), trifocals (with three lenses), and progressives (which help you see things up close).	Glasses are like special helpers for your eyes. They help you see things better, like when you read a book or look at a computer. There are many different kinds of glasses, but the most common ones have two or three lenses. These help you see things up close or from far away.
Pizza	Pizza is a flat bread that originated in Italy. It is usually topped with tomato sauce, cheese and various meats such as ham, sausage or bacon. The word pizza comes from the Italian word pizze which means "pie" (as in pie crust)	Pizza is a type of food that started in Italy. It's like a flat round cake with tomato sauce, cheese, and different meats like ham, sausage, or bacon on top. The word "pizza" comes from the Italian word "pizze" which means a type of pastry.	Pizza is a type of food that started in Italy. It's a flat bread with yummy things on top like sauce, cheese, and meats like ham or sausage. The name "pizza" comes from a word in Italian that means "pie".
The Great Pyramid	The Great Pyramid of Giza is the oldest and largest of the three pyramids in the Giza Necropolis bordering what is now El-Gizah, Egypt. It was built as a tomb for Fourth Dynasty Egyptian pharaoh Khufu (Cheops) around 2560 BC or earlier during the Old Kingdom period. The pyramid, originally named Khufu's Pyramid but known to its builders as ¨the Great Pyramid¨, has an area of about 230,000 m^2 (23 acres), though it is not quite square. It is made from 2.5 million blocks ranging in size from 1 to 10 tons each. Originally covered with casing stones that formed a smooth outer surface, some ancient sources claim that this covering was removed at some point in antiquity; however it remains one of only two large stone structures ever built by humans without any metal tools, It stands 648 feet tall and contains over 100 chambers on its four sides: King's Chamber – where pharaohs were buried – Queen's Chamber – where queens were buried, Grand Gallery - which connects King's chamber with Queen's chamber via an underground passage called "Queen's tunnel", and Subterranean Chambers - which are located under all other chambers except for Kings' chamber	The Great Pyramid of Giza is a very old building in Egypt. It was made a long, long time ago, around 2560 BC. It was built for a king named Khufu and was used as a place to keep his body after he died. The pyramid is very big and was made with big stones that weigh a lot. It's 648 feet tall and has many rooms inside, including a room for the king and one for the queen. There's also a big hallway and some rooms under the ground.	The Great Pyramid of Giza is a really old and big building in Egypt. It was made a long, long time ago to be a tomb for a king named Khufu. It's made of lots and lots of big blocks of stone, and it's so big that it's like an entire town! It's also very tall, almost as tall as a big building with 20 floors! Inside the pyramid, there are many rooms and chambers where the king and queen were buried. There's even a secret underground passage that connects the rooms!

Acknowledgement

The research leading to these results has received funding from the projects: Robots sociales para mitigar la soledad y el aislamiento en mayores (SOROLI), PID2021-123941OA-I00, funded by Agencia Estatal de Investigación (AEI), Spanish Ministerio de Ciencia e Innovación. Robots sociales para reducir la brecha digital de las personas mayores (SoRoGap), TED2021-132079B-I00, funded by Agencia Estatal de Investigación (AEI), Spanish Ministerio de Ciencia e Innovación. This publication is part of the R&D&I project PLEC2021-007819 funded by MCIN/AEI/10.13039/501100011033 and by the European Union NextGenerationEU/PRTR. This work has been supported by the Madrid Government (Comunidad de Madrid-Spain) under the Multiannual Agreement with UC3M ("Fostering Young Doctors Research", SMM4HRI-CM-UC3M), and in the context of the V PRICIT (Research and Technological Innovation Regional Programme).

References

[1] L. Laranjo, A. G. Dunn, H. L. Tong, A. B. Kocaballi, J. Chen, R. Bashir, D. Surian, B. Gallego, F. Magrabi, A. Y. S. Lau, E. Coiera, "Conversational agents in healthcare: a systematic review," *Journal of the American Medical Informatics Association*, vol. 25, pp. 1248–1258, Sept. 2018.

[2] J. Cassell, J. Sullivan, E. Churchill, S. Prevost, *Embodied Conversational Agents*. MIT Press, 2000.

[3] L. Clark, N. Pantidi, O. Cooney, P. Doyle, D. Garaialde, J. Edwards, B. Spillane, E. Gilmartin, C. Murad, C. Munteanu, V. Wade, B. R. Cowan, "What Makes a Good Conversation? Challenges in Designing Truly Conversational Agents," in *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI '19, May 2019, pp. 1–12.

