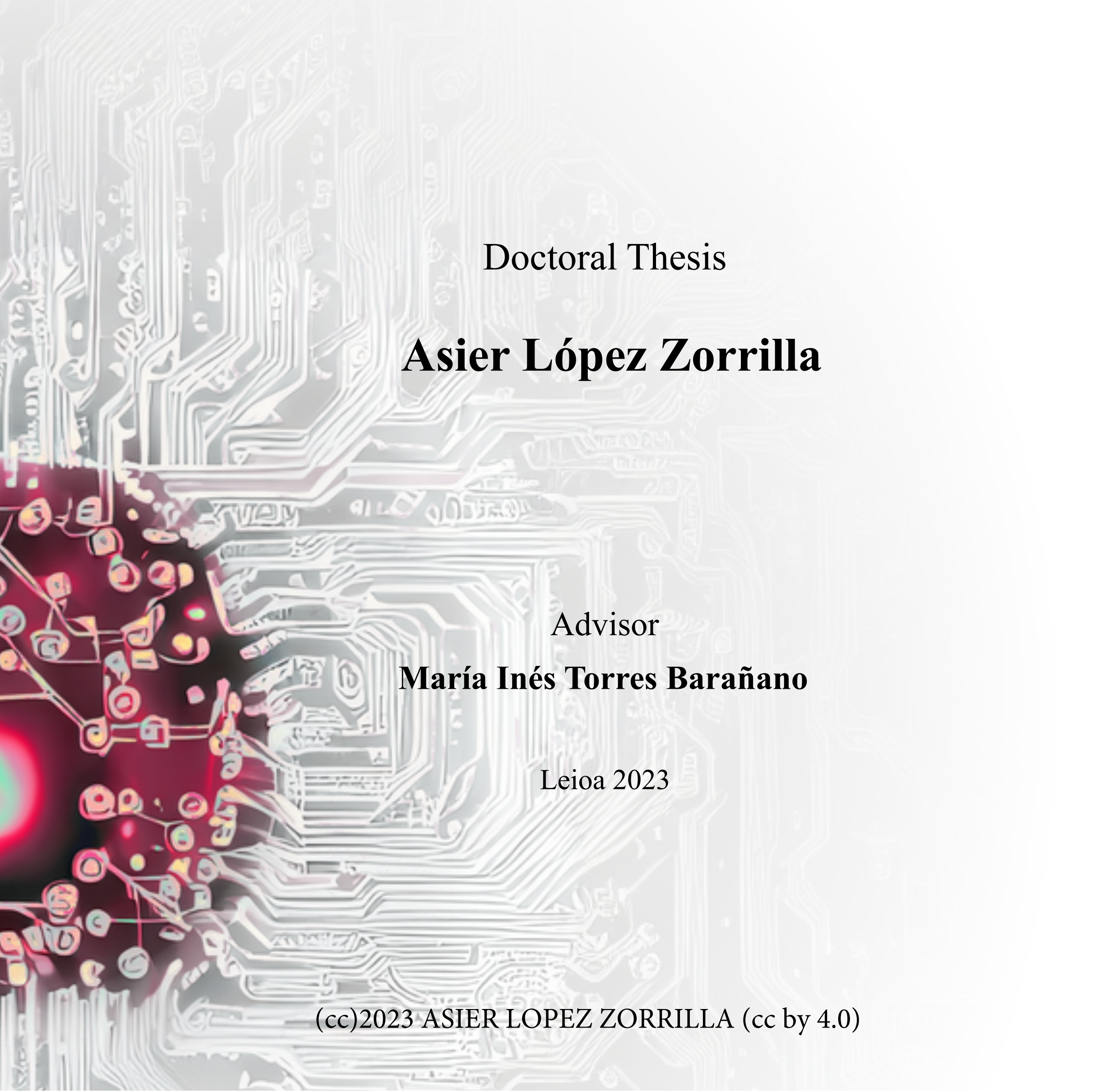
eman ta zabal zazu
Universidad del País Vasco
Euskal Herriko Unibertsitatea

ZTF-FCT
Zientzia eta Teknologia Fakultatea
Facultad de Ciencia y Tecnología

Towards structured neural spoken dialogue modelling



Doctoral Thesis

Asier López Zorrilla

Advisor
María Inés Torres Barañano

Leioa 2023

Abstract

In this thesis, we try to alleviate some of the weaknesses of the current approaches to dialogue modelling, one of the most challenging areas of Artificial Intelligence. We target three different types of dialogues (open-domain, task-oriented and coaching sessions), and use mainly machine learning algorithms to train dialogue models.

One challenge of open-domain chatbots is their lack of response variety, which can be tackled using Generative Adversarial Networks (GANs). We present two methodological contributions in this regard. On the one hand, we develop a method to circumvent the non-differentiability of text-processing GANs, which enables a gradient-based optimisation. On the other hand, we extend the conventional task of discriminators, which often operate at a single response level, to the batch level. Our proposed discriminators process and evaluate a set of responses, which makes them more robust and stable.

Meanwhile, two crucial aspects of task-oriented systems are their understanding capabilities because they need to correctly interpret what the user is looking for and their constraints), and the dialogue strategy. We propose a simple yet powerful way to improve spoken understanding and adapt the dialogue strategy by explicitly processing the user's speech signal through audio-processing transformer neural networks. We show that this improves the system's performance, especially with noisier Automatic Speech Recognisers.

Finally, coaching dialogues share properties of open-domain and task-oriented dialogues. They are somehow task-oriented because there are some tasks to be completed, such as detecting the user's objective or identifying which obstacles are not letting them fulfil their goal. However, there is no rush to complete the task, and it is more important to calmly converse with the user and make them aware of their own problems, obstacles and goals. In this context, on the one hand, we describe our collaboration in the EMPATHIC project, where a Virtual Coach capable of carrying out coaching dialogues about nutrition was built, using a modular Spoken Dialogue System. On the other hand, we model such dialogues with an end-to-end system based on Transfer Learning, for which we present two contributions. First, we show that learning dialogue

phase embeddings is a simple way for the model to generate more relevant (candidate) responses. Second, we build a deep learning system to rank these candidates according to their relevance and coherence, given the entire history of the dialogue. We show that combining these methods has a positive effect according to automatic and human evaluation metrics.

Contents

List of Figures

List of Tables

List of Abbreviations

AI Artificial Intelligence 1, 14, 74, 76, 79, 80, 82, 129

ASR Automatic Speech Recogniser 4, 53, 64–66, 129–131, 138–140, 142–146, 148–153, 155, 156, 158, 159

BPE Byte-Pair Encoding 23, 99, 133

CER Character Error Rate . . . . xi, 138, 139, 153

CHS Can't Help Score 139, 144, 150

CUQ Chatbot Usability Questionnaire . . . . xiv, 103, 112, 113

DM Dialogue Manager 3, 37, 44, 45, 54–57, 59, 60, 64, 67, 71, 72, 74, 130, 132, 157

FQ Follow-up Question 38, 41

GAN Generative Adversarial Network . ix, xiii, 3, 6–11, 13, 14, 16–18, 22, 23, 25–33, 93, 157, 158

GRU Gated Recurrent Unit 96, 97, 101, 125, 126

GSQ Goal Set Question . . . . xv, 38, 41, 168

HFQ Hedonic Feelings Questionnaire . . . . xiv, 104, 112–114

IC Intelligent Coach 55, 57

LM Language Model 9, 34, 60–62, 77, 78, 83, 85, 89, 90, 98–100, 111

LSTM Long Short Term Memory 11–13, 23, 132

MLE Maximum Likelihood Estimation ix, xiii, 5, 7, 8, 14–17, 20, 22, 24, 27–33

MLP Multilayer Perceptron 12, 13, 21, 23

MQ Motivational Question . . . . xv, 38, 41, 168

NER Name Entity Recogniser 138

NLG Natural Language Generation xiii, 3, 36, 37, 53, 54, 60–62, 64, 70–74, 99, 127, 135, 138, 157, 163

NLP Natural Language Processing 1, 4, 34, 77, 127, 129, 157

NLU Natural Language Understanding 44, 53, 54, 60, 64, 66, 70, 127

OGQ Option Generation Question . . . . xv, 38, 41, 169

OQ Obstacle Question . . . . xv, 38, 41, 169

PAQ Plan Action Question . . . . xv, 38, 41, 169

RL Reinforcement Learning 3, 7, 9, 32, 84, 130, 132, 137, 138, 140, 141, 144–146, 152, 154, 158

RNN Recurrent Neural Network 11, 12, 19, 21–23

RQ Reality Question . . . . xv, 38, 41, 169

SDS Spoken Dialogue System 1–4, 37, 44, 53, 56, 64, 72, 74, 129, 131, 150, 156, 158, 159

SL Supervised Learning ix, xi, xiv, xv, 3, 5, 15, 17, 130, 132, 140–151, 153, 154, 158

SOVV System Offered Valid Venue 139, 144, 150

SUS System Usability Questionnaire 43, 103

TRS Manual transcription 138, 140, 142, 143, 148, 150

TTS Text To Speech 54, 55, 70, 156, 159

UM User Model xi, xiv, xv, 130, 135, 138–140, 142–145, 147, 148, 152, 153, 155, 156

URS User Request Score 139, 144, 145, 150

VAAQ Virtual Agent Acceptability Questionnaire . . . . xiii, xiv, 43, 71–73

VAE Variational Autoencoder 8

VC Virtual Coach x, xiii, xiv, 3, 35–37, 39, 53–56, 58, 59, 62, 64, 66, 68, 70–74, 157, 158

WDH Whole Dialogue History x, 77–79, 86–88, 93, 96–98, 100–103, 105, 106, 110–112, 114, 115, 119, 127, 157, 158

WER Word Error Rate 66, 138, 139

WoZ Wizard of Oz ix, xiii, xiv, 35, 36, 39–43, 45, 50, 56, 62, 64–67, 71–74, 85, 86

WQ Warning Question 38, 41

Chapter 1

Introduction

Language is one of the most complex, distinctive, intriguing and interesting aspects of the human being. Arguably, it is the most important evolutionary capability developed by any species over the last few million years (Nowak, 2000). There is no other tool as efficient as language for us to exchange information. Using language, we can easily talk about the past, describe the present and make plans for the future. We can share our thoughts about the world surrounding us, produce explanations and find solutions collectively. We can teach and learn from each other, express our feelings and build strong social relationships.

Thus, it is unsurprising that language and dialogue modelling are some of the most challenging research areas within Artificial Intelligence (AI). A human dialogue can be as simple as asking the other's name and briefly presenting ourselves, but also as complex as discussing about the origin of life, or trying to convince another about certain political positions. In order to understand or take part in a conversation, a large amount of factual information about the topic is desirable, but also the cultural and social context, the particular situation of each partner, as well as everyone's objective in the dialogue. Evolution has provided humans with a powerful brain with dedicated parts to understand and produce language. This way, we are able to keep such conversations almost effortlessly. However, despite recent technological and methodological advances in Natural Language Processing (NLP), AI systems still struggle with this task. For instance, many chatbots are known to produce repetitive and not-very-informative responses in some cases, they struggle to keep long-term coherence and to carry out long and complex dialogues, and Spoken Dialogue Systems (SDSs) are unable to capture all the information present in the users' speech. In this thesis, we analyse some of the most novel and promising trends in dialogue modelling, and propose several methods to improve their performance and avoid, or at least attenuate, some of the aforementioned drawbacks.

With this we hope to bridge the gap, even if slightly, between the conversational skills of humans and machines.

Throughout this dissertation, we focus on different aspects of dialogue modelling, from the perspective of the dialogue goal, the modelling techniques and the input modality. Regarding the dialogue itself, we study conventional open-domain (Chapter 2) and task-oriented (Chapter 5) dialogues, as well as novel (in terms of dialogue modelling) coaching sessions (Chapters 3 and 4), which share some properties of open-domain and task-oriented dialogues.

In open-domain dialogues or chit-chats, there is no specific topic to talk about or task to carry out, the only goal is to generate appropriate and meaningful responses given a dialogue context and to keep the user engaged. One challenge of open-domain chatbots is their lack of response variety. As a consequence of the large number of topics they need to be able to talk about, they often learn to produce generic but dull responses. We tackle this problem in Chapter 2.

Meanwhile, task-oriented dialogue systems are often developed to provide the user with information or services that they request as soon as possible, such as hotel booking. This scenario is much closer to real-life applications of SDS, and therefore the dialogues are often spoken, and not text-based. Two crucial aspects of such systems are their understanding capabilities (because they need to correctly interpret what the user is looking for and their constraints), and the dialogue strategy. We propose a simple yet powerful way to improve user understanding and adapt the dialogue strategy at the same time through cutting-edge audio embeddings in Chapter 5.

Finally, coaching dialogues have peculiarities that do not allow to easily classify them into the aforementioned categories. As we explain more in-depth in Chapters 3 and 4, they are somehow task-oriented because there are some tasks to be completed, such as detecting the user's objective or identifying which obstacles are not letting them fulfill their goal. However, the dialogue is definitely not carried out in a conventional task-oriented manner. There is no rush to complete the task, and it is more important to calmly converse with the user and make them aware of their own problems, obstacles and potential goals they want to achieve. In this sense, coaching is also related to open-domain dialogues. However, coaching dialogues follow a clear and well-structured strategy. Constraining open-domain dialogue models to follow such a structured strategy while fulfilling the goals of the conversation is our main research question in this regard.

As for the modelling techniques, we mostly explore machine learning or data-driven approaches (Chapters 2, 4 and 5) as opposed to classical modular SDS architectures (Chapter 3). Classical SDSs still present some advantages, especially for commercial and industrial applications, where greater control over the behaviour of the system is critical. However, data-driven methodologies represent a more promising and attractive research approach.

Among the multiple options to develop statistical dialogue models, such as POMDPs (Young, 2000) or attributed bi-automata (Torres, 2013), we have opted for the popular and contemporary neural network/deep learning paradigm (LeCun et al., 2015) as our main research tool. We examine three main frameworks to train them: Generative Adversarial Networks (GANs) (Chapter 2) for open-domain, Transfer Learning (Chapter 4) for coaching sessions and Reinforcement Learning (RL) (Chapter 5) for task-oriented. Additionally, we employ Supervised Learning (SL) as a baseline for comparison purposes in some experiments.

Last, we experiment with both text-based and spoken dialogue systems. During the last few years, most research has been carried out with text-based dialogue models, due to most corpora not including audio and also because it is just simpler and there are less issues to cope with. Nonetheless, SDSs are still of big interest, because text-based interactions are not feasible or desirable in many situations. The dialogue models developed in Chapters 2 and 4 use text as input, whereas the dialogue systems presented in Chapters 3 and 5 are spoken.

Having presented an overview of the work carried out, let us present the contributions and structure of the rest of thesis, roughly in chronological order, and from more open to closer domain dialogue modelling:

Chapter 2 presents two methodological contributions for dialogue generating GANs. On the one hand, we develop a method to circumvent the non-differentiability of text-processing GANs, which enables a gradient-based optimisation. On the other hand, we extend the conventional task of discriminators, which operate at a single response level, to the batch level. Our proposed discriminators process and evaluate a set of responses, which makes them more robust. Consequently, batch-level GANs offers more varied responses and a more stable learning process than SL and response-level GAN baselines.

Chapter 3 introduces the European project EMPATHIC, the framework for Chapters 3 and 4. The main goal of this project was to develop a Virtual Coach (VC) (a SDS) to improve independent healthy-life-years of the elderly. After presenting the target coaching dialogues to be modelled, Chapter 3 describes our contributions to the Natural Language Generation (NLG) and Dialogue Manager (DM) modules of the VC, and analyses their behaviour. On the other hand, we describe the acquisition and labelling of a corpus of coaching dialogues, which is used in Chapter 4 to develop a data-driven end-to-end coaching dialogue system.

Chapter 4 shows how to leverage transfer learning for the EMPATHIC coaching task. Fine-tuning large pretrained language models is an attractive and relatively powerful approach to end-to-end dialogue modelling (Wolf et al., 2019). It yields models that are often coherent in the short term, but which struggle to model long-term dialogue strategies. This is not a problem for short conversations, but the target coaching sessions are relatively long and a concise structure needs to be followed. We overcome or alleviate this issue with two main proposals. First, we show that learning dialogue phase embeddings is a simple way for the fine-tuned model to generate more relevant (candidate) responses. Second, we build a deep learning system to rank these candidates according to their relevance and coherence, given the entire history of the dialogue. We show that combining these methods has a positive effect according to automatic and human evaluation metrics.

Chapter 5 focuses on data-driven task-oriented SDSs. One input source on which these systems base their decisions are user speech signals. Speech signals are often mapped into words first via an Automatic Speech Recogniser (ASR), and then NLP techniques are applied to understand the user and act accordingly. However, this approach is highly dependent on the ASR providing a correct transcription, which might not be the case in noisy environments, or if the user is non-native or has an uncommon accent (Litman et al., 2018). More importantly, it ignores important information in users' speech, such as their emotional mood, prosody, or the noise level of the environment, which could be key to carry out a better dialogue strategy. We explore how to include this information in end-to-end SDSs via cutting-edge audio embeddings. We show (with automatic and human evaluations) that they significantly improve SDS performance, especially if a noisier ASR is employed. Audio embeddings lead to a better user understanding, and they can also be used to determine when an ASR output is not too reliable.

Finally, **Chapter 6** presents our concluding remarks. Subsequently, we provide the **List of publications** produced throughout the PhD.

Chapter 2

Differentiable batch-level GANs for open domain dialogue

2.1 | Introduction

Open-domain dialogue systems or chatbots are systems deployed to interact with humans offering coherent responses according to the dialogue history. Unlike task-oriented dialogue systems, there is no specific goal to be achieved by any party during the interaction. The system's only goal is to generate appropriate, relevant, meaningful and human-like utterances.

This area of research has gained an increasing amount of interest from the community since the advent of sequence-to-sequence neural network models (Sutskever et al., 2014). These neural networks are capable of processing and generating sequences of data of arbitrary length, which makes them very suitable for this research (Vinyals and Le, 2015; Sordoni et al., 2015). The task of open-domain dialogue generation can easily be cast as a sequence transduction problem, where the input is the sequence of words corresponding to the last user's utterance(s), and the output are the words of the system's response.

Such neural models have usually been learnt from corpora composed of dialogue context-response pairs, via SL minimising the token-level cross-entropy loss, a method often called Maximum Likelihood Estimation (MLE) (Vinyals and Le, 2015). Movies subtitles, Twitter or online forums can be used as the source of these data. In this framework, the neural network is trained to minimise a distance between the generated response and the desired one. Although interesting performances can be obtained with this procedure, it frequently yields models that tend to generate dull and safe responses, which appear frequently in the corpus, such as *I don't know* or *I'm sorry* (Vinyals and Le, 2015; Serban et al., 2016; Li et al., 2016a). Even if much bigger training corpora are used, this phenomenon can only be alleviated slightly (Holtzman et al., 2019; Welleck et al., 2020; Li et al., 2020). One important problem of this methodology is that it does not take into account the *one-to-many* property of conversational input-output pairs (Tuan and Lee, 2019), i.e. the fact that many (and potentially very diverse) responses may be valid given the same context. For instance, there are countless valid and informative replies to the question *"What are you doing tomorrow?"*. However, the cross-entropy loss function only considers one valid response at a time. Every time the same (or very similar) input appears in the training corpus, the new *correct* output will be a new (and potentially different) one. Thus, this kind of chatbots converges to producing generic responses that often appear as responses to many outputs. A more in-depth analysis of the causes of this lack of variety in the responses can be found in Jiang and de Rijke (2018).

One of the first efforts to alleviate this issue consisted in approaching the problem with GANs (Goodfellow et al., 2014), as these had shown promising results in many data generation tasks. Other alternatives are discussed in the related work in Section 2.2. GANs allow many correct outputs, which makes much more sense for dialogue generation. The learning methodology for GANs involves training two neural networks, a generator and a discriminator, in an adversarial fashion. The generator tries to learn a data distribution, while the discriminator learns whether a given sample corresponds to the training data or has been generated by the generator. In the context of dialogue systems, the generator would be the sequence-to-sequence model, and the discriminator would act as a kind of automatic Turing Test. We give an overview of this framework in Section 2.3.

GANs were first successful in image generation tasks (Goodfellow et al., 2014; Denton et al., 2015). Text-related problems, such as machine translation (Wu et al., 2017), text generation (Yu et al., 2017; Xu et al., 2018) or image captioning (Shetty et al., 2017) were also tackled within this framework later. GANs have been applied in the research of dialogue systems too, yet only on a reduced amount of occasions. For instance, Bowman et al. (2015) and Kannan and Vinyals (2017) experimented with training discriminators that could measure the quality of the utterances generated by chatbots. On the other hand, Li et al. (2017) and Hori et al. (2019) went a step further and trained neural dialogue systems using adversarial learning.

However, gradient-based optimisation methods are not directly applicable for text-based GANs, which hardens their implementation and stability. In short, in order to adversarially optimise the generator, the output of the discriminator has to be minimised. Therefore, the input of the discriminator should be differentiable with respect to the output of the generator and its parameters. However, the output of the generator are discrete (and therefore non-differentiable) tokens, and the input of the discriminator the word vectors corresponding to those tokens. Thus, those word vectors cannot be directly used during training unless approximate derivatives are constructed. This differentiability issue is explained more in-depth in Section 2.4, in Section 2.4.1 more precisely.

In this context, our first contribution is a novel methodology to avoid this non-differentiability: approximated differentiable word vectors produced by a *top-k softmax*. In a nutshell, we show that the weighted average of the word vectors of the most probable tokens at each generation step are similar enough to the most likely one. In fact, the nearest neighbour of the averaged (or approximated) word vector corresponds to the most probable token 98% of the time, but with the advantage that they are differentiable. This methodology for allowing GANs to be trained is simpler than most of the alternatives found in the literature such as (Straight-Through) Gumble-softmax, RL or soft-argmax (see a comparison in Section 2.2.2); and we show that it is valid to train more diverse and semantically coherent generators than MLE baselines in the English version of the OpenSubtitles corpus (Lison and Tiedemann, 2016; Lison et al., 2019). Section 2.4 shows our proposal in detail.

Our second contribution is an extension of the role of the discriminator in GANs for dialogue generation. To the best of our knowledge, all the discriminators in related works operate at the response level. That is, they evaluate how appropriate a single response is given a dialogue context. We propose to provide the discriminators with a wider view of the generator's behaviour. We name our proposal batch-level discriminators. They evaluate a set of responses given a set of dialogue contexts. Thus, they are less sensitive to complex yet repetitive or not very informative responses. We give a detailed explanation and present the model in Section 2.5, and we show that they outperform both the MLE baseline and the response-level GAN in our experiments, in Sections 2.6 and 2.7.

Finally, we would like to mention that the work carried out in this chapter is an extension of the research presented in López Zorrilla et al. (2021a), where the top-K softmax procedure was introduced. We also showed that this methodology could be used for languages with fewer resources, such as Basque (López Zorrilla et al., 2019, 2020). The idea of batch-level discriminators is described in López Zorrilla et al. (2023), currently under revision.

The rest of the chapter is organised as follows. We describe related works in Section 2.2. Before getting into the details of our contributions, we give an overview of the GAN framework for dialogue generation in Section 2.3. We explain the differentiability problem and how the proposed approximated word vectors via a top-K softmax address the issue in Section 2.4. In Section 2.5 we introduce the batch-level GANs. Section 2.6 presents the experimental framework used to validate our proposals, as well as the training and implementation details. We provide the results of these experiments in Section 2.7, and end with some concluding remarks in Section 2.8.

2.2 | Related work

Let us discuss how our research relates to other works in the literature. We describe alternative approaches to increase the variety of sequence-to-sequence dialogue models in Section 2.2.1, analyse how others tackle the non-differentiability of text-based GANs in Section 2.2.2, and comment some related works regarding batch-level GANs in Section 2.2.3.

2.2.1 | Increasing the variety of sequence-to-sequence dialogue models

The lack of variety and the non-informativeness of neural dialogue models have been tackled in several ways besides using GANs. A family of solutions can be the methods consisting in modifying the training objective to avoid the limitations of MLE. The work by Li et al. (2016a) was one of the first proposals in this regard, where using Maximum Mutual Information as the objective function was explored. Zhao et al. (2017) proposed a *bag-of-words* loss for dialogue generation with Variational Autoencoders (VAEs). Frequency-aware losses were proposed by Jiang et al. (2019) and Li et al. (2020) to alleviate the low variety problem. Negative training (i.e. explicitly discouraging non-desired behaviours during training) has also been employed to this end (He and Glass, 2020; Li et al., 2022b). More complex training schemes, such as backward reasoning (training the system to predict the dialogue context given its generated response), have been proposed too (Li et al., 2021).

Another hypothesis of why sequence-to-sequence models often end up producing generic and dull responses holds that the dialogue history alone might not be sufficient for producing informative responses. Thus, a set of research works focus on providing neural dialogue systems with more information and context rather than improving the training procedure. For example, some authors have tried to give their models some kind of consistent personality (Li et al., 2016b; Zhang et al., 2018; Zheng et al., 2020; Cao et al., 2022). Others have included additional information about the topic of the conversation (Xing et al., 2016, 2017; Wu et al., 2020, 2021a) or generated dialogues based on knowledge of movies (Sun et al., 2020). More ambitiously, Ghazvininejad et al. (2018); Dinan et al. (2018) aim at enhancing neural dialogue models with the capability of reading and retrieving information from Wikipedia or similar sources, and conditioning responses based on it. In the same vein, Komeili et al. (2022) propose to learn Internet search queries given the dialogue context, and generate the response based on the search results.

Somehow related to these approaches, we can find the popular transfer learning methodology that relies on fine-tuning large pretrained Language Models (LMs) such as GPT (Radford et al., 2018), GPT-2 (Budzianowski and Vulic, 2019) or BERT (Devlin et al., 2019) on conversational data. Since these LMs are trained on large corpora with information about many topics, the resulting dialogue models still retain some of that knowledge, which contributes to generating more informative responses. Additionally, this approach is simpler in terms of implementation, which has contributed to its popularity. Furthermore, it has been shown that the transfer learning methodology is suitable for both open domain (Wolf et al., 2019; Zhang et al., 2020b; Roller et al., 2020) and task-oriented dialogue (Ham et al., 2020; Hosseini-Asl et al., 2020; Peng et al., 2020). In fact, we adopt this framework for the neural dialogue systems developed in Chapters 4 and 5.

2.2.2 | Dealing with the non-differentiability of text GANs

Besides our proposed top-K softmax and approximated word vectors approach to allow the differentiability of text GANs, there are other alternatives that can be found in the literature. The most widely used one is the RL approach proposed by Yu et al. (2017), which has been the basis for many related works (Li et al., 2017; Lin et al., 2017; Hori et al., 2019; Tuan and Lee, 2019; Nabeel et al., 2019). The main idea behind this approach is to use the discriminator's output as the reward function to train the generator. In other words, the gradient fed to the generator is a function of the discriminator's output. Thus, outputs that contributed to a good evaluation are reinforced, whereas negatively evaluated sequences are discouraged. One problem with this is that all the actions (sampling of the tokens) are assigned the same reward, which may not be appropriate, because a subsequence could be right while another part of the response could be inappropriate. Some authors have proposed some workarounds to provide independent rewards for each token (Li et al., 2017; Su et al., 2018; Tuan and Lee, 2019), which improve the performance of the generators, but are either more complex or less computationally efficient. In our approach, each token (its approximated word vector, more precisely) is naturally assigned a different gradient. Moreover, each of the top-K tokens at each generation step gets a different gradient, which should stabilise and improve the training. We explain this in detail in Section 2.4.

Other works (Kusner and Hernández-Lobato, 2016) tackle this problem with the concrete or Gumbel-softmax distribution (Maddison et al., 2016; Jang et al., 2016) in one way or another. This is a continuous relaxation of discrete random variables. In short, it transforms a probability distribution into a relaxed one-hot vector corresponding to a randomly taken sample from that distribution. That relaxed vector is different from the result of the top-K softmax in two important aspects. First, it is nondeterministic, which could be interesting but also unnecessary for our application. Second, all its elements are nonzero, which means that approximating a word vector as a weighted average according to those probabilities would imply mixing all the word vectors in the vocabulary, which again seems inadequate or excessive for our application.

A discrete version of this transformation is the Straight-Through Gumbel-softmax estimator (Bengio et al., 2013; Jang et al., 2016), which was employed by Lu et al. (2017) and Shetty et al. (2017). It serves to approximate the gradients of a one-hot vector sampled according to a probability distribution. Thus, it avoids the problem of averaging over all the word vectors, but it is still non-deterministic and non-differentiable. Nonetheless, this method provides an estimation of the gradients in this scenario. The main drawback is that it could be unstable due to the discrepancies between the forward and backward passes, as stated in the original work (Jang et al., 2016).

Last, we would like to mention some more alternatives to dealing with the differentiability issue of text GANs. Gulrajani et al. (2017) proposed to use the raw softmax outputs of the generator as the input to the discriminator, without word embeddings. However, they only applied this method to a toy example, so it is unsure whether it could serve in real-world scenarios. On the other hand, Zhang et al. (2016) uses a soft-argmax transformation, which is probably the closest to our work. First, they force all the logits to be greater than 1 (or equal to 0), then they multiply them by a constant, compute the softmax and generate averaged word vectors. This method is similar to ours with the k in the top-K softmax equal to the vocabulary size. Nonetheless, our experiments (see Section 2.7.1) indicate that the lower values of k result in a much better approximation of the most probable word vectors.

2.2.3 | BATCH-LEVEL GANs

Batch or minibatch discrimination was first proposed by Salimans et al. (2016), although in a very different form than our proposal. They compute/learn a set of handcrafted batch-level statistics and include it in a layer at the end of the discriminator. Karras et al. (2018) propose a similar strategy; to compute the standard deviation of input features and feed it to the last discriminator layer. However, these two discriminators still work over single samples, rather than batches. Closer to our GAN architecture, Lucas et al. (2018) present a permutation-invariant discriminator architecture that processes sets of instances. They propose to train their discriminator with mixed batches of real and fake samples, and to predict the ratio of real instances. These works show the potential of batch-level discrimination in the image generation task. Nevertheless, we are not aware of any similar work for dialogue –or, in general, text– applications.

2.3 | The GAN framework

Let us present the GAN framework for dialogue generation. We first introduce the two neural networks that are trained adversarially in Section 2.3.1, and then explain the optimisation procedure in Section 2.3.2.

2.3.1 | Components

2.3.1.1 | Generator

In the GAN framework for dialogue, the generator is in charge of generating the system or bot's response given the dialogue history. Since we are interested in end-to-end dialogue modelling, both the dialogue history and the response are represented as a sequence of tokens or words. In our experiments, we restrict the dialogue history to the last turn, because we found it challenging enough to make the GANs converge this way. Our generator is a Recurrent Neural Network (RNN) sequence-to-sequence network with attention (Bahdanau et al., 2014), as depicted in Figure 2.1.

The network produces a response as follows. Given an input sequence of length T of discrete integer tokens $x = x_1, x_2, ..., x_T$, the corresponding sequence of vectorial word representations $\mathbf{v} = \mathbf{v}_1, \mathbf{v}_2, ..., \mathbf{v}_T$ can be obtained via the word vector matrix $\mathbf{W}$, just by taking the corresponding row $\mathbf{v}_i = \mathbf{W}[x_i]$ per each token x_i. The size of $\mathbf{W}$ is $V \times D$, where V is the vocabulary size and D the dimension of each word vector. The encoder takes this sequence of vectors and produces another sequence of vectors of the same length $\mathbf{h} = \mathbf{h}_1, \mathbf{h}_2, ..., \mathbf{h}_T = \text{encoder}(\mathbf{v})$. In our work, the encoder is a deep bidirectional Long Short Term Memory (LSTM) RNN (Hochreiter and Schmidhuber, 1997).

To proceed with the generation of the output sequence $y = y_1, y_2, ..., y_\tau$,

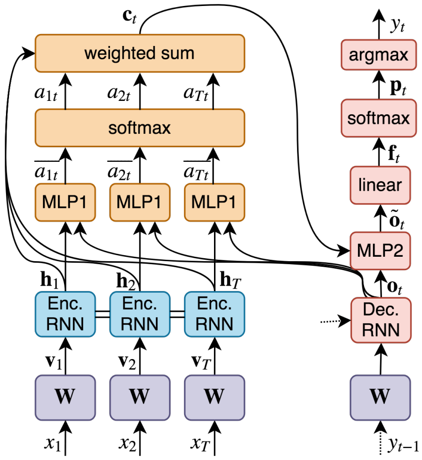


Figure 2.1.: A diagram of the chosen sequence-to-sequence network: blue transformations refer to the encoder, orange to the attention mechanism, purple to the word matrix, and red to the decoder. For simplicity, only the time step t of the decoding is shown.

a global attention mechanism is applied as in (Luong et al., 2015). At the time step t of generation, the decoder is fed with the discrete integer token generated at the previous time step, y_{t-1}. Then the corresponding word vector $\mathbf{W}[y_{t-1}]$ is used as input to the decoder's RNN, which returns $\mathbf{o}_t$. Of course, due to the architecture of RNNs, $\mathbf{o}_t$ is conditioned, though implicitly, not only to y_{t-1} but also to all the previously generated tokens. In our experiments this neural network is also a deep LSTM. $\mathbf{o}_t$ is then transformed to $\tilde{\mathbf{o}}_t$ via a Multilayer Perceptron (MLP), named MLP2, that takes as input o_t and also $\mathbf{c}_t$, the context-vector produced by the attention mechanism at time step t. $\mathbf{c}_t$ is a weighted average of the encoder's output vectors:

$$\mathbf{c}_t = \sum_{j=1}^{T} a_{jt}\mathbf{h}_j \ , \tag{2.1}$$

where a_{jt} is the score between $\mathbf{h}_j$ and $\mathbf{o}_t$, i.e., how much attention should be put on the output of the encoder at the encoding time step j on the time step t of the decoding phase. a_{jt} is a softmax-normalised scalar output of another MLP, MLP1, that takes as input $\mathbf{h}_j$ and $\mathbf{o}_t$, and outputs $\overline{a_{jt}}$. With the softmax normalisation, we ensure that all the scores at time step t are positive and sum one:

$$a_{jt} = \frac{\exp\left(\overline{a_{jt}}\right)}{\sum_{j'=1}^{T} \exp(\overline{a_{j't}})} \tag{2.2}$$

Finally, $\tilde{\mathbf{o}}_t$ is linearly projected to a vector of dimension V: $\mathbf{f}_t = \text{linear}(\tilde{\mathbf{o}}_t)$. This vector represents an unnormalised probability distribution over all possible words in the vocabulary. A softmax normalisation is then applied to $\mathbf{f}_t$ to get $\mathbf{p}_t = \text{softmax}(\mathbf{f}_t)$, the normalised version of $\mathbf{f}_t$. The output token at time step t, y_t, can be sampled from $\mathbf{p}_t$ taking the argument of the maxima:

$$y_t = \arg\max_i (\mathbf{p}_t[i]) \tag{2.3}$$

The generation stops at time τ, when the end-of-sequence token is output.

2.3.1.2 | Baseline Discriminator

The typical task of the discriminator is to judge how natural or human a response is given the dialogue history. As aforementioned, one of our contributions is to redefine or extend this concept to the batch-level, and perform this evaluation based on a set of input-outputs, rather than just one. However, let us first explain the idea of typical discriminators for dialogue modelling. We describe our proposal in more detail later in Section 2.5.

Our baseline discriminator is composed of two deep bidirectional LSTMs, as illustrated in Figure 2.2. One is devoted to process the dialogue history x, and the other one, the response r. The integer sequences x and r are converted to word vector sequences via the same word vector matrix $\mathbf{W}$ as the generator. Then, the last outputs of the encoders are concatenated and processed by a standard MLP. Last, a scalar between 0 and 1 is output (using a sigmoid activation function), which indicates the probability of r being produced by a bot. In other words, it should output values closer to 0 if r was present in the corpus, and closer to 1 otherwise.

2.3.2 | Training procedure

Before getting into the details of the loss functions to train the neural networks (Section 2.3.2.1) and the training loop (Section 2.3.2.2), we provide an overview of the training procedure (Section 2.3.2.3) of dialogue GANs.

2.3.2.1 | Overview

The methodology to train a dialogue system in the GAN framework involves iteratively updating the generator and the discriminator. The generator is trained to fool the discriminator and make it predict that its responses are human-like,

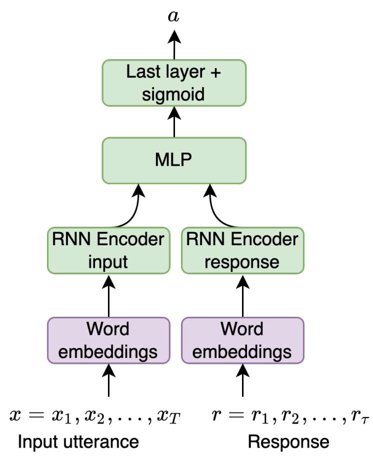
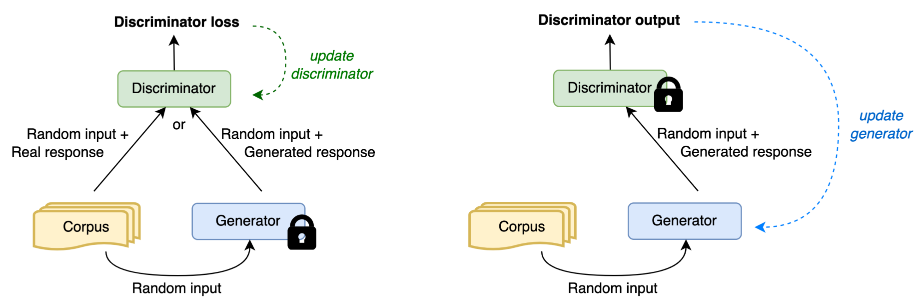


Figure 2.2.: The architecture of the baseline discriminator for dialogue GANs.

and in contrast the discriminator is trained to distinguish between human and bot responses. In most AI areas, this is done with two optimisation procedures: 1) the discriminator is trained to discriminate between samples generated by the generator and sampled from a corpus; and 2) the generator is trained to minimise the output of the discriminator (assuming that lower discriminator outputs correspond to samples of the corpus). This process is illustrated in Figure 2.3.



(a) Optimisation of the discriminator.

(b) Optimisation of the generator.

Figure 2.3.: Main two steps for the GAN optimisation procedure. The lock indicates when the parameters of the networks are frozen.

Additionally and specifically for the dialogue task, a third optimisation step is usually introduced in order to make the whole optimisation process more stable. It consists of performing a MLE of the parameters of the generator to predict the response in the corpus, as represented in Figure 2.4. This approach has been adopted in many works, such as Li et al. (2017); Hori et al. (2019).

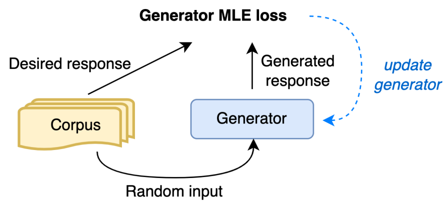


Figure 2.4.: MLE of the parameters of the generator via SL.

2.3.2.2 | Loss functions

Before getting into the details of the training loop and its implementation, let us define the loss functions corresponding to the aforementioned three optimisation procedures.

A MLE of the parameters of the generator is carried out by minimising the token-level cross-entropy loss L_{MLE} (Figure 2.4):

$$L_{MLE} = \frac{1}{|\mathcal{B}_{\mathcal{MLE}}|} \sum_{x,s \in \mathcal{B}_{\mathcal{MLE}}} \frac{1}{|s|} \sum_{t=1}^{|s|} -\log \mathbf{p}_t[s_t] \ , \tag{2.4}$$

where $\mathcal{B}_{\mathcal{MLE}}$ is a batch composed of pairs of inputs x and desired outputs s sampled from the training data, s_t each of the words in s, and $\mathbf{p}_t[s_t]$ the output of the network in the t-th time step corresponding to the token s_t. We omit the output's dependence on x to keep the notation simple. We employ the teacher forcing strategy, i.e., in the t-th step of the decoding we feed the ground true token s_{t-1} to the decoder's RNN instead of the prediction y_{t-1}. We experimented with other sampling techniques, such as scheduled sampling (Bengio et al., 2015), but did not find any improvement.

Regarding the optimisation of the discriminator (Figure 2.3a), its parameters are updated to minimise a binary cross-entropy loss:

$$L_D = \frac{1}{|\mathcal{B}_D|} \sum_{x,r,l \in \mathcal{B}_D} -\left[l \cdot \log a + (1-l) \cdot \log(1-a)\right] \ , \tag{2.5}$$

where $\mathcal{B}_D$ is a batch composed of tuples of input utterances x, responses r, and Boolean labels l indicating whether r was sampled from the corpus ($l = 0$) or generated by the generator ($l = 1$), and a the output of the network given x and r.

Last, the adversarial loss for the generator is the output of the discriminator (Figure 2.3b), after the latter is fed with a batch of input utterances and the responses of the generator to those same input utterances:

$$L_G = \frac{1}{|\mathcal{B}_G|} \sum_{x \in \mathcal{B}_G} a \ , \tag{2.6}$$

where $\mathcal{B}_G$ is a batch composed of input utterances x. a is the output of the discriminator given x.

2.3.2.3 | Training loop

The idea of GANs is to carry out the three aforementioned optimisation processes iteratively. However, it is common to first pretrain the generator and then discriminator to prevent, once again, the GAN from diverging. We also pretrain the word vector matrix in the same corpus, using fastext (Busta et al., 2015). The generator is pretrained according to the MLE criteria (Equation 2.4), and the discriminator with the responses generated by the pretrained generator and with responses from the corpus. In order to stabilise the rest of the training process and to avoid the catastrophic forgetting phenomenon of the discriminator, we sample responses of the generator to a given input periodically, and add them to a corpus of generator's turns denoted as $\mathcal{C}_D$. Thus, the discriminator is fed with outputs of different versions of the generator, but the outputs of the most recent ones are selected with a higher probability. Additionally, we also employ a heuristic to filter *good* bot responses that may not serve well as negative examples, as explained in more detail in Section 2.6.4.

Then, the main training loop starts, and the generator and discriminator are trained adversarially for many iterations. An iteration starts by training the generator to minimise the output of the discriminator according to Equation 2.6 during a number of iterations. This optimisation process is not trivial and we focus on it later in Section 2.4. After the optimisation of the generator we expand the corpus $\mathcal{C}_D$ with the current state of the generator, and train the discriminator during another number of iterations. We finally repeat this process by training the generator, adding samples to $\mathcal{C}_D$ and training the discriminator, but this time training the corpus according to the MLE criteria.

The whole procedure is summarised in Algorithm 1.

Algorithm 1: An Adversarial Training Strategy for Neural Dialogue Models.

Require: Generator G, Discriminator D, Corpus $\mathcal{C}$, training hyper-parameters.
Pretrain word vector matrix $\mathbf{W}$ on $\mathcal{C}$.
Pretrain G minimising L_{MLE} (Equation 2.4).
Initialise $\mathcal{C}_D$ with G's responses.
Pretrain D minimising L_D (Equation 2.5).
for the number of total iterations, and with a decaying learning rate **do**
Update G minimising L_G on inputs x in $\mathcal{C}$ (Equation 2.6).
Add (x, y) pairs to $\mathcal{C}_D$ using G.
Update D minimising L_D.
Update G minimising L_{MLE} on $\mathcal{C}$.
Add (x, y) pairs to $\mathcal{C}_D$ using G.
Update D minimising L_D.

2.4 | Top-K Softmax and approximated word vectors

Let us get into the details of the differentiability problem of text-generating GANs (Section 2.4.1) and explain our proposal to avoid this issue (Section 2.4.2).

2.4.1 | The differentiability problem

It is straightforward to optimise the parameters of the discriminator to minimise the loss function L_D (Equation 2.5, Figure 2.3a), and also the generator's parameters to minimise the MLE loss (Equation 2.4, Figure 2.4). The losses are differentiable with respect to the parameters of the corresponding networks, and therefore standard SL gradient-based methods can be employed to optimise the models according to Algorithm 1.