[4] V. Klingspor, Y. Demiris, M. Kaiser, "Human- robot-communication and machine learning," *Applied Artificial Intelligence*, vol. 11, 03 1999.

[5] C. Clavel, Z. Callejas, "Sentiment Analysis: From Opinion Mining to Human-Agent Interaction," *IEEE Transactions on Affective Computing*, vol. 7, pp. 74–93, Jan. 2016.

[6] J. Woo, J. Botzheim, N. Kubota, "Conversation system for natural communication with robot partner," in *2014 10th France-Japan/ 8th Europe-Asia Congress on Mecatronics (MECATRONICS2014- Tokyo)*, Nov. 2014.

[7] A. Fujita, A. Kameda, A. Kawazoe, Y. Miyao, "Overview of Todai robot project and evaluation framework of its NLP-based problem solving," in *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland, May 2014, pp. 2590–2597, European Language Resources Association (ELRA).

[8] I. A. Hameed, "Using natural language processing (NLP) for designing socially intelligent robots," in *2016 Joint IEEE International Conference on*

Development and Learning and Epigenetic Robotics (ICDL-EpiRob), Sept. 2016.

[9] T. Williams, "A consultant framework for natural language processing in integrated robot architectures," *IEEE Intelligent Informatics Bulletin*, vol. 18, pp. 10–14, 2017.

[10] W. Kahuttanaseth, A. Dressler, C. Netramai, "Commanding mobile robot movement based on natural language processing with RNN encoderdecoder," in *2018 5th International Conference on Business and Industrial Research (ICBIR)*, May 2018, pp. 161–166.

[11] W. Budiharto, V. Andreas, A. A. S. Gunawan, "Deep learning-based question answering system for intelligent humanoid robot," *Journal of Big Data*, vol. 7, p. 77, Dec. 2020, doi: 10.1186/s40537-020-00341-6.

[12] M. Seo, A. Kembhavi, A. Farhadi, H. Hajishirzi, "Bidirectional Attention Flow for Machine Comprehension," *arXiv:1611.01603 [cs]*, June 2018. arXiv: 1611.01603.

[13] S. Arroni, Y. Galán, X. Guzmán-Guzmán, E. R. Núñez-Valdez, A. Gómez, "Sentiment analysis and classification of hotel opinions in twitter with the transformer architecture," *International Journal of Interactive Multimedia and Artificial Intelligence, vol. 8, no. 1, pp. 53-63*, 2023.

[14] J. Zhou, T. Li, S. J. Fong, N. Dey, R. González-Crespo, "Exploring chatgpt's potential for consultation, recommendations and report diagnosis: Gastric cancer and gastroscopy reports' case," *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 8, no. 2, pp. 7-13, 2023.

[15] M. Rheu, J. Y. Shin, W. Peng, J. Huh-Yoo, "Systematic Review: Trust-Building Factors and Implications for Conversational Agent Design," *International Journal of Human–Computer Interaction*, vol. 37, pp. 81–96, Jan. 2021.

[16] R. B. Miller, "Response time in man-computer conversational transactions," in *Proceedings of the December 9-11, 1968, fall joint computer conference, part I*, 1968, pp. 267–277.

[17] T. Shiwa, T. Kanda, M. Imai, H. Ishiguro, N. Hagita, "How quickly should communication robots respond?," in *2008 3rd ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, 2008, pp. 153–160, IEEE.

[18] R. R. Murphy, T. Nomura, A. Billard, J. L. Burke, "Human–Robot Interaction," *IEEE Robotics Automation Magazine*, vol. 17, pp. 85–89, June 2010, doi: 10.1109/MRA.2010.936953. Conference Name: IEEE Robotics Automation Magazine.

[19] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin, "Attention is All you Need," in *Advances in Neural Information Processing Systems*, vol. 30, 2017, Curran Associates, Inc.

[20] I. Sutskever, O. Vinyals, Q. V. Le, "Sequence to Sequence Learning with Neural Networks," in *Advances in Neural Information Processing Systems*, vol. 27, 2014, Curran Associates, Inc.

[21] "The Illustrated Transformer – Jay Alammar – Visualizing machine learning one concept at a time." [Online]. Available: https://jalammar.github.io/illustrated-transformer/.

[22] D. Rothman, *Transformers for Natural Language Processing: Build innovative deep neural network architectures for NLP with Python, PyTorch, TensorFlow, BERT, RoBERTa, and more*. Packt Publishing Ltd, Jan. 2021. Google-Books-ID: Cr0YEAAAQBAJ.