However, it is not possible to differentiate the output of the discriminator, or L_G (Equation 2.6, Figure 2.3b), with respect to the parameters of the generator. This happens because the sequence of token probability distributions of the generator output are transformed into discrete–and therefore not differentiable– tokens, so their word vectors can be processed by the discriminator. This is done via a (non-differentiable) argmax operation. Additionally, selecting the word vector corresponding to a discrete token is not differentiable either, as represented in Equation 2.7, and later in Figure 2.6a.

$$\mathbf{f}_t \xrightarrow[\longleftarrow]{\text{softmax}} \mathbf{p}_t \xrightarrow[\not\longleftarrow]{\text{argmax}} y_t \xrightarrow[\not\longleftarrow]{\mathbf{W}[y_t]} \mathbf{u}_t \ , \tag{2.7}$$

where $\mathbf{f}_t$ is the unnormalised probability distribution over all possible words in the vocabulary in step t of the generation, $\mathbf{p}_t$ the softmax-normalised version of $\mathbf{f}_t$, y_t the argument of the maxima of $\mathbf{p}_t$, and $\mathbf{u}_t$ is the word vector corresponding to token y_t. Note that this notation is the same as the one used to present the GAN framework in Section 2.3. Green arrows indicate that an operation is differentiable, whereas red arrows that it is not. Thus, $\mathbf{u}_t$ is not differentiable with respect to the generator's parameters, and therefore neither is the discriminator's output (because it is a function of $\mathbf{u_t}$), which makes the optimisation procedure not straightforward.

2.4.2 | A DIFFERENTIABLE GAN THROUGH THE TOP-K SOFTMAX

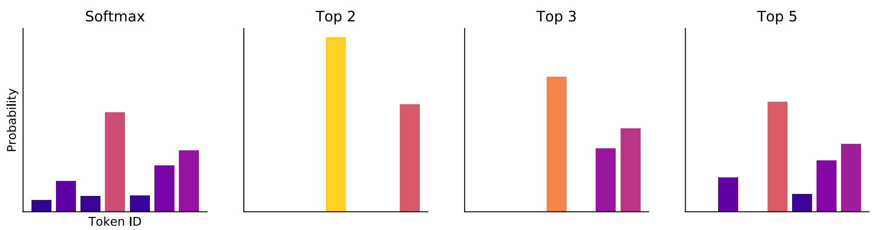


Figure 2.5.: On the left, a graphical example of the softmax normalisation of a $\mathbf{f}_t$ distribution. The rest of the plots show the top-K softmax normalisations of $\mathbf{f}_t$ for different values of k.

We present a strategy to circumvent this differentiability issue and allow the adversarial optimisation of the generator. More precisely, we propose a computation path that approximates $\mathbf{u}_t$, the word vector according to the most probable token at a generation step t, in a fully differentiable manner. This allows the generator to be trained with very convenient gradient-based methods. The idea behind this path is to generate a word vector $\tilde{\mathbf{u}}_t$, hopefully similar to $\mathbf{u}_t$, as a weighted average of the word vectors of the k most probable words according to $\mathbf{f}_t$. $k \geq 2$ is an integer parameter of the transformation. In short, the differentiable computation path is as follows:

$$\mathbf{f}_t \xrightarrow[\longleftarrow]{\text{top-K}} \mathbf{k}_t, \tilde{\mathbf{f}}_t \xrightarrow[\longleftarrow]{\text{softmax}} \mathbf{k}_t, \tilde{\mathbf{p}}_t \xrightarrow[\longleftarrow]{\sum_i \tilde{\mathbf{p}}_t[i] \cdot \mathbf{W}[\mathbf{k}_t[i]]} \tilde{\mathbf{u}}_t \tag{2.8}$$

The first operation in Equation 2.8 performs a selection of the *top-K* elements in $\mathbf{f}_t$. It outputs $\mathbf{k}_t$ and $\tilde{\mathbf{f}}_t$. $\mathbf{k}_t$ are the indices corresponding to the k elements in $\mathbf{f}_t$ with the highest values, and $\tilde{\mathbf{f}}_t$ are those values. In other words, $\mathbf{k}_t$ represents the k most probable words, whereas $\tilde{\mathbf{f}}_t$ their unnormalised probabilities. The second operation is just a softmax normalisation of these k probabilities. It converts $\tilde{\mathbf{f}}_t$ into $\tilde{\mathbf{p}}_t$. At this point, the probabilities corresponding to all the tokens in the vocabulary are zero (they are not in $\tilde{\mathbf{p}}_t$), except for the k most probable tokens, which probabilities sum up to 1. See Figure 2.5 for a graphical example of the top-K softmax. Finally, the approximated word vector that will be fed to the discriminator's RNN is computed as the weighted average of the word vectors corresponding to tokens $\mathbf{k}_t$, where the weights are the probabilities $\tilde{\mathbf{p}}_t$:

$$\tilde{\mathbf{u}}_t = \sum_{i=1}^{k} \tilde{\mathbf{p}}_t[i] \cdot \mathbf{W}[\mathbf{k}_t[i]] \tag{2.9}$$

Thus, each element in $\tilde{\mathbf{u}}_t$ is differentiable with respect to each probability $\tilde{\mathbf{p}}_t$, and the partial derivative is just the associated word vector value:

$$\frac{\partial \tilde{\mathbf{u}}_t[j]}{\partial \tilde{\mathbf{p}}_t[i]} = \mathbf{W}[\mathbf{k}_t[i]][j] \tag{2.10}$$

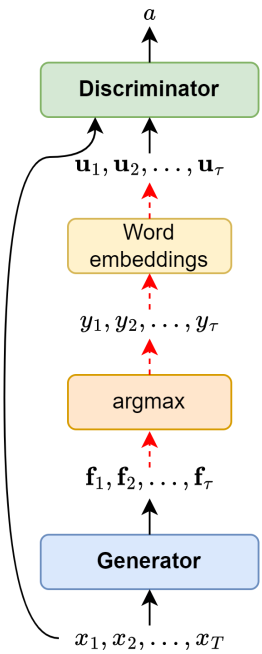
where j is an arbitrary index from 1 to the size of the word vectors. In the same manner, $\tilde{\mathbf{p}}_t[i]$ is differentiable with respect to the top-K elements of $\mathbf{f}_t$, since the transformation (a softmax) is differentiable. The computation path is summarised and compared to the non-differentiable baseline in Figure 2.6.

Our results in Section 2.7.1 show that $\tilde{\mathbf{u}}_t$ is a good approximation of $\mathbf{u}_t$, especially when k is small. In fact, $\mathbf{u}_t$ is the nearest neighbour of $\tilde{\mathbf{u}}_t$ the 98% of the times with $k = 2$. Not only that, we also show that the proposed computation path serves its purpose of allowing fruitful adversarial learning of sequence-to-sequence neural dialogue models, as models trained with this methodology generate more diverse responses (Section 2.7).

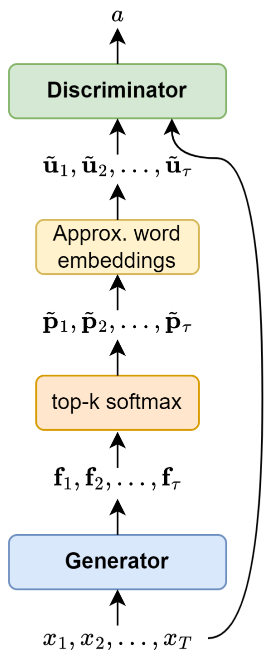
2.5 | Batch-level GANs

2.5.1 | Issues of response-level discriminators

The discriminator explained in Section 2.3.1.2 and most of the discriminators found in the literature work at the response-level, i.e. they evaluate how adequate a response is given the dialogue context. Although this approach has been shown to be valid and useful to build more diverse open-domain dialogue models, it still presents some drawbacks. We have noticed that, sometimes,



(a) Baseline.



(b) Proposal.

Figure 2.6.: Comparison of (a) the baseline non-differentiable computation path for obtaining the word vectors corresponding to the generator's output, and (b) our proposed differentiable path based on the top-K softmax.

generators are able to minimise the discriminator's output by generating only a handful of slightly long and complex sentences, almost regardless of the input. One such response we have found in our experiments is: *you have no choice but to leave him, and you will never forgive him for that, and you will never forgive me.* This effect only lasts a few iterations, until the discriminator is trained to recognise those sentences as not human. However, it often happens again in other stages of the training, with different responses, which results in a more unstable and less effective learning process than desired, as we show in Section 2.7.

This issue is related to the discriminator evaluating only one response at a time. This implies that it has no way to recognise whether the generator is generating some sentence many times (in the same training stage). It can only analyse if a response makes sense or not given the dialogue history. The generator can take advantage of it and learn to produce long and complex, but general sentences which are only slightly coherent with the dialogue history. This sporadic effect is similar to the one observed after a common MLE training, but with longer sentences rather than very short ones.

2.5.2 | Batch-level discriminator

We propose to extend the idea of response-level discriminators to the batch level (López Zorrilla et al., 2023). Our proposed batch-level discriminators combine the response-level predictions of the previously explained baseline discriminator (Section 2.3.1.2) with batch-level predictions that provide a bigger picture of the behaviour of the generator (or the nature of the real data distribution). In other words, while response-level discriminators only aim at answering the question of "*how good is this response given this previous turn?*", batch-level discriminators also tackle the question of "*how does this set of context/response pairs look like?*". In this way, the generator should have more difficulties in fooling the discriminator with long and complex but similar responses. Intuitively, the batch-level discriminator could easily see that many responses in the input batch are complex but similar, and should identify that batch as generated or non-human. We show that using such discriminators improves the variability in the responses of the generator and stabilises the training.

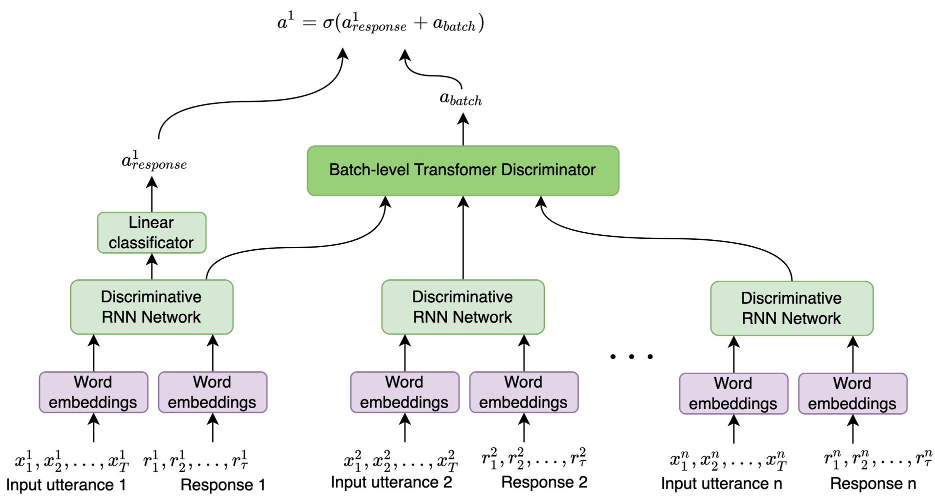


Figure 2.7.: Diagram of the proposed batch-level discriminator.

Figure 2.7 shows a diagram of the proposed architecture for the batch-level discriminator. All the response-level discriminators (the Discriminative RNN in the figure) are the same network, i.e. they share the same parameters. The response-level contribution to the output (denoted as $a^1_{response}$ in the diagram) is computed in the same way as in the baseline discriminator (Section 2.3.1.2). Regarding the batch-level contribution a_{batch}, it is computed from the representations of a set of n dialogue context-response pairs. In our experiments, we found that $n = 8$ is a good enough value to produce interesting results. For simplicity and efficiency, these representations are obtained with the response-level discriminator; they are the output of the MLP in the discriminator, see Figure 2.2

for more details. The sentence-level representations are processed by a standard Transformer encoder (Vaswani et al., 2017) without position embeddings so that its output is not affected by the order of the sequence of representations. This cannot be done easily with RNNs, and therefore a Transformer network is proposed to be used instead. The Transformer encoder produces n output vectors, one per input, which are averaged out. Lastly, a linear layer is used to compute a_{batch}, which is added to the response-level contribution; and the sigmoid function is applied to this sum to provide the discriminator's output. We would like to note that, within each batch of n samples the response-level contribution is different for every sample, whereas the batch-level contribution is the same.

2.6 | EXPERIMENTAL SETUP AND TRAINING DETAILS

We carry out three sets of experiments to validate and analyse the proposed methodologies. First, we analyse the quality of the approximated word vectors presented in Section 2.4. Second, we compare the MLE baseline and the two GAN architectures (response-level and batch-level) in terms of: the variability of the generated responses, the similarity with (multiple) ground truth references (explained next in Section 2.6.4), and also regarding the accuracy of the discriminators. Last, we provide the results of a preliminary human evaluation and some examples of sentences generated by the different models.

In this section, we provide all the details of those experiments. We describe the corpus and its preprocessing in Section 2.6.1, the details of the choices for the neural network and the optimisation parameters in Sections 2.6.2 and 2.6.3 respectively, and a heuristic to evaluate the performance of the generators as well as to stabilise the training procedure in Section 2.6.4.

2.6.1 | CORPUS AND PREPROCESSING

The experiments were carried out with the English version of the OpenSubtitles2018 corpus (Lison and Tiedemann, 2016; Lison et al., 2019), which is composed of around 400M utterances from movie subtitles. As proposed in (Vinyals and Le, 2015), since the turns are not clearly indicated, we treat each utterance as the desired output for the previous one. However, we do not consider that a utterance follows the previous one when the time difference between them is higher than three seconds. After this process, 241M input-output pairs were formed.

As for the text preprocessing, we removed uncommon and non-informative symbols and characters. We employed a Byte-Pair Encoding (BPE) tokeniser (Sennrich et al., 2016) to tokenise the clean text. This way, the most common words are represented as a single token while the less frequent ones are broken down into several subword tokens. The selected size for the vocabulary was 30000. We pretrained 300-dimensional word vectors of these tokens (subwords) in the corpus, with FastText (Bojanowski et al., 2016), and kept optimising them throughout the training. We also tried randomly initialising the word vectors, but experienced a slower learning.

Last, we would like to note that we did not split the corpus into any train/test partition, because the amount of training examples we process during training is significantly lower than examples in the corpus (241M examples in the corpus vs. 77M examples sampled once during training). Thus, every example processed by any component of the GAN is *new* during training; there are no repeated examples. All the metrics we report are computed with training examples because no overfitting should be possible.

2.6.2 | Details of the neural network architectures

Let us now give details about the architecture of the sequence-to-sequence generator. The deep bidirectional RNN encoder is made of two LSTM networks (one per direction) of 4 layers, 512 cells each. On the other hand, the decoder's LSTM has 4 layers of 1028 cells. The MLP that converts $\mathbf{o}_t$ and $\mathbf{c}_t$ into $\tilde{\mathbf{o}}_t$ (see Section 2.3.1.1 for more details) has one *leaky*-ReLU layer. The size of $\tilde{\mathbf{o}}_t$ is 500. The MLP that computes the attention score has two layers. The first one is a 250-sized hyperbolic tangent layer, and the second is a linear output layer that computes the scalar score. A dropout probability of 0.1 is used after each RNN layer during training.

Regarding the discriminator, the deep bidirectional encoders in charge of processing the input and response share the same architecture: they are bidirectional LSTM networks of 3 layers, 512 cells each, similar to the generator's encoder. The last output vector is then fed to a MLP of two layers: a *leaky*-ReLU layer of size 256 followed by a single sigmoidal unit. The transformer of the batch-level discriminator takes as input the output of the first MLP layer. Thus, the size of the transformer layers is 256 too. It is made of two layers, with four heads each. A dropout probability of 0.1 is used after each RNN and transformer layer during training.

2.6.3 | GAN Optimisation hyper-parameters

The most promising hyper-parameters we have found for the training procedure presented in Algorithm 1 are summarised next. We provide details about the pretraining first, and then about the adversarial training loop.

We pretrained the generator during 200.000 iterations with a fixed learning rate of 0.001. AdamW (Loshchilov and Hutter, 2017) was used in all optimisation processes. We sampled 12.800 responses from that generator (256 every 4000 iterations). The discriminator was then pretrained during 2000 iterations, with the same learning rate. All the batches fed to the discriminator were balanced: there was a human example per each generator's example. Human and generator's example were uncorrelated; they did not share the input. The batches for the batch-level discriminator were split into subsets of 8 samples to compute the batch-level response evaluations. All instances in the subsets belong to the same category (corpus or generated).

The adversarial learning loop was run 250 times. The initial learning rate was 0.001 with a decaying factor of 0.996 when training the discriminator and the generator with the MLE criteria. It was ten times smaller when training the generator to minimise the output of the discriminator. Inside the loop, every generator optimisation step was run during 200 iterations, including both the MLE and adversarial learning optimisation. After each of these steps, 5000 input-response pairs were sampled from the generator, and the discriminator was trained during 100 iterations. The chosen value for the k parameter of the top-K softmax was 2.

It is worth mentioning that we did not vary the architectural hyper-parameters much during our experiments. They are similar to many other sequence-to-sequence networks in the literature. On the other hand, we noticed that selecting good and stable training hyper-parameters is challenging. This requires a deeper and more specific research that we leave for future work.

2.6.4 | Response evaluation and filtering

We measure the semantic adequacy of the generated responses via LaBSE sentence embedding similarity (Feng et al., 2022). In particular, this is defined as the cosine product between the sentence embedding of the produced and target responses. Additionally, we also try to take into account the fact that many responses can be valid given the same dialogue history, even if they are not semantically similar. In order to find other valid responses, we first search for similar inputs in the corpus, using LaBSE embeddings too. We consider the responses to these similar inputs as valid responses to the original input of the generator, and compare its output to them. We consider the maximum value among all the comparisons as the measure of the quality of the generated response. We use a threshold of 0.8 to find similar inputs. We found that this value represents a good balance for keeping the quality of the comparisons, while being flexible enough to find a number of similar inputs. We report the percentage of responses whose semantic similarity with the best reference is higher than the threshold (0.8). Figure 2.8 illustrates the reference search in a hypothetical two-dimensional sentence embedding space. The green points correspond to the original input and the ground truth. Points in purple and pink represent pairs close in the input space, and whose outputs could be valid responses given the original input. The sentences are close in the input space, but could be very far in the output space.

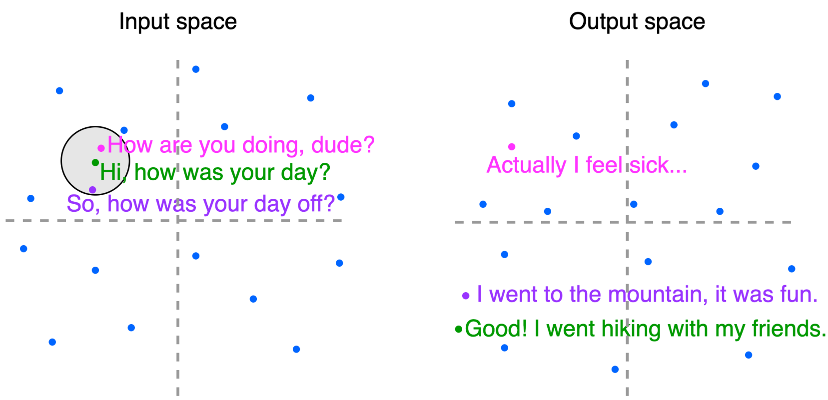


Figure 2.8.: Illustration of the search for additional output references in the corpus given an input.

We take advantage of this metric to develop a heuristic to further improve the training process of GANs. We propose to slightly modify a step on the main training loop of GANs, Section 2.3.2.3. As aforementioned, in order to train the discriminator, a set of generator responses are sampled every time it is updated, and these responses will later be fed into the discriminator as non-human– and therefore *bad* or non convenient– responses. However, there are cases, especially after the generator has been trained for a while, where the responses it produces might be completely acceptable. Using this input-output pairs as negative examples can therefore deteriorate and slow down the training. We propose to filter responses with high scores in the semantic similarity metric (>0.8) from the corpus of generator's responses used to train the discriminator.

2.7 | Results

Let us now present the results of the three sets of experiments aforementioned in Section 2.6.

2.7.1 | Quality of the approximated word vectors

In order to measure the quality of the approximated word vectors, we computed which was the closest word vector to each approximated one according to the euclidean distance, for different values of k. To this end, we selected 1000 random input sentences from the corpus, fed them to a trained generator, and analysed the probability distribution of each of the generated tokens.

With $k = 2$, the closest word vector was the correct one–i.e., the one with the highest probability– 98% of the times if we consider all the produced tokens, and 97% if we do not consider repetitions. These two percentages decrease to 83%/69% respectively with $k = 3$, and to 74%/60% with $k = 4$. Figure 2.9 shows this statistic for more values of k. We therefore conclude that the proposed method to make the output of the discriminator differentiable with respect to the generator's parameters is appropriate, especially with $k = 2$, the value used in our experiments.

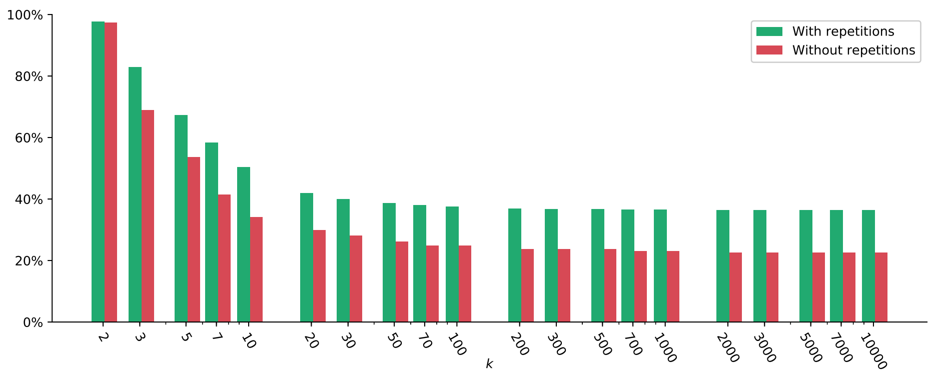


Figure 2.9.: Frequency of the most likely word vector being the nearest neighbour of the approximated word vector produced by the top-K softmax, for different values of k.

2.7.2 | Response variety

The main goal of dialogue GANs is to increase the variety of responses of sequence-to-sequence generators. We measure the variety throughout the training process (both the MLE pretraining and the two GAN optimisation processes) with the *distinct-1* (Dist-1), *distinct-2* (Dist-2), *distinct-3* (Dist-3) and *distinct-sentences* (Dist-S) metrics, proposed in Li et al. (2016a) and still in use currently (Luo and Chien, 2021; Li et al., 2022b). Dist-1, Dist-2 and Dist-3 are the number of distinct unigrams, bigrams and three-grams (at the token level) in generated responses. The values are normalised by total number of generated tokens to avoid favouring long sentences. On the other hand, Dist-S is the ratio of different responses. We computed the metrics with batches of 256 random inputs, and averaged them over 8 independent training runs.

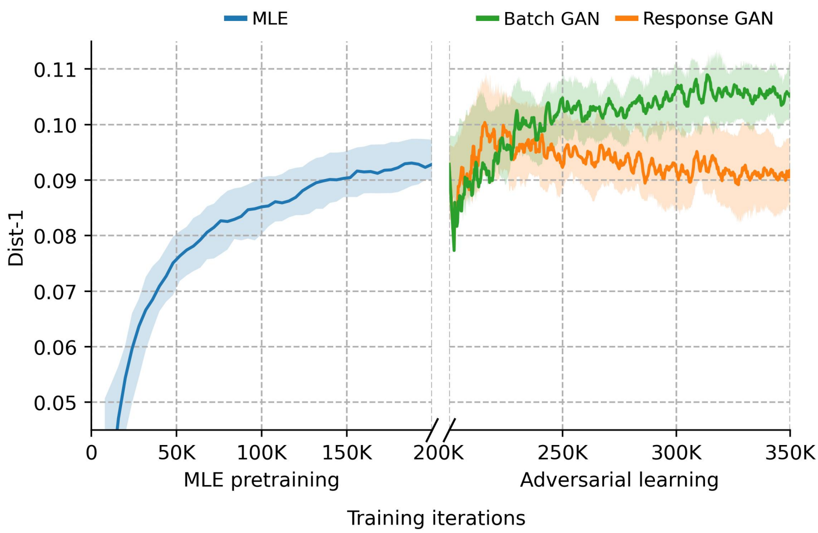


Figure 2.10.: Evolution of the Dist-1 metric throughout MLE pretraining and adversarial learning.

The results in terms of Dist-1, Dist-2, Dist-3 and Dist-S are shown in Figures 2.10, 2.11, 2.12 and 2.13, respectively. The x-axis of these plots is broken in two. The first half (iterations from 0 to 200K) corresponds to the MLE pretraining, while the second half (iterations 200K to 350K) to the adversarial learning, with the two proposed GAN models. In addition to the average values after 8 training runs, we also illustrate the first and second tertiles (the 33th and 67th percentiles) as shaded areas, to provide information about the statistical variability of the results.

The four plots follow a similar pattern. During the first 50K iterations of the MLE pretraining stage, the variety of the responses increases highly. Then it stabilises and the improvement is less notorious until iteration 200K, when the pretraining is complete. The variety sharply drops right after the adversarial learning begins, with both sentence-level and batch-level GANs. But then the GANs rapidly stabilise and start producing more and more varied responses. It

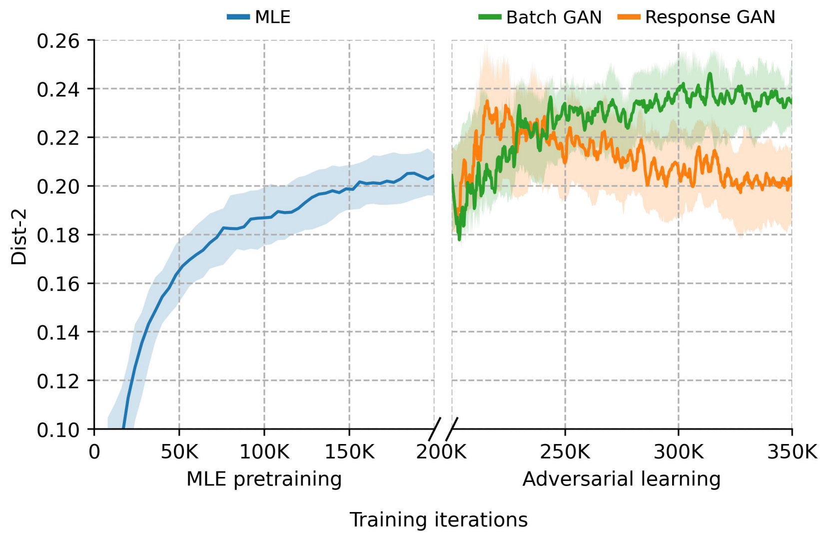


Figure 2.11.: Evolution of the Dist-2 metric throughout MLE pretraining and adversarial learning.

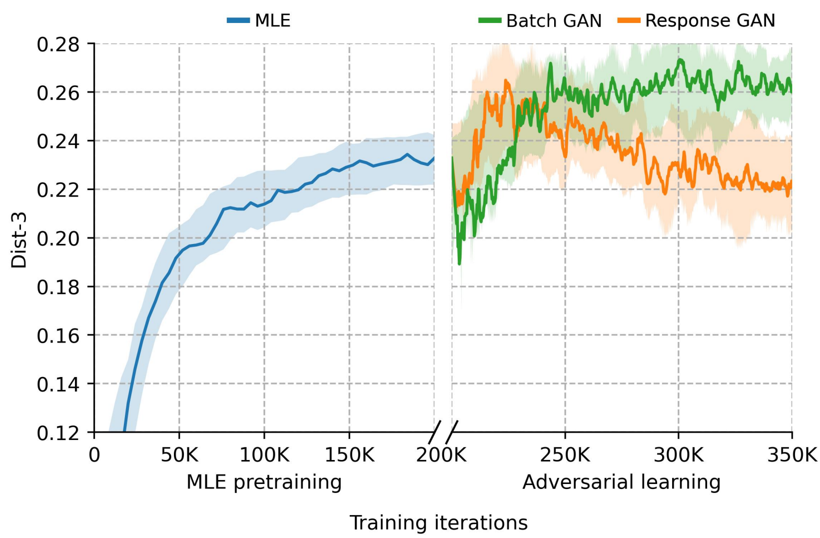


Figure 2.12.: Evolution of the Dist-3 metric throughout the MLE pretraining and adversarial learning.

is interesting that the improvement is much higher in the response-level GANs as opposed to the batch-level one. This is due to the response-level discriminator being much simpler: it can be trained faster, but because of its limitations (see Section 2.5.1) its peak is lower and it even diverges (the results deteriorate with time). That is, after around 25K adversarial learning iterations (225K itera-

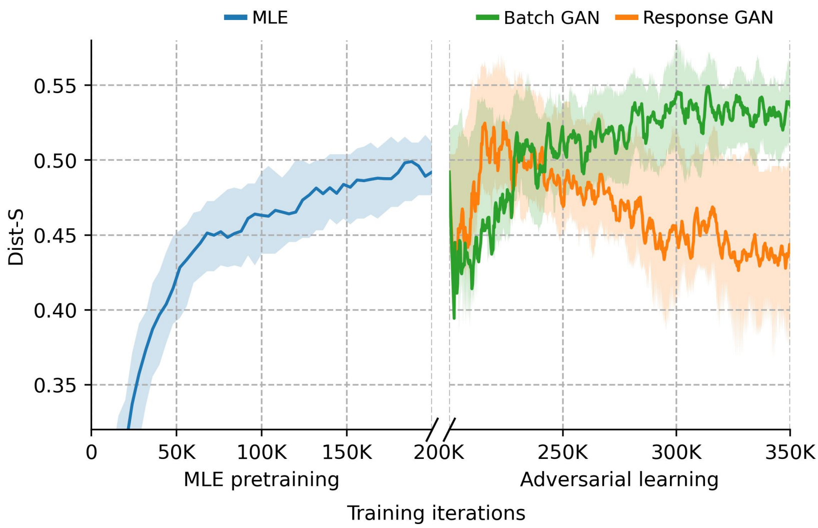


Figure 2.13.: Evolution of the Dist-S metric throughout the MLE pretraining and adversarial learning.

tions in total), its performance starts to decrease and it ends up with a response variety similar to or lower than the MLE baseline, with a greater variance. On the other hand, the batch-level discriminator processes more information and thus it takes longer to train the GAN. However, it keeps improving throughout the 150K adversarial learning iterations, and its peak is higher than the MLE baseline and the response-level GAN.

To sum up, both GAN models provide more varied responses than the MLE baseline according to the four implemented metrics. This improvement is higher and the training is more stable with batch-level GAN than with the response-level GAN. This validates the top-K softmax approach to build fully differentiable GANs as well as our proposed batch-level discriminator.

2.7.3 | Performance – percentage of filtered responses

As aforementioned, we also report the percentage of good responses not included as negative examples for the discriminator according to the methodology presented in Section 2.6.4. The evolution of this metric is shown in Figure 2.14.

This result further validates our proposal, especially the batch-level GAN. Not only does it lead to more varied responses than the MLE baseline, but it does so without hurting the quality of the responses. In fact, these are also se-

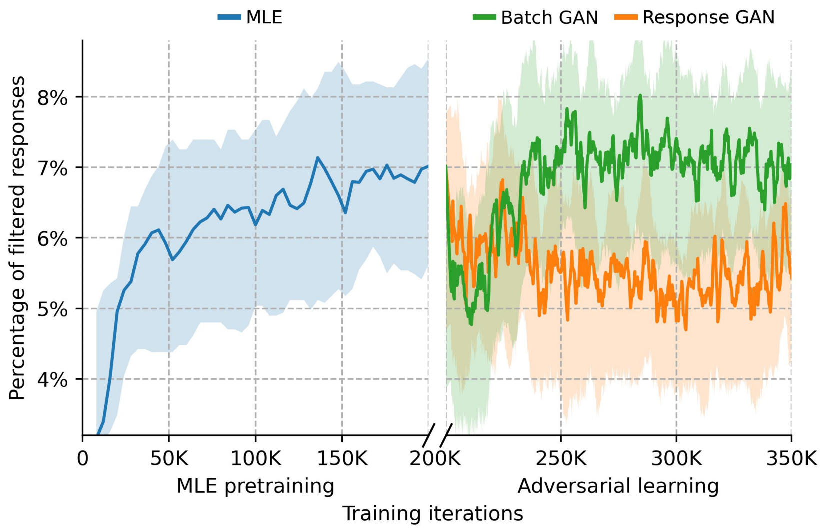


Figure 2.14.: Evolution of the percentage of good responses not included as negative examples for the discriminator throughout the MLE pretraining and adversarial learning.

mantically more adequate, even though only slightly, according to this metric. On the other hand, the response-level GAN performs slightly worse than the MLE baseline. Once again, this might be due to the intrinsic limitations of such GANs: the generator can easily minimise the discriminator's output by producing long and complex sentences, but these are not necessarily semantically appropriate given the dialogue context.

2.7.4 | Discriminator accuracy

The last automatic metric we tracked was the discriminators' accuracy. This is shown in Figure 2.15. The accuracy is already fairly high from the beginning because the discriminators were pretrained with responses from the MLE baseline before starting to compute the accuracy. The accuracy was computed with instances unseen during training.

This plot is aligned with the learning curves shown previously. The response-level discriminator learns faster, i.e. it is able to obtain a higher accuracy in the first 50K to 100K iterations (250K to 300K iterations if we start counting from the beginning of the training process, as shown in the plot). Nonetheless, the accuracy improves only slightly after the pretraining. The batch-level discriminator, as expected, takes longer to train, but it ends with a higher accuracy than

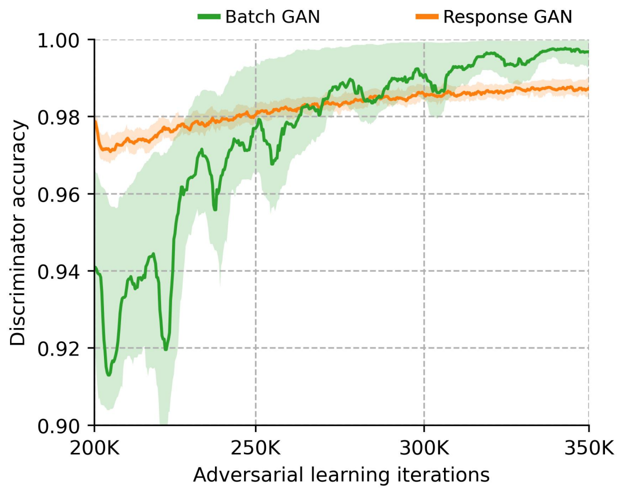


Figure 2.15.: Evolution of the accuracy of the discriminators throughout adversarial learning.

the response-level one. This ratifies once again that this kind of discriminator has a greater potential.

2.7.5 | Preliminary human evaluation

Let us show a preliminary human comparison between the MLE baseline dialogue model and the response-level GAN. 10 human evaluators, mostly Ph.D. students or postdoctoral researchers on Electronics or Computer Science, took part in this experiment. Their average age was 26.9, and the gender distribution was 30% females, 70% males.

The evaluators were asked to to interact freely with the two models for a few minutes, first with the MLE baseline, and then with response-level GAN. These interactions were text-based, and a simple app was developed to this end. The judges were also told to carry out similar dialogues with the two system to avoid any bias in this regard. On average, the resulting dialogues were 25 exchanges long. Then, they filled a short questionnaire made of three questions: “Which model offered a bigger variety of responses?”, “Which model’s responses were more coherent with your turns/questions?”, and “Which model was more informative?”.

7 out of the 10 evaluators opined that the final system was more variate and informative, and there was a draw (5-5) in terms of coherence. This result is aligned with the conclusions drawn from the automatic metrics.

2.7.6 | Generation examples

Finally, we would like to show some responses that showcase the aforementioned discussed behaviour of the different models. Table 2.1 contains responses of four instances of the MLE baseline, response-level GAN and batch-level GAN to the same four input utterances. Each response has been generated by independent models; we use four of the eight trained models to obtain the automatic metric results to this end. We have picked responses generated after the training finished for the MLE baseline and for the batch-level GAN. As for the response-level GAN, the responses correspond to earlier stages of training, before the GAN slightly diverges, according to the automatic metrics.

The difference in the variability of the responses is quite noticeable in these examples. The MLE baseline produces generic or dull responses much more frequently than both GAN models, which tend to produce more complex and informative sentences. Furthermore, the MLE responses to the same input are more similar. This is especially visible in the case of the first input *Yesterday I saw you in the mountains*, where the four instances produced extremely similar outputs. We would also like to mention the phenomenon of the occasionally repetitive long and complex outputs of the response-level GAN. The third instance (third row) responded with *What do you want me to say, huh?* to the first and second inputs. We have noticed that this kind of behaviour occurs at many stages during training, as previously discussed. Batch-level GANs reduce this issue significantly.

2.8 | Conclusions

We have presented a novel methodology to allow text-generating models to be trained in the GAN framework, the top-k softmax, as well as batch-level discriminators that address some of the issues of the commonly used response-level GANs. We have shown that good approximations of the word vector corresponding to each of the generated tokens can be obtained with the top-k softmax. Furthermore, GAN models trained using the top-k softmax provide more varied responses than the MLE baseline in our experiments. As for the future, it would be good to directly compare this methodology with other alternatives to circumvent the differentiability problem, such as RL or the Gumble-softmax approach (see Section 2.2.2), to quantitatively analyse its pros and cons. In any way, the obtained improvement is higher and the training is more stable with batch-level GAN than with the response-level GAN. This validates the top-k softmax approach to build fully differentiable GANs as well as our proposed batch-level discriminators.

Table 2.1.: Responses of four instances of the MLE baseline and the two GAN models to the same inputs.

Input utterance: *Yesterday I saw you in the mountains.*		
MLE baseline	**Response-level GAN**	**Batch-level GAN**
I was in the woods.	You were in the mountains?	You saw me, didn't you?
I was in the mountains.	You were the one who was there.	You know, you can't see me.
I was in the woods.	What do you want me to say, huh?	I was in the woods.
I saw you in the woods.	I saw you in the mountains.	You know, I've been trying to call you all day long.

Input utterance: *Could you repeat that, please?*		
MLE baseline	**Response-level GAN**	**Batch-level GAN**
I'm sorry.	I'm sorry I didn't get a chance to say goodbye.	You're a man of respect for the truth.
I'm not sure.	It's a good idea.	I'm not a doctor anymore, but I'm a writer.
I'm not going to let you do this.	What do you want me to say, huh?	And I'm sorry about your mother's death.
I'm not sure.	I'm sorry about what happened last night.	You can do better than that, sir.

Input utterance: *Tell me, what do you like?*		
MLE baseline	**Response-level GAN**	**Batch-level GAN**
I like to think you're a good man.	I like the way you look at me.	I like the way you look at them.
I don't know.	I don't know what I like about you, but I like it.	What do you like?
I like to eat.	I don't know, man, I don't think you understand, man, but...	I like to be alone.
I like the way you look at me.	I like your hair.	I like to be a little more comfortable.

Input utterance: *Could you come with us to that place?*		
MLE baseline	**Response-level GAN**	**Batch-level GAN**
I'm not sure.	I'll see if I'll have a drink with you.	What are we going to do?
I'll be right there.	Yes, of course.	I'm not going anywhere, dad.
I'm going to go.	I don't think you understand how many people you have.	I'll be right here.
I'm not sure.	I don't want to see that.	I don't want to go.

Last, we would like to conclude by noting that, even if GANs are still widely used for applications such as (conditional) image generation (Li et al., 2022a; Hu et al., 2021; Sauer et al., 2021; Yu et al., 2021), their popularity for dialogue-related tasks has dropped recently. This is probably due to more attractive and simpler alternatives having been proposed recently, such as transfer learning from large pretrained LMs. However, we believe that our research (especially batch-level discriminators) is still relevant because many other NLP systems, such as GPT-2-based dialogue managers (see Chapter 4, Section 4.4.4) or BERT-based question answering systems (Devlin et al., 2019), use discriminators that could potentially benefit from our proposals.

Chapter 3

Contributions to the EMPATHIC project

3.1 | Introduction

Let us introduce the European H2020 Project EMPATHIC[1] (López Zorrilla et al., 2018; Torres et al., 2019; Brinkschulte et al., 2021), which is the framework adopted in this chapter and the next one. It was aimed to research, innovate, explore and validate new interaction paradigms and platforms for future generations of personalised Virtual Coaches (VCs) to improve independent healthy-life-years of the elderly. EMPATHIC presented a multidisciplinary consortium made of research groups from European universities, health-related institutions and technological companies. Namely, experts from the University of the Basque Country UPV/EHU, University of Barcelona, Institut Mines-Télécom, Università degli Studi della Campania Luigi Vanvitelli, Oslo University Hospital, OSATEK, e-Seniors Association, Tunstall Healthcare, Intelligent Voice and Acapela Group took part in the project.

The technical development of the EMPATHIC VC consisted of four main stages (depicted in Figure 3.1): a data acquisition stage (Stage I) where human-computer dialogues were acquired through a Wizard of Oz (WoZ) technique; the labelling, annotation, and structuring of these data to form the EMPATHIC corpus (Stage II); the actual development of the modules for the VC prototype (Stage III); and its validation with the target population (Stage IV). All data acquisition and validation experiments were carried out in Spain, France and Norway, leading to a multilingual and multicultural corpus and prototype. In this chapter, we describe these four stages, focusing on our contributions, which are

[1] http://www.empathic-project.eu/

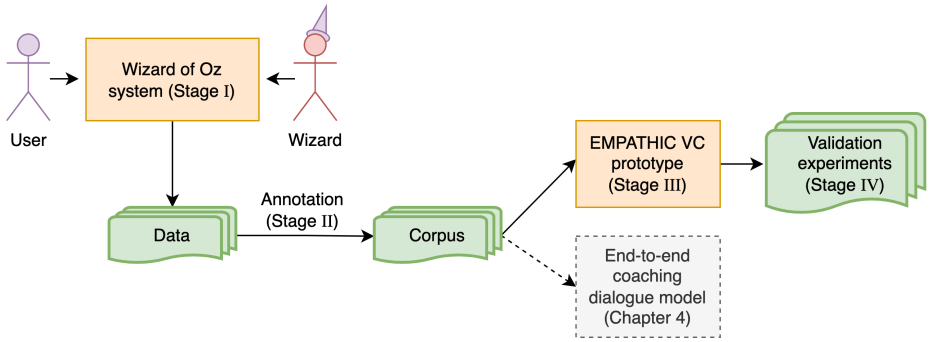


Figure 3.1.: Schema of the organisation of this chapter.

a result of collaborative work with colleagues of the project. Our main contributions are summarised next:

I. In the data acquisition process through the WoZ technique:
   - Design of the scenarios used to simulate the system behaviour.

II. In the labelling and annotation of the acquired corpus:
   - Design of the semantic labelling taxonomy and annotation procedure.
   - Analysis of the results of the semantic labelling.

III. In the development of the VC prototype:
   - Design of the dialogue strategy, specified through dialogue trees.
   - Design of the NLG post-processing.

IV. Additionally, and aside from the collaboration with members of the consortium:
   - Analysis of the interactions of the VC prototype with the target population.

Mind that the EMPATHIC VC is not fully data-driven, which is a difference to the dialogue systems presented in the rest of the chapters. The EMPATHIC corpus was also used to develop an end-to-end dialogue model, which is presented in Chapter 4, as represented in the schema in Figure 3.1.

This chapter is organised as follows. Section 3.2 provides an overview of the target dialogues of EMPATHIC–coaching sessions. Then, Sections 3.3, 3.4, 3.5 and 3.6 present the contributions of the four stages in Figure 3.1. In Section 3.3 the designed WoZ scenarios corresponding to Stage I are described. Section

3.4 presents the EMPATHIC corpus: big numbers first, and then a more thorough description of the semantic labelling taxonomy, procedure and results (i.e., Stage II). Section 3.5 includes an overview of the EMPATHIC VC developed in Stage III, then a more in-depth analysis of our contributions to its DM and NLG. Section 3.6 contains the analysis of the interactions between the target population of EMPATHIC and the VC prototype (Stage IV). Last, Section 3.7 presents some concluding remarks.