[23] A. M. P. Bras¸oveanu, R. Andonie, "Visualizing Transformers for NLP: A Brief Survey," in *2020 24th International Conference Information Visualisation (IV)*, Sept. 2020, pp. 270–279. ISSN: 2375-0138.

[24] K. Weiss, T. M. Khoshgoftaar, D. Wang, "A survey of transfer learning," *Journal of Big Data*, vol. 3, p. 9, May 2016.

[25] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever, "Improving Language Understanding by Generative Pre-Training," *OpenAI*, p. 12, 2018.

[26] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, P. J. Liu, "Exploring the Limits of Transfer Learning with a Unified Text-to- Text Transformer," *arXiv:1910.10683 [cs, stat]*, July 2020. arXiv: 1910.10683.

[27] L. Floridi, M. Chiriatti, "GPT-3: Its Nature, Scope, Limits, and Consequences," *Minds and Machines*, vol. 30, pp. 681–694, Dec. 2020, doi: 10.1007/s11023-020-09548-1.

[28] C. Stevenson, I. Smal, M. Baas, R. Grasman, H. van der Maas, "Putting GPT-3's Creativity to the (Alternative Uses) Test," in *International Conference on Innovative Computing and Cloud Computing*, 2022, arXiv. Version Number: 1.

[29] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, D. Amodei, "Language Models are Few-Shot Learners," *ArXiv*, May 2020.

[30] J. Liu, D. Shen, Y. Zhang, B. Dolan, L. Carin, W. Chen, "What makes good in-context examples for gpt-3?," *arXiv preprint arXiv:2101.06804*, 2021.

[31] G. Poesia, O. Polozov, V. Le, A. Tiwari, G. Soares, C. Meek, S. Gulwani, "Synchromesh: Reliable code generation from pre-trained language models," *arXiv preprint arXiv:2201.11227*, 2022.

[32] T. Goyal, J. J. Li, G. Durrett, "News summarization and evaluation in the era of gpt-3," *arXiv preprint arXiv:2209.12356*, 2022.

[33] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, P. J. Liu, *et al.*, "Exploring the limits of transfer learning with a unified text-to- text transformer.," *The Journal of Machine Learning Research*, vol. 21, no. 140, pp. 1–67, 2020.

[34] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, *et al.*, "Transformers: State-of-the-art natural language processing," in *Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations*, 2020, pp. 38–45.

[35] L. Xue, N. Constant, A. Roberts, M. Kale, R. Al- Rfou, A. Siddhant, A. Barua, C. Raffel, "mT5: A massively multilingual pre-trained text-to-text transformer," in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Online, June 2021, pp. 483–498, Association for Computational Linguistics.

[36] Y. Yang, Y. Zhang, C. Tar, J. Baldridge, "PAWS- X: A cross-lingual adversarial dataset for paraphrase identification," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Hong Kong, China, Nov. 2019, pp. 3687–3692, Association for Computational Linguistics.

[37] J. Zhang, Y. Zhao, M. Saleh, P. J. Liu, "Pegasus: Pre- training with extracted gap-sentences for abstractive summarization," in *Proceedings of the 37th International Conference on Machine Learning*, ICML'20, 2020, JMLR.org.

[38] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[39] J. Canete, G. Chaperon, R. Fuentes, J.-H. Ho, H. Kang, J. Pérez, "Spanish pre-trained bert model and evaluation data," *Pml4dc at iclr*, vol. 2020, pp. 1–10, 2020.

[40] X. Zhao, B. F. Malle, "Spontaneous perspective taking toward robots: The unique impact of humanlike appearance," *Cognition*, vol. 224, p. 105076, July 2022, doi: 10.1016/j.cognition.2022.105076.

[41] M. A. Salichs, A. Castro, E. Salichs, E. Fernandez, M. Maroto, J. J. Gamboa, S. Marques, J. C. Castillo, F. Alonso, M. Malfaz, "Mini: A New Social Robot for the Elderly," *International Journal of Social Robotics*, vol. 12, pp. 1231–1249, Dec. 2020, doi: 10.1007/s12369-020-00687-0.

[42] K. Papineni, S. Roukos, T. Ward, W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, 2002, pp. 311–318.

[43] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, Y. Artzi, "BERTScore: Evaluating Text Generation with BERT," in *Eighth International Conference on Learning Representations*, Apr. 2020.