3.2 | GROW coaching dialogues

The main purpose of acquiring a corpus of dialogues within the EMPATHIC project was to generate high-quality data to train and/or design the modules that would make up the VC. This VC was intended to carry out coaching dialogues in Spanish, French and Norwegian; and therefore the acquired corpus should also contain this kind of conversations. According to the International Coaching Community, the essence of coaching is "*to help a person change in the way they wish and helping them go in the direction they want to go*[2]". It is important to remark that when coaching someone it is the coachee who has the answers, not the coach. Thus, the goal of the coach is to make the coachee reflect and help them find these answers, not to just tell them what they should do.

Experts use several coaching techniques to try to make the coachees[3] realise how they could improve their habits or reach their goals. For EMPATHIC, the GROW coaching model (Whitmore, 1992) was selected, since it was suggested by experts in the area, on the grounds that the GROW dialogues are much more structured than in other coaching methodologies (Justo et al., 2020). Thus, the GROW model should be very suitable for both the data acquisition process and the dialogue modelling for the automatic prototype.

In short, a full GROW coaching session consists of four main phases: Goals or objectives, Reality, Options, and Will or action plan. During the first phase, the agent aims to get explicit objectives from the user, for instance, increasing the daily fruit intake or reducing the amount of sugar in their diet. During the Reality phase, the coach and the user analyse the user's personal context and try to detect potential obstacles that currently hinder the achievement of the previously established objectives. As for the Options phase, the goal is to generate and find options and tools that the user may use to face the aforementioned obstacles and achieve their objectives. In the last phase, the objective is to specify an action plan for the user to carry out in order to advance towards their objective.

[2] https://internationalcoachingcommunity.com/what-is-coaching/

[3] We often refer to the coachee as the *user*, because they *use* the SDS.

For instance, Table 3.1 shows a representative fragment from a fictional GROW session corresponding to the Reality and Options phases, where the coach provokes a clear reflection on the user. This example is part of a number of examples written by an expert coach to assist in the development of EMPATHIC (Sayas, 2018a,b,c).

Table 3.1.: A fragment of a coaching session extracted from our corpus.

Agent:	*So, Pablo, have you ever eaten consistently 2 or 3 pieces of fruit?*
User:	*When my wife was healthier, she used to take care of buying the fruit. Thus it was easier for me to eat it.*
Agent:	*And at any other time of your life?*
User:	*Long time ago, when I lived alone, I used to take care of buying the fruit myself, and I ate it more frequently.*
Agent:	*What does this information suggest to you about your objective?*
User:	*Well... That it's something that basically depends on me.*
Agent:	*So, can you see anything you could do to get closer to your objective?*
User:	*Uhmm... I should start thinking how I am going to organise to buy the fruit.*

Each phase of the GROW model is also characterised by the type of questions employed by the coach, as depicted in Figure 3.2. Goal Set Questions (GSQs) try to identify and clarify the goal and commitment to it. Motivational Questions (MQs) are also used in the first phase of the conversation, to motivate the user by making them realise the potential benefits of achieving their goal. Reality Questions (RQs) and Obstacle Questions (OQs) are the main types of question for the Reality phase. RQs assess the current situation of the user with respect to the objective, while OQs aim at exploring the user's internal obstacles and blocks limiting their progression. OQs are also used in the Options phase along with Option Generation Questions (OGQs). These focus on making the user aware of the options or possibilities that could be used to get closer to the goal. Last, the Will phase is characterised by Plan Action Questions (PAQs), which are aimed at specifying an action plan to reach, or get closer to the goal. Additionally, the coach can also employ Follow-up Questions (FQs) to check the progress after a coaching session and Warning Questions (WQs) to assess risk situations related to the health status of the user. Besides all the GROW-like questions, non-domain specific responses such as confirmations, backchannels or greetings, might also be used in every phase of the session by the coach.

3.3 | DATA ACQUISITION: WOZ SCENARIOS

After establishing the coaching methodology to be used and modeled in EMPATHIC, the next step was to plan a data acquisition procedure to generate a corpus of human-computer GROW sessions. This corpus would help to train and design many modules of the EMPATHIC prototype, but also to provide a

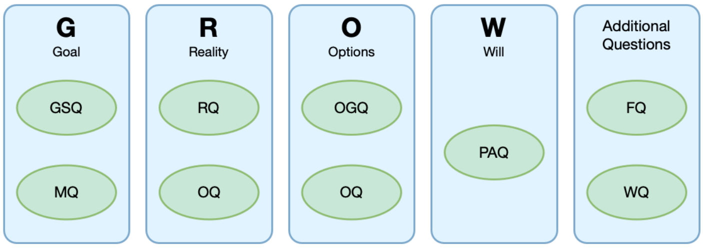


Figure 3.2.: Question types per GROW phase.

better understanding of how independent elderly users would react to virtual and automatic coaching sessions (Esposito et al., 2021).

A WoZ technique (Dahlbäck et al., 1993; Riek, 2012) was used to acquire the data, since it allows to collect conversations similar to the target ones. This method consists in asking participants to interact with a system that they believe to be autonomous, but which is actually controlled by a human expert–the wizard. Thus, the participants should act as they would when facing a real automatic system; they will often talk more slowly or express themselves more concisely, for instance. In EMPATHIC, GROW coaching sessions needed to be implemented in the WoZ trials. While in-person coaching sessions often last between 30 and 60 minutes, the EMPATHIC GROW WoZ sessions were decided to be around 10 minutes long. The reason for this is that longer conversations may be too hard to model, and that participants (healthy elderly) may not be willing to interact with a VC for longer.

Two separate scenarios (dialogue types) were planned for the WoZ sessions. First, we designed an introductory scenario that was used to engage the user and make them feel comfortable in the interaction with the system. In this scenario, the system (controlled by the wizard) presents itself and briefly describes the coaching methodology it will be following. Afterwards, it talks with the user about their hobbies, such as travelling, music and reading, but always with coaching in mind. In fact, the introductory dialogue can already be considered (the beginning of) a coaching session, since the wizard uses a coaching language trying to make the user reflect whenever possible. Second, a (partial) GROW session on nutrition was simulated. The 10 minutes limit was generally enough to complete the first phase of the GROW structure, the Goals. Sometimes, the conversation was very fluid, and more phases were completed. The nutrition topic was selected because it is a key factor in healthy ageing. According to the World Health Organization, "*good nutrition can help to preserve cognitive function, delay care dependency, and reverse frailty*" when ageing[4].

The EMPATHIC WoZ platform was based on the one developed by Schlögl

[4]https://www.who.int/news-room/fact-sheets/detail/ageing-and-health

et al. (2010). The wizard could respond to the user's audio/video using and modifying predefined candidates or composing a response if none of the predefined ones were suitable. Our main contribution in this aspect was to define and arrange the sets of predefined responses, so that health professionals, the wizards, could carry out GROW-like dialogues. First, a set of general, non-domain-specific responses was designed, which could be used in any scenario or phase. A total number of 37 sentences were designed, which can be found in Appendix A. These were always visible to the wizard, as seen in the right-hand side of Figure 3.3.

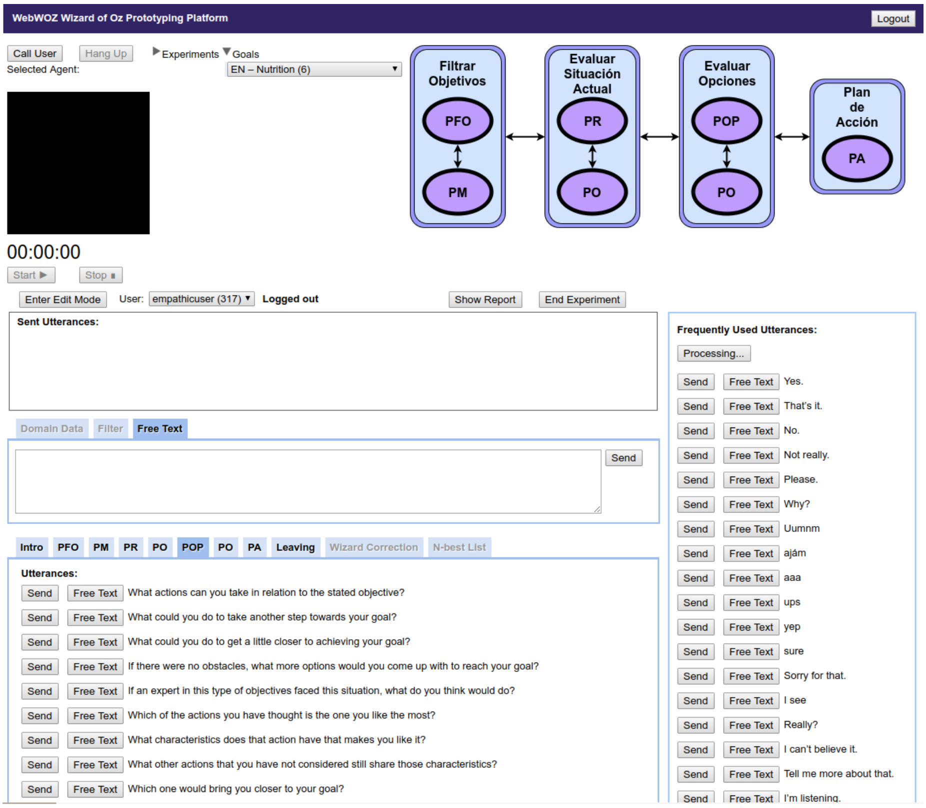


Figure 3.3.: Interface for the wizard in the WoZ trials of EMPATHIC. The abbreviations for the question types are in Spanish, as English was only used as an intermediate language for translating the scenarios.

As for the introductory dialogue, four sets of sentences were designed: an introduction to the project (made of 31 predefined turns), travelling (16), music (17) and goodbye (5). The wizard can select which set of questions to use in the interface so only the more relevant candidates appear on the screen, as shown in the bottom left of Figure 3.3. Using the introductory set of questions, the system presents itself and the project and asks some general questions about the potential hobbies of the user. Then, depending on how the first phase developed, the system and the participant talk about one of the user's hobbies, preferably travelling or music (or both), because the set of predefined questions was only designed for these topics. Last, the system thanks the user for their time and says goodbye with the last set.

Regarding the GROW session about nutrition, the sets of questions match the ones in Figure 3.2 adapted to the nutrition topic. Four greetings were included, 15 GSQs, 14 MQs, 8 RQs, 7 OQs, 10 OGQs, 7 PAQs and 3 goodbye turns. Note that WQs were not included in the platform, because experts recommended against generating health alerts in the system, and FQs are also missing because they only make sense for a second GROW session. The sets of questions for both the introductory and GROW nutrition scenarios can be found in Appendix A.

We instructed six wizards (two in Spain, two in France and two in Norway) to use this platform and these scenarios to carry out GROW dialogues. The wizards were staff of the end-user partners of EMPATHIC, namely OSATEK, e-Seniors Association and Oslo University Hospital. We also showed them how to modify the scenarios, so that they could adapt them throughout the WoZ trials. Thus, the final sets of questions vary slightly between countries.

The first WoZ acquisition process was carried out successfully, and later the second round of experiments, named WoZ+, was prepared. In this case, two dialogues per user were developed too, but the topics differed: the first one was still the GROW session about nutrition, but the second one was a GROW session about physical activity. This time, and after the experience gained in the first round of data acquisition, the wizards were responsible for building the physical activity scenario from scratch.

3.4 | EMPATHIC CORPUS

Let us now summarise the statistics of the acquired dialogues and their annotation (Section 3.4.1), focusing on the semantic labelling, where we contributed the most (Section 3.4.2).

3.4.1 | SUMMARY

The EMPATHIC corpus contains the data gathered in both WoZ and WoZ+ trials. The WoZ data was collected with the scenarios we designed, and it has been fully annotated, including the semantic labelling that we explain next in Section 3.4.2. It is also the basis of the data-driven dialogue model explained in the next chapter. In the case of WoZ+ trials (Section 3.4.1.2), dialogues were not labelled. The EMPATHIC corpus is available at: http://catalog.elra.info/en-us/repository/browse/ELRA-S0414/.

3.4.1.1 | WoZ trials

A total number of 153 participants took part in the WoZ trials. Table 3.2 shows their demographic information (gender and average age) divided by country. We also provide information about their quality of life via WHOQOL-BREF questionnaire (World Health Organization et al., 1996) (in a 0-100 scale), and about their depression level via the Geriatric Depression Scale (GDS) questionnaire (Sheikh and Yesavage, 1986) (in a 0-30 scale). WHOQOL-BREF scores lower than 45 indicate a poor quality of life, values between 45 and 65 a moderate quality of life, and values above 65 a relatively high quality of life (Bani-Issa, 2011). On the other hand, GDS scores lower than 10 are normal, values between 10 and 20 indicate mild depression, and values above 20 correspond to severe depression.

Table 3.2.: Demographic data of the participants of the WoZ experiments.

	Spain	France	Norway
Number of participants	78	44	31
Gender	24 M, 54 F	16 M, 28 F	10 M, 21 F
Average age	69.5	73.5	74.8
Avg. WHOQOL-BREF score	72.1	70.1	69.0
Avg. GDS score	4.8	4.4	5.7

The statistics of this partition are summarised in Table 3.3. The number of videos or audios does not exactly match the number of transcribed dialogues due to some technical issues. Since almost every participant interacted with the WoZ in the first two scenarios, around half of the corpus corresponds to introductory sessions and the other half to GROW sessions on nutrition.

Table 3.3.: WoZ corpus summary (*ann.* stands for different annotators).

	Spain	France	Norway
Video/audio files	142	76	68
Time	23:11:06	11:04:06	8:57:31
Transcribed dialogues	142	68	60
User dialogue act annotation files	142	68	60
System dialogue act annotation files	142	68	60
Emotion annotation from audio files	134×3 ann.	76×2 ann.	60×2 ann.
Emotion annotation (crowd) from audio chunks	4521×5 ann.	-	-
Emotion annotation from video files	134×2 ann.	76×2 ann.	60×2 ann.
Biometry annotation files	134	-	-

The recordings of the sessions were manually transcribed, and the system and user turns were labeled semantically. The users' audio, video and text were also labeled in terms of emotions (Letaifa and Torres, 2021; de Velasco et al., 2022). Besides the data annotation, a number of questionnaires were administered to the participants: before the interaction, the aforementioned WHOQOL-BREF and GDS questionnaires were administered; and the Virtual Agent Acceptability Questionnaire (VAAQ) questionnaire for the acceptance of the automatic system (Esposito et al., 2018), System Usability Questionnaire (SUS) questionnaire for its usability (Brooke, 1996) and a self-annotation of their emotions, after the interaction.

Additionally, the transcriptions of the dialogues were translated into the three target languages to increase the amount of data. English was used as an intermediary language to ease the translation procedure. As a result, the corpus of dialogues is available in English too. Table 3.4 shows a comparison between the original amount of data and the data after the translations.

Table 3.4.: General statistics of the corpus of WoZ (translated) dialogues.

	Original data			Total after translations
	Spanish	French	Norwegian	(same for all languages)
Number of dialogues	142	68	62	272
Number of system turns	4813	1776	1324	7913
System turns per dialogue	33.9	26.1	21.4	29.1

3.4.1.2 | WoZ+ trials

On the other hand, the second WoZ experiments, the so-called WoZ+ trials, contain GROW sessions about nutrition and physical activity. Table 3.5 shows demographic data as well as the average quality of life and depression level for the 101 participants, while Table 3.6 summarises the statistics for this partition. In this case, the dialogues were not labeled in terms of semantics or emotions, and were not translated into the other target languages either, due to budget limitations. Anyway, the same questionnaires as in the WoZ trials were administrated this time.

3.4.2 | Semantic label taxonomy for the language understanding task

As aforementioned, the WoZ partition of the EMPATHIC corpus was labelled semantically and in terms of emotions. We focused on the semantics of the users, described in depth in Montenegro et al. (2019a). The design of the (user's)

Table 3.5.: Demographic data of the participants of the WoZ+ experiments.

	Spain	France	Norway
Number of participants	26	12	63
Gender	10 M, 16 F	4 M, 8 F	31 M, 32 F
Average age	70.0	75.8	72.6
Avg. WHOQOL-BREF score	70.1	71.4	74.1
Avg. GDS score	6.7	5.2	4.0

Table 3.6.: WoZ+ corpus summary.

	Spain	France	Norway
Video/audio files	52	24	131
Time	8:45:42	4:17:24	22:50:04
Transcribed dialogues	52	-	126

semantic label taxonomy in a SDS is essential because it defines the input space for the DM. The DM performs actions based on this representation, so it must be informative enough to allow the DM to execute the desired dialogue strategy, a GROW coaching strategy in this case. On the other hand, the taxonomy should not be too complex or open, because otherwise the Natural Language Understanding (NLU) module may have a hard time correctly classifying the user utterances. Considering these constraints, we proposed a multidimensional hierarchical taxonomy with four types of labels: *Topic*, *Intent*, *Polarity*, and *Entity* labels.

3.4.2.1 | DEFINITION OF THE TAXONOMY

We propose a hierarchical structure for the *Topic* and *Intent* labels. This means that an utterance is labeled by multiple tags that can be ordered from more general to more specific. Such labelling can be graphically represented using a tree (see Figures 3.4 and 3.5). In this structure, the closer a label is to the root, the more general it is; while the closer to the leaves, the more specific. The rationale behind the use of hierarchical labels is to allow the DM to receive more fine-grain information when possible, and less refined labels when no other choice is available. In addition, the NLU module also benefits from this, because it can be less precise at the time of making predictions in those situations in which the confidence is not high enough to discriminate between a set of labels, selecting the parent label.

The *Topic* label classifies the utterance to determine the general context in which the conversation is framed. Figure 3.4 shows the topic label tag set organised as a tree. Four main groups can be recognised: *sport and leisure*, *nutrition*, *family* and *other*, which are further split into more detailed categories. The first two groups were designed mainly for introductory and nutrition dialogues, and the *other* label for those generic sentences such as greetings or confirmations that cannot be related to any topic. The *family* group was intended to be used in later stages of the project, but finally dialogues about social activity were not recorded.

The *Intent* label classifies the utterance in classes related to the user's communicative intentions (e.g. *question*, *inform*, etc.). Figure 3.5 shows the hierarchical structure for the Intent tags. General purpose intents such as greetings, doubts or thanking, are grouped under the *Generic* and *Question* labels, and were inspired in the DIT++ taxonomy of general-purpose communicative functions (Bunt, 2009). These aim at providing the DM with some basic understanding. On the other hand, we also defined a set of task-specific labels related to the GROW model of coaching, aimed at helping the DM to detect Goals, Realities, Obstacles and Wills of the topics of interest.

The *Polarity* label aims at representing the sentiment associated with the semantics of the user turn, which can be very relevant to provide exploitable information to DM. We distinguish between three levels of polarity: positive, neutral and negative.

The *Entity* label classifies particular elements of user turns that provide specific semantic information. Table 3.7 shows the selected entities for the EMPATHIC project.

Table 3.7.: List of name entity categories.

People	Quantities	Objects/Utensils	Films/TV Series	Nature
Actions	Frequencies	Ordinal numbers	Nationalities	Emotions
Books	Music/Bands	Cardinal numbers	Relative dates	Diseases
Family	Sport/Leisure	Places/Buildings/Organizations	Meteorology	
Food	Time amount	Paintings/Sculpture/Art	Absolute dates	

In Figure 3.6, a labelling example is illustrated. This fragment of a dialogue has been labeled by a human, and it can be seen that sometimes it is not possible to reach the tree leaves of the *Topic* or *Intent* dimensions. For instance, the topic of the third user turn is clearly nutrition, but it's not about regularity, quantity or variety.

3.4.2.2 | Semantic annotation procedure and results

In order to semantically label the user turns of the WoZ dialogues, we instructed some annotators about the labels, the GROW model, and the context of the EMPATHIC project, in Spain, France and Norway. Each annotator labeled roughly

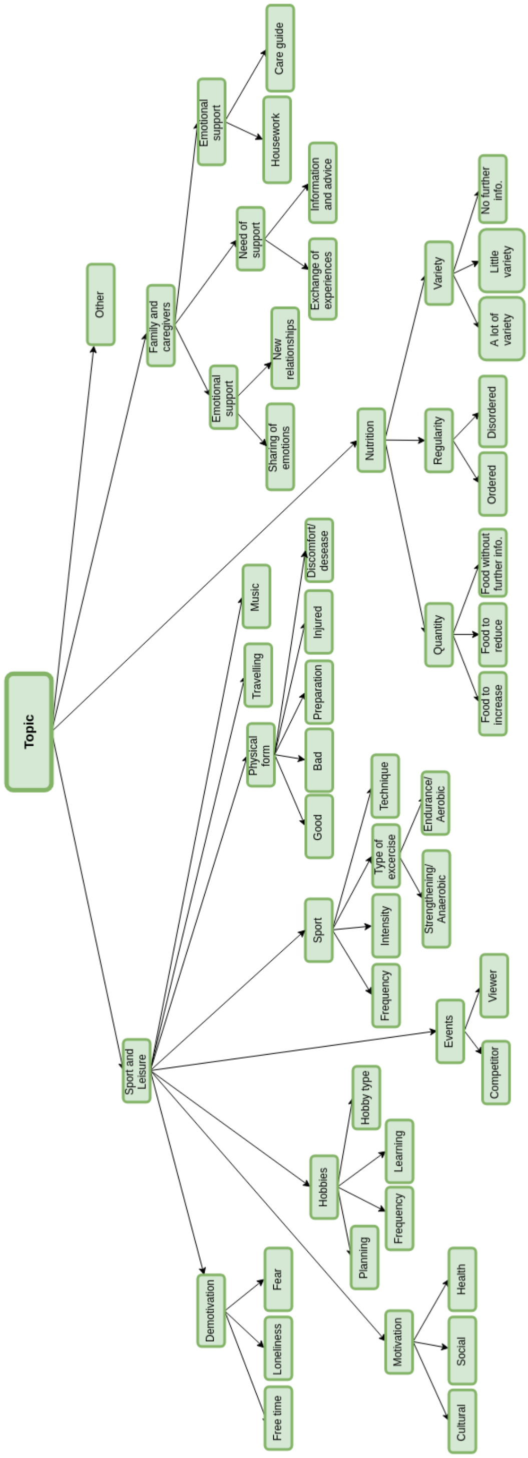


Figure 3.4.: Topic label tree.

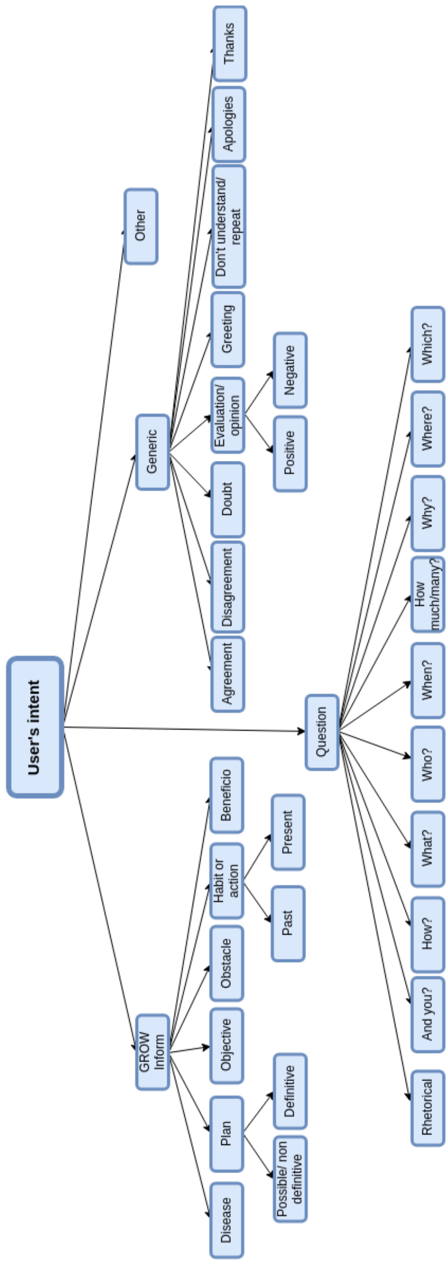


Figure 3.5.: Intent label tree.

the same number of dialogues per country, corresponding to both the introduction and the nutrition scenarios. Each dialogue was labelled by one annotator only. Nevertheless, all the annotators worked together to deal with doubts and disagreements, under close supervision, resulting in a collaborative annotation task.

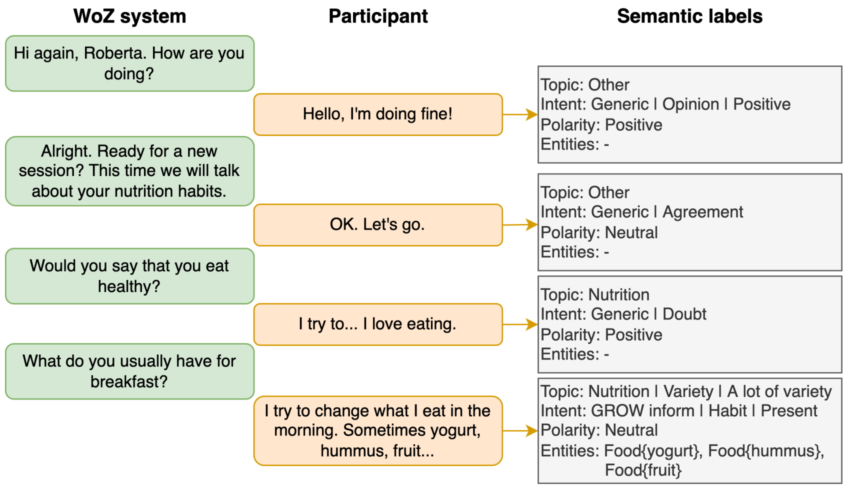


Figure 3.6.: Semantic labelling example.

Since more than one intent and/or topic may appear per turn, the annotators divided each turn into subsentences that roughly correspond to uttered clauses, so unique intent and topic labels can be assigned to each of these subsentences. To do so and to carry out the annotation procedure, we developed an annotation tool that provides a simple command-line interface. Annotators took around an hour to label each dialogue, on average.

In total, 7,842 user turns were labeled aggregating the results of the three countries. The turns were split in 15,661 subsentences, and 19,108 name entities were identified. The size of the corpus is the same in Spanish, French, Norwegian and English, due to the translations. Tables 3.8 and 3.9 show the number of subsentences labeled with each topic and intent label, respectively, and also the frequency of these labels. For the sake of clarity, we only show labels up to depth 2 labels in the trees. The numbers include the sum of all the sublabels below those. The sets marked with the symbol * refer to the rest of non mentioned sublabels. Table 3.10 contains the number of subsentences and frequencies corresponding to the three possible polarity levels. Last, Table 3.11 shows the number of identified name entities per category, and also divided by the number of user turns.

Regarding the frequency of the topic labels in Table 3.8. The *Other* label is the most frequent label, which is selected 66.0% of the time, and comprises generic utterances such as affirmations or greetings. The number of subsentences labeled with *Nutrition* (15.6%) or *Sport and leisure* (17.5%) labels is very similar. An interesting difference between the *Nutrition* and *Sport and leisure* annotations is that the *Nutrition* label alone, without any other sublabel, is much more frequently selected than the *Sport and leisure* one. This means that the defined sublabels under the *Sport and leisure* node in Figure 3.4 cover almost entirely the

Table 3.8.: Frequencies and number of subsentences corresponding to the most frequent topic labels.

Frequent topic labels	Number of subsentences	Frequency
Family and caregivers	131	0.8%
Nutrition (only)	1387	8.9%
Nutrition - Quantity	456	2.9%
Nutrition - Regularity	374	2.4%
Nutrition - Variety	233	1.5%
Sport and leisure (only)	172	1.1%
Sport and leisure - Travelling	1064	6.8%
Sport and leisure - Music	294	1.9%
Sport and leisure - Hobbies	770	4.9%
*Sport and leisure - **	608	3.9%
Other	10344	66.0%

Table 3.9.: Frequencies and number of subsentences corresponding to the most frequent intent labels.

Frequent intent labels	Number of subsentences	Frequency
Generic - Agreement	3127	20.0%
Generic - Disagreement	560	3.6%
Generic - Evaluation/opinion	3599	23.0%
Generic - Doubt	453	2.9%
Generic - Greeting	653	4.2%
*Generic - **	671	4.3%
GROW inform - Habit	2514	16.1%
GROW inform - Plan	406	2.6%
GROW inform - Goal	255	1.6%
GROW inform - Obstacle	266	1.7%
*GROW inform - **	438	2.8%
Question	548	3.5%
Other	2171	13.8%

spectrum of possible topics in the WoZ experiments. On the other hand, this is not often the case when the participant talks about nutrition. This is because the three *Nutrition* sublabels are quite specific, and not always necessary to understand the user. For example, a simple utterance such as “I like apples" would only be labeled as *Nutrition*, but the meaning of the whole sentence could be inferred from the intent (*General-Opinion-Positive*), polarity (*Positive*) and entity (*Food*{*apples*}) labels. We analyse the relationship between the different dimensions of the semantic labels at the end of the section.

As for the intent label distribution of Table 3.9, generic communicative intents are prevalent, especially agreements and opinions. Among the opinions, around 55% of them were positive, only 9% negative, and the rest were not labeled as positive or negative. The fact that agreements and positive opinions are

Table 3.10.: Frequencies of the polarity labels.

Polarity labels	Number of subsentences	Frequency
Positive	3134	20.0%
Neutral	12075	77.1%
Negative	452	2.9%

Table 3.11.: Distribution of name entities.

Entity labels	Number of entities	Entities per user turn ($\times 10^{-2}$)
Absolute dates	267	3.4
Actions	4022	51.3
Books	63	0.8
Cardinal numbers	691	8.8
Emotions	378	4.8
Family	296	3.8
Films/TV series	30	0.4
Food	3762	48.0
Frequencies	954	12.2
Diseases	113	1.4
Meteorology	54	0.7
Music/Bands	204	2.6
Nationalities	181	2.3
Nature	35	0.4
Objects/Utensils	518	6.6
Ordinal numbers	91	1.2
Paintings/Sculpture/Art	30	0.4
People	864	11.1
Places/Buildings/Organizations	1845	23.5
Quantities	2159	27.5
Relative dates	1151	14.7
Sport/Leisure	926	11.8
Time amount	474	6.0

much more frequent than disagreements and negative opinions, respectively, points out a certain positive attitude of the participants towards the WoZ system. This can also be seen in the polarity label distribution (Table 3.10), which indicates that only 2.9% of the subsentences were negative. In respect of the subsentences labeled as *GROW inform*, the majority referred to habits. This shows that the wizard spent a lot of time analysing the participants' routines. Last, it can also be seen clearly that, as planned, the participants did not ask many questions to the system. Instead, the sessions consisted of the wizard asking and the users talking, which is a big difference from many health-related dialogue systems that focus more on question answering than on having actual conversations.

The main identified name entities were, as shown in Table 3.11, *Actions*, *Food*, *Quantities*, *Places/Buildings/Organizations*, *Relative dates*, *Frequencies*, *Sport/Leisure* and *People*; which makes sense for an introductory dialogue and for a GROW coaching session for nutrition.

Let us now analyse the relationship between the topic, intent and entity labels through Sankey diagrams, in Figures 3.7, 3.8 and 3.9. In these figures, the most representative sets of labels of two dimensions face each other. The flows that connect the labels from one side with the other, represent the number of subsentences that are labeled with the two connected labels.

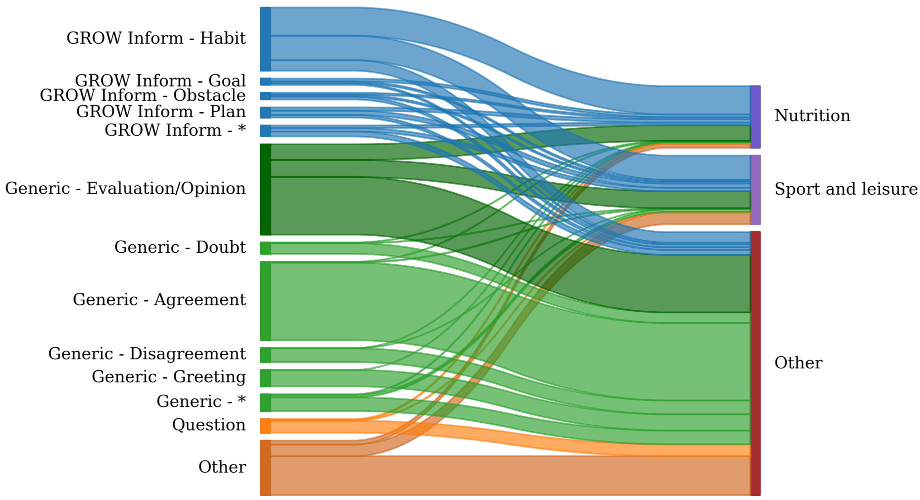


Figure 3.7.: Relation between intent (left) and topic (right) labels.

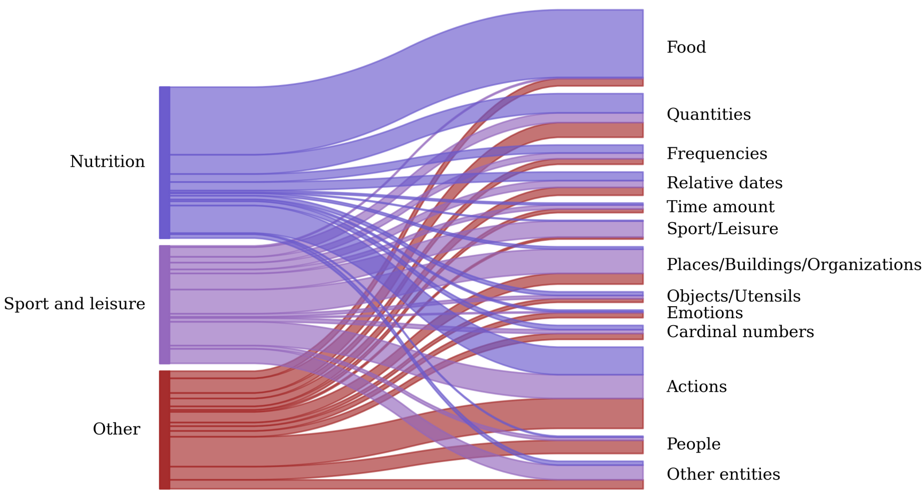


Figure 3.8.: Relation between topic (left) and entity (right) labels.

Let us start with the Sankey diagram for the intent and topic modalities, in Figure 3.7. The first thing that can be seen is that the labels of the *GROW In-*

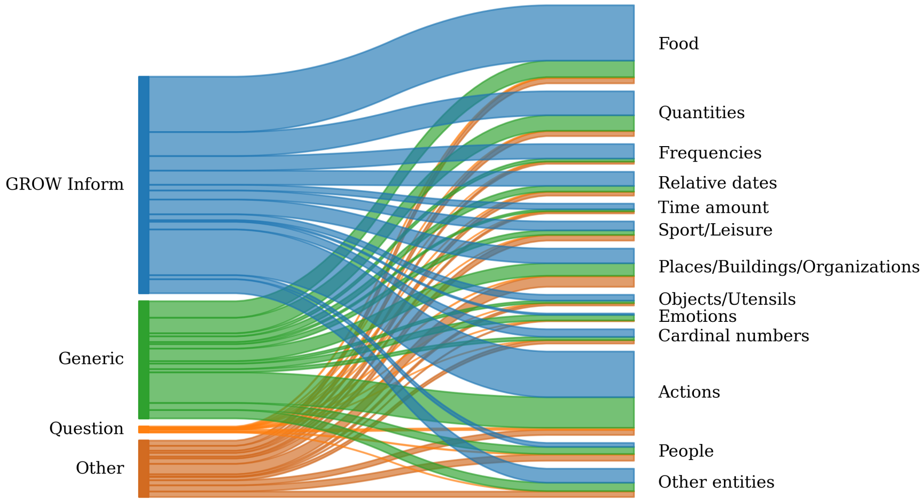


Figure 3.9.: Relation between intent (left) and entity (right) labels.

form family (in blue) mostly relate to the *Nutrition* and *Sport and leisure* topics. At first sight, it may seem unexpected that coaching responses appear when talking about *Sport and leisure*, because this is the main topic in the introductory sessions, where the GROW coaching session is not being carried out. The reason is that, as aforementioned, most of *GROW Inform* turns are about the participants' habits, and the wizard often asked about habits even in the introductory dialogue, to get to know the user better and to potentially detect improvable routines. On the other hand, there was no identifiable topic for most of the subsentences labeled with *Generic* intents (in green), as expected. The only significant exception happens with the *Evaluation/opinion* sublabel (highlighted in dark green), which also covers more complex turns where the user gives opinions, for instance, about their nutrition, hobbies, or some food. Similar to the rest of *Generic* intents, general Questions (in orange) are mostly related to the Other topic too.

The relations of the topic and intent labels with the identified name entities are represented in Figures 3.8 and 3.9, respectively. In this case, the width of the flow depicts the number of name entities identified in subsentences labelled with a given topic or intent. This is why the widths of the flows of the entity and intent labels differ from the ones in Figure 3.7. In Figure 3.8, for example, the width of the *Nutrition* or *Sport and leisure* sets of labels are equal or higher than the *Other* topic label, even if the 66.0% of sublabels were labelled as *Other*. This is due to much fewer entities appearing in subsenteces labelled as *Other*. Likewise, in Figure 3.9, the flow corresponding to *GROW Inform* subsentences is wider than the *Generic* one, even though there are fewer *GROW Inform* subsentences than *Generic* subsentences, because *GROW Inform* turns are much longer and more informative, and therefore contain more name entities.

The flows between the entities and the labels of the other modalities are consistent with the analysis so far, and with the expected relations. For instance, *Food* entities are mostly detected in subsentences about *Nutrition* and with *GROW Inform* communicative intent, and *Sport/Leisure* and *Places/Buildings/Organizations* entities in subsentences about *Sport and leisure*. Interestingly, *Quantities*, *Frequencies* and *Relative dates* entities are quite related to *GROW Inform* subsentences, which indicate that, as aforementioned, these turns are very informative and contain a lot of information that can be exploited by a VC.

3.5 | Contributions to the EMPATHIC Virtual Coach prototype

Besides the EMPATHIC corpus, another important outcome of the project was the development of a VC capable of carrying out GROW coaching sessions about nutrition. We collaborated on the design of the dialogue strategy (Section 3.5.2), and on the design and validation of the NLG post-process (Section 3.5.3). Before presenting our contributions, we provide a general description of all the modules that make up the VC in Section 3.5.1.

3.5.1 | Overview

The VC consists of a multimodal automatic dialogue system whose users can interact with through any device that has a web browser (PCs, Tablets or Smartphones, for example). This audiovisual interaction is based on the analysis of the voice and the images obtained from the microphone and camera available on the devices. In addition, the system allows the user to enter text when required. Figure 3.10 shows a senior interacting with the system. The picture has been extracted from a video of the EMPATHIC experiments in France, which is accessible at Olaso et al. (2021), subtitled in English.

Figure 3.11 shows the main components of the VC, where the modules we contributed to are highlighted. The NLU is highlighted too due to the definition of the taxonomy of the semantic labels. The system architecture mainly follows the conventional structure of SDSs, with the addition of some less conventional components to provide the system with additional capabilities. Each of the components is described as follows:

- ASR. The ASR was developed by the EMPATHIC partner Intelligent Voice[5]. It processes the user's voice and transforms it into a sequence

[5]https://intelligentvoice.com

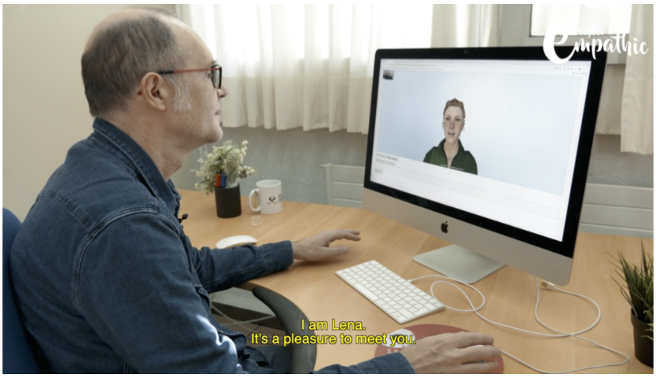


Figure 3.10.: A senior interacting with the EMPATHIC VC.

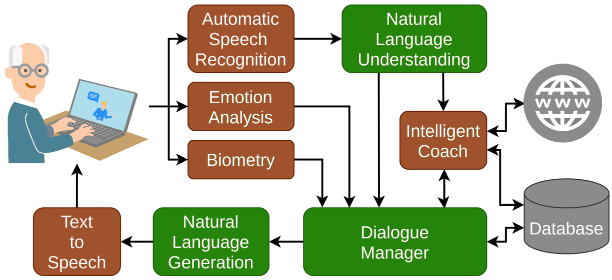


Figure 3.11.: EMPATHIC system schema. Green boxes highlight the components we mainly contributed to.

of words for each target language (López Zorrilla et al., 2016). It allows choosing between several speech recognition engines.

- NLU. The purpose of the NLU is to obtain the semantic representation (meaning) of the sequence of words obtained by the ASR. This meaning is represented with the dialogue act taxonomy explained back in Section 3.4.2 (Montenegro et al., 2019a). It also detects the end of the user turns (Montenegro et al., 2021).
- DM. It is responsible for determining what action the system should take in reaction to the user's turn. To this end, the DM takes into account the state of the dialogue as well as the input from several components, as shown in Figure 3.11. The proposed DM and the designed dialogues are fully explained in Section 3.5.2 (Vázquez et al., 2023) and its behaviour is analysed in Section 3.6.
- Natural Language Generation. This module is in charge of building and generating the language of the VC interventions. It transforms the DM output into a sequence of words. The proposed NLG methodology is fully explained in Vázquez (2019); Vázquez et al. (2023). In Section 3.5.3 we analyse more deeply our contribution to the EMPATHIC NLG.
- Text To Speech (TTS). It converts the text provided by the NLG into voice

and thus generates the oral response to be provided by the VC. The TTS, developed by Acapela[6], can generate male and female emotional voices for the three target languages.

- Emotion Analysis. One of the additional features of the VC is the ability to analyse user emotions. The emotion analysis is carried out from three sources: from the speech signal (Justo et al., 2020; Letaifa and Torres, 2021; Greco et al., 2021; de Velasco et al., 2022), from facial expressions (Nasri et al., 2020; Greco et al., 2021), and the gaze and eye position (Palmero et al., 2018). The output of these modules is fused to get a more accurate estimate of the user's emotional status (Huang et al., 2020), which is then provided to the DM.
- Biometry. In order to get access to the system, users have to undergo an authentication process. To this end, the system implements facial biometry from the analysis of the video sequences (Hmani et al., 2021).
- Intelligent Coach (IC). This component is responsible for live monitoring the conversations. It can take decisions about the ongoing dialogue such as redirecting the dialogue flow, e.g. changing the dialogue topic, when needed. In addition, this component can provide information extracted from external resources to the DM when requested. Some use cases for this component are described in Section 3.5.2.
- Visual Agent. The system communicates with the user through a talking 3D animated character[7]. It can implement different voices aligned with the movement of the lips as well as perform some head and facial gestures. Some of its movements are adapted to the semantic meaning of the system's utterances.

3.5.2 | Dialogue strategy

In this section, we first describe the proposed methodology for the DM in Section 3.5.2.1. Then, Sections 3.5.2.2 and 3.5.2.3 describe the task specification trees proposed to implement the introductory and coaching dialogues, respectively.

3.5.2.1 | Dialogue manager

For the management of the EMPATHIC dialogues, we grounded our design on a planning-based DM, RavenClaw (Bohus and Rudnicky, 2009). It is flexible and scalable enough to address domains needing different planning and communication skills. However, the tasks previously addressed are quite simple,

[6] https://www.acapela-group.com
[7] The avatar was provided by Institut Mines Télécom Paris.

mainly consisting of information-access (Raux et al., 2005; Ghigi et al., 2014; Olaso and Torres, 2017; Serras et al., 2019b) or scheduling (Olaso et al., 2016), which strongly differ from the coaching sessions that the VC has to manage in EMPATHIC. We show that more complex dialogue strategies can also be implemented in this framework (Vázquez et al., 2023).