Javier Sevilla Salcedo

Javier Sevilla Salcedo is a Ph.D. candidate and researcher at the Robotics Lab at the Carlos III University of Madrid. His academic journey began with a B.Sc. in Industrial Electronics and Automation Engineering from the University of Jaén. He furthered his education with a M.Sc. in Robotics and Automation at his current institution. His career includes over four years of extensive research in diverse areas such as Artificial Intelligence, Robotics, and NLP, which involved his work at two prominent research labs. Although Natural Language Processing remains his main focus, his research integrates it with Social Robotics, Deep Learning, Robot Perception, and Cognitive Robotics.

Enrique Fernández Rodicio

Enrique Fernández Rodicio received the B.Sc. degree in Industrial Engineering from the Carlos III University of Madrid in 2014, the M.Sc. degree in Robotics and Automation from the Carlos III University of Madrid, Spain, in 2016, and the Ph.D in Electric, Electronic, and Automation Engineering from the Carlos III University of Madrid in 2021. He is a research assistant at the RoboticsLab Research Group, inside the Department of Systems Engineering and Automation of the Carlos III University of Madrid, Madrid, Spain. His present research lines are related to human-robot interaction, dialogue management, and expressiveness management. Enrique Fernández Rodicio received the B.Sc. degree in Industrial Engineering from the Carlos III University of Madrid in 2014, the M.Sc. degree in Robotics and Automation from the Carlos III University of Madrid, Spain, in 2016, and the Ph.D in Electric, Electronic, and Automation Engineering from the Carlos III University of Madrid in 2021. He is a research assistant at the RoboticsLab Research Group, inside the Department of Systems Engineering and Automation of the Carlos III University of Madrid, Madrid, Spain. His present research lines are related to human-robot interaction, dialogue management, and expressiveness management.

Laura Martín Galván

Laura Martín Galván received the BSc. degree in Electronic, Robotic, and Mechatronic Engineering from the University of Málaga, Spain, in 2020. She then received her MSc. in Robotics and Automation from the Carlos III University of Madrid in 2022. Since 2021, she has been involved in the Robotics Lab at the University Carlos III of Madrid, where her research interests have included Natural Language Processing with the use of Artificial Intelligence in Social Robots.

Álvaro Castro González

Álvaro Castro González received the B.Sc. degree in computer engineering from the University of León, León, Spain, in 2005, and the M.Sc. and Ph.D. degrees in robotics and automation from the Carlos III University of Madrid, Madrid, Spain, in 2008 and 2012, respectively. He is currently an Assistant Professor with the Department of Systems Engineering and Automation, Carlos III University of Madrid, and member of the Robotics Lab Research Group. He has been involved in several national, European, and corporate sponsored research projects. His research interests include human–robot interaction, social robots, expressiveness in robots, decision-making, and artificial emotions.

José Carlos Castillo

José Carlos Castillo is an assistant professor at the University Carlos III of Madrid. He obtained his Ph.D. in Computer Science from the University of Castilla-La Mancha, Spain, in 2012. From 2006 to 2012, he worked at the natural and artificial Interaction Systems group at the Albacete Research Institute of Informatics, Spain, working on computer vision techniques for detecting human activities and frameworks for intelligent monitoring and activity interpretation. From 2012 to 2013, he worked as a post-doctoral researcher at the Institute for Systems and Robotics (ISR), Instituto Superior Técnico (IST) of Lisbon, where he was involved in the development of networked robot systems, robotics and computer vision and intelligent control systems. Since September 2013, he has been combining teaching and research at the Robotics Lab at the University Carlos III of Madrid. He focuses on multimodal perception for Human–Robot Interaction for the mild cognitive impaired.

Miguel A. Salichs

Miguel A. Salichs is a full professor of the Systems Engineering and Automation Department at Carlos III University of Madrid (UC3M). He received the Electrical Engineering and Ph.D. degrees from Polytechnic University of Madrid. His research interests include autonomous social robots, multimodal human-robot interaction, mind models and cognitive architectures. He was Vicerrector of the UC3M, member of the Policy Committee of the International Federation of Automatic Control (IFAC), Chairman of the Technical Committee on Intelligent Autonomous Vehicles of IFAC, responsible of the Spanish National Research Program on Industrial Design and Production, President of the Spanish Society on Automation and Control (CEA), and the Spanish representative at the European Robotics Research Network (EURON). He is currently Coordinator of the Spanish Robotics Technology Platform (HispaRob), President of the Foundation of the Spanish Society on Automation and Control, and President of Area at the Spanish Research Agency.