RavenClaw develops a management structure based on distributed software agents that first specify the dialogue task at the design level and then execute the dialogue flow at the running time, as follows:

- Dialogue task specification. It follows a hierarchical plan that is defined by a tree of dialogue agents, where each agent is responsible for managing a specific subtask. Two different kinds of agents can be found in the tree:
  - Internal agents or non-terminal nodes, represented as blue nodes in Figures 3.12 to 3.18, are used to encapsulate subsections of the dialogues and control the execution of their children agents.
  - Terminal nodes, represented in red in the aforementioned figures, are responsible for implementing precise actions. For instance, `Inform` nodes produce an output, `Request` nodes ask for some information from the user, and `Expect` nodes continuously listen for some information without requesting it. Last, green nodes in Figure 3.12 represent `Execute` nodes connected to other modules of the SDS or external resources.
- Dialogue management. The DM executes a given dialogue task specification tree traversing it in Depth First order. However, this order can be altered under specific preconditions, triggers or success/failure criteria of the internal agents, as well as by external triggers. A dashboard that stores relevant information is used to keep the consistency of the dialogue when travelling the trees.

3.5.2.2 | INTRODUCTORY DIALOGUE

Figure 3.12 shows the dialogue task specification tree of the introductory dialogue. In a similar fashion to the first session of the WoZ experiments, it deals with user-friendly topics aimed to get basic information about the users, and also provides some context about coaching and the EMPATHIC project.

The session starts with the `Execute CheckFirstUse` node, which is used to select the `IsFirstUse` or `IsNotFirstUse` agents, based on the user ID. This allows the DM to behave differently if it is the first time a given user interacts with it or not, and provides the functionality of stopping in the middle of a session and continuing later from the same dialogue state. If `IsFirstUse` agent is selected, the Biometry module (see Figure 3.11) is triggered by the green agent

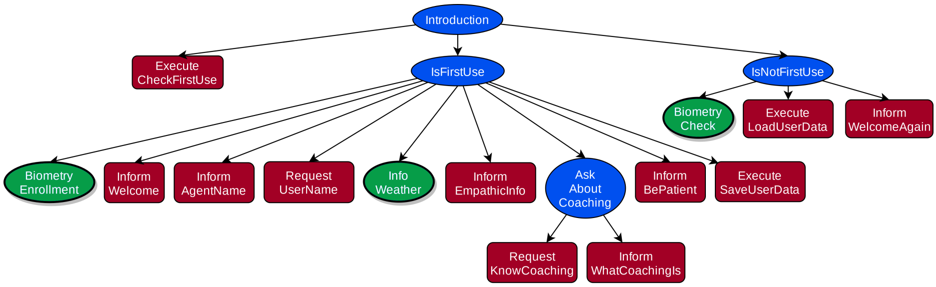


Figure 3.12.: Task specification tree for the introductory dialogue.

`Biometry Enrollment` to create a user profile from the video frames. If the user had already talked to the system, the Biometry module would authenticate them. The `Inform Welcome` terminal red note prompts a message to welcome the user, whereas `Inform AgentName` gets the name of the agent from the dashboard and then retrieves it in a message. In the same way, `Request Username` requests the user's name and stores it in the dashboard when provided. Next, the `Info Weather` green node is a good example of how the IC can be consulted by the DM to obtain information that can enrich the dialogues by adapting them to external conditions. This node provides weather forecasts to the user, based on their location. The remaining of the tree in Figure 3.12 provides information about the project via `Inform EmpathicInfo`, inspects whether the user is familiar with coaching methods (`Request KnowCoaching`), provides some more information in this regard (`Inform WhatCoachingIs`), gently asks the user to be patient with the system (`Inform BePatient`), and saves all the information acquired so far (`Execute SaveUserData`).

Additionally, the IC module in Figure 3.11 is live-monitoring the conversations and might redirect the dialogue flow. As a proof of concept for this capability, we added an agent that is able to provide users with culinary recipes (not shown in Figure 3.12). If the IC detects that a particular food has been mentioned several times during the conversation, it sends to the DM an order to provide the user with a recipe related to that food. Similarly, another agent was added to provide weather information when required.

3.5.2.3 | Coaching dialogues for the nutrition scenario

The nutrition dialogue design was based on the GROW coaching methodology (Section 3.2) and also took into account the user behaviours shown in the collected corpus of EMPATHIC (Section 3.4).

Figure 3.13 shows the task specification tree for the nutrition scenario. The children of the nutrition agent are highly connected to the GROW model phases. Following coaching experts' advice, a motivation phase (`M` in Figure

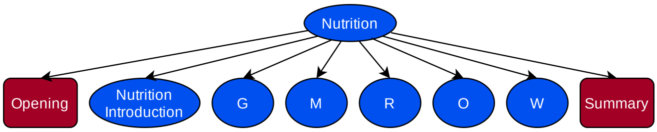


Figure 3.13.: Task specification tree for the Nutrition agent.

3.13), was additionally integrated by the nutrition agent to explore the potential motivations leading the users to change their nutritional habits. The GROW phase is preceded by an introductory nutrition dialogue (`Nutrition Introduction`). Mind this is not the introductory dialogue described in the previous section. Instead, it is designed to seek potential inappropriate habits by examining users' intake routines in their daily meals, namely breakfast, lunch and dinner.

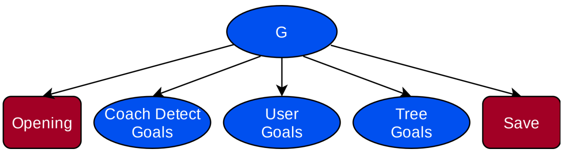


Figure 3.14.: Nutrition dialogues: specification for the Goal phase.

The Goal (G) subtree shown in Figure 3.14 is aimed at getting a nutrition goal from the user. We developed three strategies to this end. The main one starts by providing an analysis of the previously discussed users' nutritional habits. The VC enumerates detected potential issues, and asks the user if they would like to select a goal related to any of these issues. If this is not successful, the VC explicitly asks the user to specify what they wish to change regarding their nutritional habits. Last, if the system is still unable to detect the user's goal, it traverses some predefined user goals trying to set the one the user would like to solve.

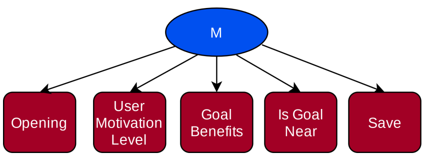


Figure 3.15.: Nutrition dialogues: specification for the Motivation phase.

Figure 3.15 shows a Motivation phase (M) aimed at: 1) evaluating the level of the user's motivation in relation to the objective, 2) increasing this motivation by discussing some potential benefits of the behavioural change, and/or 3) at exploring the nearness of the goal.

The purpose of the Reality (R) phase in Figure 3.16 is to find out which are the main obstacles the user has to achieve their goal. In particular, the system tries to detect possible obstacles based on the information obtained in the previous

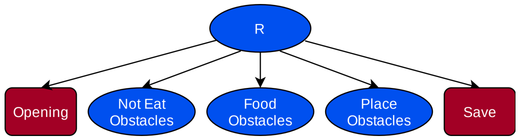


Figure 3.16.: Nutrition dialogues: specification for the Reality phase.

phases. Some possible obstacles could be: a lack of regularity in the main meals, excess of salt, sugar or carbohydrate intake, or an inappropriate environment for meals such as the workplace, among others.

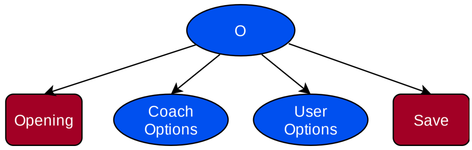


Figure 3.17.: Nutrition dialogues: specification for the Options phase.

The Options (O) subagent in Figure 3.17 explores the users' options to make some steps towards the completion of their goal. It focuses on three nutrition aspects, namely quantity, regularity and variety. If the goal is bound to one of these categories, the reduction of salt intake for example, the system goes ahead to the following subagent. Otherwise, the VC proposes a set of options related to their main issues. If the user does not select any of them, the system explicitly asks the user for a proposal.

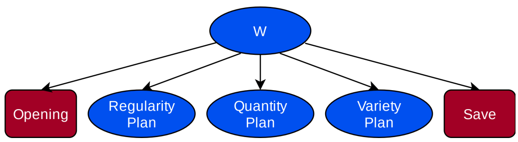


Figure 3.18.: Nutrition dialogues: specification for the Will phase.

The Will (W) phase, in Figure 3.18, is aimed to obtain a specific action plan that users should execute to achieve their goal. Potential action plans explored by the system and detailed by the user could be: increasing or decreasing the quantity of a given food, adding a new food to the user's diet, substituting some food with another, modifying the frequency of eating a specific food, or defining a regular timetable for having breakfast, lunch and/or dinner.

Finally and before closing the session, the system performs a summary with the main points and decisions taken, validates the users' commitment with the goals, and repeats the next steps to achieve them.

The proposed dialogue strategy is validated and evaluated in Section 3.6, where the behaviour of the EMPATHIC VC, guided by this DM, is analysed.

3.5.3 | Design and offline validation of the NLG post-process

The EMPATHIC NLG is a multilingual template-based language generation system. As any template-based system, it maps conceptual (non-linguistic) representations (i.e. dialogue acts output by the DM) to a template or set of templates. Templates are linguistic structures with possible slots that are filled to obtain well-formed sentences. These slot values are mostly name entities detected by the NLU and selected by the DM. However, not every template is appropriate for the selected slot values, because the resulting sentence could be grammatically incorrect or semantically incoherent. For example, let us say that the NLU detects the entity `dates`, which is a fairly common one. Table 3.12 shows two cases where only one of the two template candidates is correct. In these cases, the correctness is determined by the verb tense.

Table 3.12.: Examples of correct and incorrect NLG templates depending on the slot value.

Attribute value	Template	Generated sentence	Correct?
`date`=yesterday	What are you going to do `date`?	What are you going to do yesterday?	No
	What did you do `date`?	What did you do yesterday?	Yes
`date`=Mondays	What are you going to do `date`?	What are you going to do Mondays?	No
	What are you going to do on `date`?	What are you going to do on Mondays?	Yes

In some cases, the correct template can be selected after analysing some grammatical aspects of the slot values, such as the number, gender or countability, depending on the language. Nevertheless, in other cases, an ad hock analysis would be too complex. For instance, potential errors may arise from wrong verb conjugations, wrong or lack of determinants and pronouns (in Spanish or French), wrong prepositions and their position in the sentence in Norwegian, or semantically incoherent verbs given some food (such as *drink meat* or *eat water*).

3.5.3.1 | NLG postprocessing

Consequently, our contribution to the EMPATHIC NLG is the proposal of using a LM of the target language to select a correct option among the set of candidates. The NLG selects the candidate for which the LM estimates the highest probability. We have employed the GPT-2 neural LM architecture, which has been adopted and proved to be successful in many natural language processing tasks (Radford et al., 2019). However, the publicly available GPT-2 models are mainly developed for the English language, but we are interested in French, Spanish and Norwegian. Thus, we trained these networks from scratch for the three target languages.

For this purpose, the Spanish, French and Norwegian versions of Wikipedia and OpenSubtitles (Lison and Tiedemann, 2016) corpora were selected. This choice is based on the availability of the corpora for the three target languages. Wikipedia contains information about millions of topics, and OpenSubtitles mainly consists of conversations. A fraction of the Norwegian version of the OSCAR text corpus (Ortiz Suárez et al., 2020) was also included since the amount of data for Norwegian was much lower than for Spanish and French. OSCAR is a subset of Common Crawl, and thus it is made of web-scrapped text from the Internet. Table 3.13 shows some statistics of our training data for each language. Apart from the NLG postprocessing, these GPT-2 models were also employed to train a fully end-to-end coaching model, described later in Chapter 4.

Table 3.13.: Statistics of the corpora used to pretrain the GPT2 model in Spanish, French and Norwegian. In Norwegian, values in brackets refer to the data before the addition of a fraction of the OSCAR corpus.

	Spanish	French	Norwegian
Amount of raw text	10 GB	7 GB	5 GB (1 GB)
Number of sentences	230M	121M	30M (14M)
Running words	1.7B	1.3B	750M (150M)

We also tested the well-known statistical N-grams approach. But as Section 3.5.3.2 shows, its performance was clearly worse than the one obtained by the transformer-based LMs, due to the difficulty of the task.

3.5.3.2 | Postprocessing performance

We built 10 tests to measure the performance of the trained GPT-2 networks and N-grams LMs. Each test contains several tuples of sentences related to the task where only one is correct. We built the tests manually after analysing errors of previous versions of the NLG. We measured the percentage of the times each LM selected the correct candidate, i.e. the accuracy. Table 3.14 summarises the task to be carried out in each test. The beginning of the test name indicates which language it has been designed for.

The accuracies at selecting the correct candidate in the different tasks are shown in Table 3.15. The N-gram model is a 3-gram model, and we show two results for each GPT-2 model in each language, which correspond to models trained after 1 and 2 epochs, respectively. Table 3.15 demonstrates that the GPT-2 models outperform the N-gram model in all the tasks, validating the inclusion of this kind of transformer models in the NLG. It is particularly interesting that even though in Spanish and French training the models during two epochs improved the results compared to only one, this did not happen in Norwegian, where the 1-epoch GPT-2 model was the best performing LM. This is probably due to the less amount of good quality data available in that language, which may have caused the neural network to slightly overfit.

3.6 | Analysis and validation of the behaviour of the EMPATHIC VC

Last, we present an analysis of the results of the interaction tests between the EMPATHIC VC and the target population, which were carried out at the end of the project. We compare the outcome of these experiments with previous WoZ and WoZ+ trials whenever possible, in order to understand how far (or close) our automatic system is from human-operated ones.

We begin the analysis by presenting the experimental conditions in Section 3.6.1. Then, Section 3.6.2 offers a first glimpse of the conversations carried out with the system, with some statistics about dialogue and turn lengths. We compare those metrics with the WoZ experiments. Subsequently, in Section 3.6.3 we analyse the dialogue flow followed by the VC to check if its behaviour was the desired one, i.e. the strategy specified through dialogue trees. Section 3.6.4 provides information about task completion. In our case, this metric indicates how many phases of the GROW model users completed per session. Afterwards, we measure the online NLG performance in Section 3.6.5. Finally, we analyse the human acceptance of the VC prototype and compare it with the WoZ system.

3.6.1 | Experimental conditions and data

In total, 79 elderly participants took part in these tests: 31 in Spain, 22 in France and 26 in Norway. Due to the Covid-19 pandemic, the trials were carried out remotely. Each participant carried out a coaching session about nutrition, often split into two parts to let them take a break if they desired to. Demographic information about the participants is shown in Table 3.16, along with information about their quality of life (measured in a 0-100 scale via the WHOQOL-BREF questionnaire (World Health Organization et al., 1996)) and depression level (measured in a 0-30 scale via the GDS questionnaire (Sheikh and Yesavage, 1986)). More information on how to interpret these scores can be found in

Table 3.14.: Summary of the tasks to analyse the LMs' performance at selecting correct templates. In the examples, the attribute is underlined.

Name	Nb. of tuples	Nb. of options	Brief description	Correct sentence example	Incorrect sentence example
es_verb_time	1000	2	The verb has to match the adverbial of time.	¿Y qué ha sucedido ayer?	¿Y qué sucederá ayer?
es_verb_num	45	2	The verb conjugation has to match the number of the subject.	¿Cómo va a ser de momento el desayuno?	¿Cómo van a ser de momento el desayuno?
es_det	250	4	The determinant, if necessary, has to match the attribute.	¿Qué vas a hacer con el vino?	¿Qué vas a hacer con los vino?
es_det_verb	240	6	es_verb_num and es_det tasks combined.	Así que me dices que te gusta la natación.	Así que me dices que te gustan los natación.
es_food	40	8	es_det task with the additional condition that the selected verb makes sense with the attribute.	¿Cuántas manzanas te gustaría comer?	¿Cuánta manzanas te gustaría beber?
fr_verb_time	1400	2	The verb has to match the adverbial of time.	Et qui était avec vous autrefois?	Et qui sera avec vous autrefois?
fr_verb_num	120	2	The verb conjugation has to match the number of the subject.	Que vous ont apporté les vins?	Que vous a apporté les vins?
fr_det_pron	1640	8	The determinant and pronoun, if necessary, have to match the attribute.	Dans quelle mesure ce deuxième plat vous rapproche-t-il pour atteindre votre objectif?	Dans quelle mesure les ce deuxième plat vous rapprochent-elles pour atteindre votre objectif?
fr_food	567	4	Distinguish between countable and uncountable food names.	Quelle quantité de sucre?	Combien de sucre?
no_verb_prep	104	4	The attribute has to fit with the verb and the preposition. Its placement has to be correct as well.	Ønsker du å spise nå?	Ønsket du å spise i nå?

Table 3.15.: Template selection accuracies. The models are different for each language. Since the number of candidates also differs across tasks, the performance of a random classifier is provided as a reference.

Accuracy	Random classifier	N-grams (N=3)	GPT-2 (1 epoch)	GPT-2 (2 epochs)
es_verb_time	50.00	52.86	63.93	**80.75**
es_verb_num	50.00	55.56	77.78	**86.87**
es_det	25.00	26.55	49.09	**96.00**
es_det_verb	16.67	29.26	60.37	**98.15**
es_food	12.50	30.00	10.00	**60.00**
fr_verb_time	50.00	59.69	58.44	**68.94**
fr_verb_num	50.00	50.00	64.17	**82.50**
fr_det_pron	12.50	12.50	36.85	**44.80**
fr_food	25.00	36.79	39.26	**53.33**
no_verb_prep	25.00	26.47	**76.94**	73.86

Section 3.4.1.1. Table 3.17 contains a summary of the amount of data recorded. These data are also distributed along with the EMPATHIC corpus (Section 3.4).

Table 3.16.: Data about the participants that interacted with the EMPATHIC VC.

	Spain	France	Norway
Number of participants	31	22	26
Female participants	17 M, 14 F	10 M, 12 F	18 M, 8 F
Avg. age	71.6	68.4	73.4
Avg. WHOQOL-BREF score	68.6	65.1	75.2
Avg. GDS score	4.2	6.8	3.7

Table 3.17.: Summary of the data acquired from the interactions with the EMPATHIC VC.

	Spain	France	Norway
Video/audio files	108	86	91
Time	7:50:54	6:57:56	6:01:36
Transcribed dialogues	106	-	51

The number of videos is higher than twice the number of users because sometimes the system got stuck and had to be restarted. These are also distributed because they could be used for research purposes. Besides manual transcriptions (in Spanish and Norwegian), automatic ASR transcriptions of all sessions are provided too, as well as NLU outputs and NLG inputs. The results of the same questionnaires as in the WoZ trials are distributed too.

3.6.2 | Dialogue and turn lengths

In this section, we present some statistics of the dialogues carried out between the participants and the SDS. Figures 3.19 and 3.20 show that the system was able to keep long conversations, both in terms of the number of turns and total time. On average, the dialogues lasted 16.5 minutes in Spanish, 19.9 in French and 20.4 in Norwegian, without including the duration of the break. In terms of the number of turns, the dialogues in Spanish were 27.7 turns long on average, 35.0 in French and 37.2 in Norwegian. The main reason the system was able to hold significantly longer conversations in French and Norwegian is that the Spanish tests were the first to be carried out. Consistently, the DM was improved to address major prototype issues that had occurred during the tests in Spain[8]. Therefore, the French and Norwegian versions were more stable and

[8] We would like to note that despite these small stability improvements, the dialogue strategy and language generation were virtually equal in the three languages. The main differences in the rest of the modules were due to the different amounts of external training data.

fewer dialogues had to end prematurely. However, an additional reason for the longer conversations in French and especially in Norwegian, is the lower ASR performance. In these languages, users had to repeat some information more frequently until the system correctly understood it.

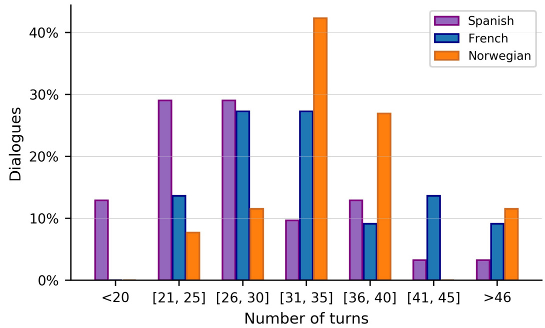


Figure 3.19.: Histogram of the number of turns per dialogue in the human evaluation of the final prototype.

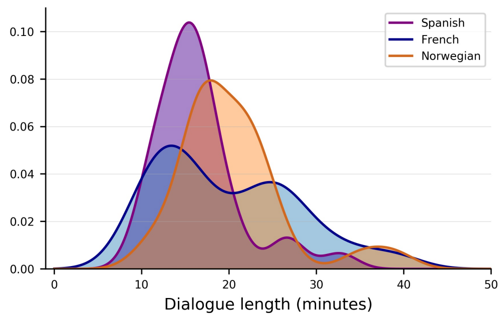


Figure 3.20.: Distribution of the time (in minutes) the dialogues lasted per language in the human evaluation of the final prototype.

Let us now pay close attention to the number of words per user turn. This metric is very relevant because longer responses often correlate with higher user engagement (Ghandeharioun et al., 2019). Intuitively, if the users are comfortable talking to a system and they feel they are being understood correctly and responded to coherently, they are much more likely to be more talkative and provide more information. On the other hand, if the system is having trouble understanding what the user is saying and makes them repeat information frequently, chances are that they will answer with much fewer words so that the system understands them better. Figure 3.21 shows a histogram of the number of words per user turn, divided per language, while Table 3.18 compare the average results with the WoZ experiments. In the table, the star symbol (*) indicates when a result is significantly better than its counterpart. More specifically, it means that p-value$\leq$0.05 using Welch's t-test, which tests whether two populations have equal means, without assuming equal variances. We use such a statistical test and p-value threshold in all the comparisons in this chapter.

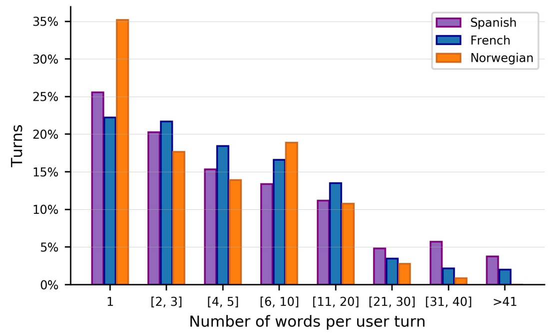


Figure 3.21.: Histogram of the number of words per user turn in the human evaluation of the final prototype.

Table 3.18.: Average number of words per user turn in WoZ and automatic VC experiments. The mark * indicates statistical significance.

	WoZ and WoZ+ combined	VC prototype
Spanish	**12.9***	9.5
French	**18.2***	7.6
Norwegian	**17.9***	5.4

The distribution shown in Figure 3.21 indicates that even though the dialogues were quite long, not all the user turns were so. Many turns were made of very few words, but a few of them were really long. As aforementioned, this depends on the willingness of the user to interact with the system, but also on the system's questions. In any case, longer responses were produced in Spanish, then in French, and the shortest were in Norwegian. Shorter responses in Norwegian were conditioned by worse performing ASR[9] and NLU, caused by the lesser amount of language resources for that languages. On the other hand, the difference between Spanish and French might be due to Spanish being the mother tongue of the main developers, and thus the system was tested mostly in this language, leading to fewer understanding errors. Cultural aspects of the users could also be a potential explanation.

In any case, the user engagement level was significantly lower in the tests with the automatic VC than in the WoZ trials, as expected. This is shown in Table 3.18 in terms of turn lengths, and further validated later in Section 3.6.6 with human questionnaires. Even if the difference could be partly explained

[9]The Word Error Rate (WER) achieved by the selected ASRs was 24.69 for Spanish, 30.41 for French and 43.34 for Norwegian.

by the conversational style of the wizards (which may use more open questions than the automatic prototype), it still indicates that users are much more talkative when talking to humans (or, for that matter, to a WoZ system) than to automatic systems. Last, a potential reason to explain why the number of words per user turn is lower in Spanish than in French and Norwegian could be the more direct conversational style of the Basque culture.

3.6.3 | Dialogue flow

The dialogue flow followed in the test dialogues can be helpful to better understand the task, its complexity and to validate the behaviour of the DM, described in Section 3.5.2. Figure 3.22 shows this dialogue flow in the form of a directed graph. The system turns are grouped into different nodes depending on their dialogue act. The arrows represent significant transitions, i.e., an arrow from a node A to a node B indicates that system turns grouped in node B have followed turns grouped in node A many times. The arrows are drawn if the corresponding transition happened at least 10% of the time node A was visited. This is done to keep the graph clearer and more representative. The nodes are coloured according to the dialogue phase they belong to. The arrows' width indicates the number of times a path was taken; wider arrows represent more common transitions.

First of all, the sequential nature of the automatic GROW sessions can clearly be seen in Figure 3.22, which also means that the implementation and the design of the DM were correct and that the system acted as expected. In the graph, the nodes corresponding to the same dialogue phase are clustered together, and they only precede nodes of the same phase or the next one. The main exceptions are premature endings of the session, and jumping from the Reality phase to the Will phase skipping the Options phase. Premature endings happen when the user and the system do not successfully agree on some aspects needed to proceed. For example, the session might finish in the Goal phase if the user does not have any goal, in the Options phase if the system is not able to understand the next steps the user proposes to achieve their goal, or in the Will phase if the system and the user are not able to specify an action plan.

Another interesting behaviour can also be recognised in Figure 3.22 via the cycles within the dialogue phase. On the one hand, self-loops (a node with a transition to itself) are due to the nature of the nodes of the graph: they are not system turns, but groups of them. The turns are grouped according to the semantics of the dialogue acts. For instance, the self-loop in *request user name* happens because after the system asks for the user name, it asks for confirmation in the next turn, but these two turns are gathered in the same node. The other loops that often appear in the graph are transitions from the last nodes

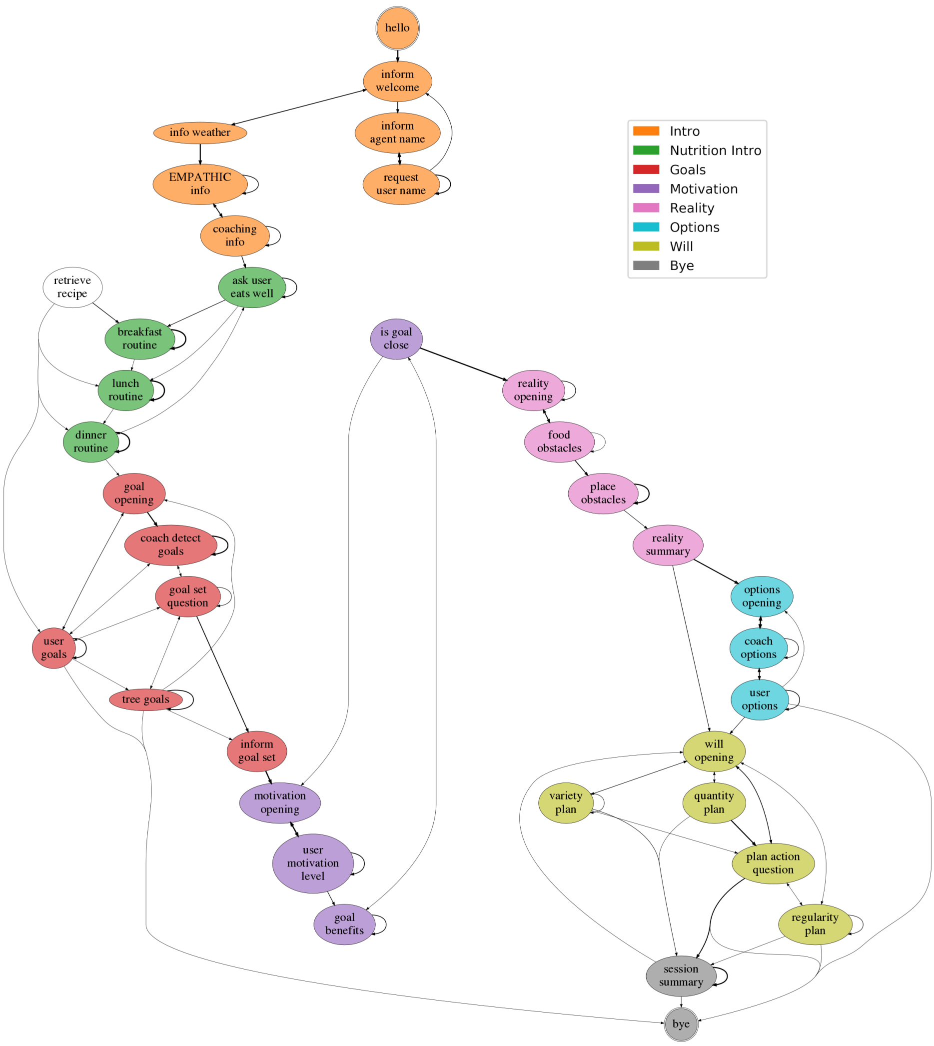


Figure 3.22.: Dialogue flow graph obtained from the interactions between users and the final prototype of the VC. The nodes are groups of system turns that are triggered at a given point of the dialogue, and the arrows indicate common transitions in these interactions. The colours indicate the dialogue phase the grouped turns belong to.

of a GROW phase to the first. This happens in the Goal phase (see the transitions from *tree goals* or *user goals* to *goal opening*), in the Motivation phase (from *is goal close* to *motivation opening*), in the Options phase (from *user options* to *options opening*) and in the Will phase in many occasions. These cycles are due to the system stopping and then restarting again. As aforementioned, since the conversations are typically long, the system was prepared to offer the participants some rest, and it could also stop if the user desired to do so at any moment. After the break, the dialogue is restarted from the beginning of the GROW phase the conversation was at, and keeping track of all the previously discussed topics and decisions.

Finally, we would also like to mention the behaviour of the *retrieve recipe* node. It is not coloured because it does not belong to any dialogue phase intrinsically; it is activated only when the user repeats the name of a given food. In this case, the system provides healthy recipes related to it in the middle of the dialogue, which could be useful to the participant. As expected, these system turns are triggered mostly in the Nutrition Introduction phase, where the participant tells the system about their nutrition routine.

In summary, the dialogue flow validates the design of our dialogue engine– and also of the system, in general. This dialogue flow corresponds to a successful implementation of the dialogue trees designed in Section 3.5.2.

3.6.4 | Task-completion

Task-oriented dialogue systems' performance is usually measured via task-completion metrics, if the task is relatively simple, such as retrieving some information or booking restaurants. For instance, in the case of restaurant reservation, the task completion would indicate the percentage of dialogues where the system successfully books a restaurant satisfying the user constraints. However, our task is far more complex and therefore it cannot be easily measured when the task is completed, or to what extent. In order to provide an idea of how well the system is doing at carrying out the GROW sessions, we analyse the percentage of the dialogue phases successfully finished throughout the dialogue with end-users. This is shown in Figure 3.23.

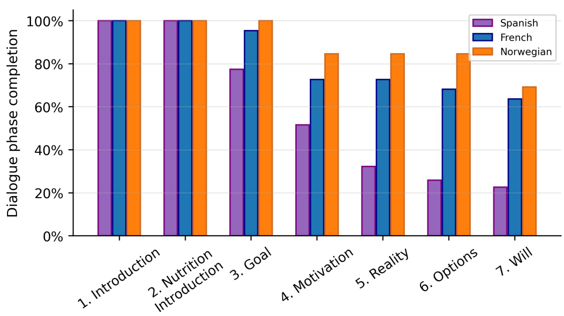


Figure 3.23.: Percentage of dialogue phases the system completed on average, per language.

Let us first highlight that the task completion is higher in Norwegian and French than in Spanish, especially in the later phases of the conversation. This is related to the aforementioned initial bugs in the Spanish version, which sometimes made the sessions end prematurely, or did not allow the system to restart correctly after the break.

We would like to remark the task-completion percentage at the Goal phase. It is 100% in Norwegian, very close to that number in French, and almost 80% in Spanish. This is already a very successful result, since establishing a goal the user would like to accomplish is the longest and most complex task in these dialogues. In fact, in previous Wizard of Oz experiments within the project, the dialogues where a user goal was found were considered successful (Justo et al., 2020). However, due to the length of the dialogues and the complexity of this first stage, many users were tired at this point and this is the reason why the principal drop in the task-completion is found in the fourth phase (Motivation). In some cases, the users considered there was no need to change their nutrition habits, so the session did not go further. The consequent drops are mostly caused by the system not being able to fulfil the objectives of each phase.

In the end, around 65% of the participants in Norway and French were able to establish not only a goal but also a plan to get closer to it in the near future, while around 25% of the participants in Spain were able to do so. This suggests that our proposal is valid to produce long and complex dialogues which can potentially improve nutrition-related habits, especially after the improvements of the VC implemented in Norwegian and French.

3.6.5 | NLG PERFORMANCE

As in Laranjo et al. (2018), we have computed the ratio of user turns labelled as a repetition request by the NLU as a measure of the quality of the NLG: the more repetition request from the users, the more likely the NLG is producing not-understandable sentences. Note that this metric is a lower bound of the actual NLG performance; it may happen that the user does not understand the system due to inaccuracies in other modules, such as the TTS, for example. The repetition request ratios obtained are shown in Table 3.19. Since the average number of turns per dialogue is 32.9, there is roughly only one repetition request per dialogue, which points out that the sentences produced by the NLG are highly comprehensible.

Table 3.19.: NLG errors measured as the percentage of repetition request turns by the user per dialogue.

Spanish	French	Norwegian
3.4%	3.5%	5.0%

3.6.6 | Human acceptance

Finally, we measure participants' perception of the VC prototype as well as their perception of the conversation flow, using an extended version of the VAAQ (Esposito et al., 2018), which can be found in Appendix B. In addition to the four subquestionnaires of the VAAQ, a short questionnaire about the agent's intelligibility was included too. In short, the users were asked about their perception of the system about the following qualities:

- Pragmatic qualities, which focus on the usefulness, usability, and accomplishment of the tasks of the proposed system, in this case, the GROW session.
- Hedonic qualities (identity), which are related to the system's personality.
- Hedonic qualities (feelings), which focus on how captivating the system is, and how the users felt while conversing with it.
- Attractiveness, which focuses on how tempting and attractive the interaction with the agent is.
- Intelligibility, which refers to the system's output, including the generated language and voice.

The questions are formulated to be answered on a 5-point Likert scale, which allows computing the score of each subquestionnaire easily, between 0 and 100 in this case. The results, in terms of average score and 95% confidence interval, are shown in Tables 3.20, 3.21 and 3.22; for Spanish, French and Norwegian, respectively. For comparison purposes, we also show the results of these questionnaires for the WoZ and WoZ+ experiments aggregated.

Table 3.20.: VAAQ average score and 95% confidence interval (in square brackets) per subquestionnaire for WoZ and WoZ+ experiments aggregated, and for the VC prototype, in Spanish. The mark * indicates statistical significance.

	WoZ (+)	VC prototype
Pragmatic qualities	**63.03**, [60.20, 65.86]	58.47, [52.32, 64.62]
Hedonic qualities (identity)	**71.67***, [68.86, 74.49]	65.44, [60.05, 70.84]
Hedonic qualities (feelings)	**62.45***, [59.15, 65.75]	52.50, [44.42, 60.58]
Attractiveness	**64.69**, [62.08, 67.31]	61.36, [54.15, 68.57]
Intelligibility	**70.97**, [64.16, 77.77]	63.61, [58.02, 69.20]

According to the three result tables, the system obtains mostly positive results (>50), which indicates a correct behaviour of the integrated VC and confirms the good design of the DM and NLG in this very challenging task. However, due to the complexity of developing automatic GROW sessions, there is still room for improvement, as shown by the difference between the automatic VC and WoZ results.

Table 3.21.: VAAQ average score and 95% confidence interval (in square brackets) per subquestionnaire for WoZ and WoZ+ experiments aggregated, and for the VC prototype, in French. The mark * indicates statistical significance.

	WoZ (+)	VC prototype
Pragmatic qualities	**60.79**, [56.11, 65.47]	51.81, [43.42, 60.21]
Hedonic qualities (identity)	**76.99**, [72.38, 81.61]	71.38, [64.52, 78.23]
Hedonic qualities (feelings)	**64.36***, [58.47, 70.25]	45.65, [36.11, 55.19]
Attractiveness	**66.89**, [62.99, 70.79]	61.78, [54.04, 69.51]
Intelligibility	62.50, [49.68, 75.32]	**67.75**, [61.48, 74.02]

Table 3.22.: VAAQ average score and 95% confidence interval (in square brackets) per subquestionnaire for WoZ and WoZ+ experiments aggregated, and for the VC prototype, in Norwegian. The mark * indicates statistical significance.

	WoZ (+)	VC prototype
Pragmatic qualities	**57.17**, [53.73, 60.62]	47.50, [37.57, 57.43]
Hedonic qualities (identity)	**70.88**, [67.75, 74.00]	66.99, [58.35, 75.62]
Hedonic qualities (feelings)	**56.93**, [52.88, 60.98]	48.88, [38.30, 59.45]
Attractiveness	**57.34**, [54.17, 60.51]	50.80, [41.74, 59.86]
Intelligibility	**64.85**, [61.11, 68.60]	63.62, [55.62, 71.62]

If we compare the results obtained in the three countries, the human perception of the Spanish system is similar to the French one, and better than the Norwegian one. This correlates well with conclusions extracted from turn lengths (in Section 3.6.2), and once again emphasises the influence of other modules besides the DM in SDSs, which are probably the cause of these differences, as previously explained. On the other hand, and as expected, VAAQ scores are once higher for WoZ experiments than for the automatic system. Nonetheless, the differences are significant only in three cases, as opposed to the previous comparison in terms of turn length (see Figure 3.18). This indicates that even if the WoZ system is notoriously more engaging and makes the users more talkative, their perception of the VC prototype is not significantly worse in many aspects.

To provide a more detailed view of the user's perception of the system, we also show the score corresponding to seven specific questions related to the NLG and DM modules, in Table 3.23. These questions can help us gain a deeper insight into the positive points of the system, and also into its drawbacks. Questions marked with a dagger (†) ask about potential negative opinions on the system, but higher scores always mean higher performance. On the other hand, it is also necessary to say that the questionnaire was administered in the three languages, and that sometimes the results are not completely comparable due to subtle differences in the connotation of the employed words.

Table 3.23.: Scores of seven VAAQ questions for WoZ and WoZ+ experiments aggregated, and for the VC prototype. The mark * indicates statistical significance. All the scores are in the range of 0-100, and higher scores always indicate better performance.

WoZ(+)/VC proto.	Spanish	French	Norwegian
I think that communicating with the agent is simple and easy.	**72.3**/66.7	53.1/**57.6**	**66.0**/53.8
I think that communicating with the agent is useless.†	**70.0**/66.7	**67.9**/62.0	**63.3**/54.8
I think the agent is very human.	**48.7***/34.2	**57.7***/40.2	**48.9**/46.2
I think that communicating with the agent is enjoyable.	54.7/**63.3**	60.3/**63.0**	**45.1**/38.5
I think that communicating with the agent is engaging.	**69.6***/56.9	**72.3***/48.9	**60.1**/52.9
I think that communicating with the agent is stressful.†	76.0/**78.3**	85.3/**87.0**	**67.1**/55.8
The agent can be easily understood.	**88.0**/82.5	75.0/**82.6**	82.3/**82.7**

With scores between 50 and 75, depending on the country, the participants considered that communicating with the agent was rather simple and easy, and that this communication was not useless. In other words, the users were, in general, able to take advantage of the virtual GROW sessions. In comparison with the WoZ system, the biggest difference happens with the Norwegian system, due to the aforementioned reasons.

The next four questions evaluate other aspects of our system: it is not very human, and the communication, even though useful and enjoyable (in Spanish and French), it is not particularly engaging. This indicates that the interaction with the system is far from perfect, but since our work represents one of the first steps in building complex coaching systems, we find it acceptable. It is noteworthy that users find the VC prototype slightly more enjoyable than the WoZ system in Spanish and French, even if not significantly. We hypothesise that this might be due to the increased delay of the WoZ system, produced by the wizard having to think and (sometimes) write the next response. Regarding how engaging the automatic VC is, the difference with the WoZ is once again notable (as in Section 3.6.2). This also suggests that using the turn length as an engagement metric can be appropriate.

When the participants were asked whether the communication was stressful, the French and Spanish answered quite strongly that it is not–even less than the WoZ system. According to the rest of our analysis, Norwegian users found it more stressful. Finally, the last question confirms the good performance of the NLG: the users in the three countries thought that the agent can easily be understood, which could not be possible had the NLG produced grammatically or semantically incorrect sentences.

3.7 | Conclusions

The EMPATHIC project has been one of the first big efforts to explore how VCs may improve independent healthy-life-years of elderly. Besides the technical aspects discussed in this chapter (and the ones to be discussed in the next one), EMPATHIC has provided an interesting inside into how the target population behaves when interacting with this kind of automatic systems. This behaviour is reflected in the data recollected through the WoZ trials and labelled, among others, according to the proposed semantic label taxonomy; as well as on the experiments with the automatic VC. User impressions are also documented in the administered questionnaires. All of this information is included in the released EMPATHIC corpus, which should be useful for future research in this area.

On the methodological side, we would like to note that the modules of the EMPATHIC prototype, and particularly the DM and NLG, have been developed using rather conventional or classical methodologies, instead of cutting-edge data-driven approaches. While the performance of fully data-driven models has remained unmatched in many areas of AI for many years already, such as image recognition (Deng et al., 2009) or machine translation (Sutskever et al., 2014; Bahdanau et al., 2014; Vaswani et al., 2017), this has not yet been the case for dialogue management or language generation. The main reason for this is the little or no control over their behaviour once they have been trained, and the potential lack of domain-specific data at the beginning of research or industrial projects. Even if statistical models produced the desired responses many times, the fact that their behaviour cannot easily be controlled is a big issue for NLG and particularly for DM. This underlines that there is still a need for future work if data-driven dialogue models want to be employed regularly in real-life applications. Also, hybrid models of data-driven and rule-based DMs (Williams et al., 2017; Griol and Callejas, 2019) may be an interesting alternative to this issue, which could benefit from the best of both worlds.

Last, and regarding the transferability of the EMPATHIC VC, it is the basis of the SDS of the project GO-ON (Tainta et al., 2022). This SDS will be used in a clinical study about dementia to assess whether such systems (together with other tools) could be helpful to delay or prevent Alzheimer's disease.

Chapter 4

End-to-end EMPATHIC nutrition coaching chatbot

4.1 | Introduction

This chapter presents an end-to-end coaching chatbot for the nutrition domain, developed with the corpus gathered within the EMPATHIC project. We present several methods to allow open-domain dialogue techniques to be applicable in tasks typically tackled with rule-based dialogue systems.

The application of chatbots in healthcare and well-being is a rapidly growing research area. These conversational agents aim at improving some or many aspects of the users' health. For instance, they may be used to help diagnose, treat or prevent diseases like asthma (Kadariya et al., 2019) or cancer (Belfin et al., 2019; Siglen et al., 2022), monitor health-related parameters (Richards and Caldwell, 2017), prevent and treat mental health disorders (Saha et al., 2021; Abdulrahman et al., 2022; Callejas et al., 2020), or to provoke reflection (Kocielnik et al., 2018) and motivate healthy behaviour changes (Olafsson et al., 2020) to, e.g., increase the amount of fruit (Bickmore et al., 2013), control weight (Huang et al., 2018, 2021) or cease smoking (Dubosson et al., 2017; Alphonse et al., 2022). These tasks differ considerably from the classical application domains of dialogues systems (Tenorio-Laranga et al., 2019; Olaso et al., 2021), which have often been devoted just to providing some information or service to the user, such as checking the weather or restaurant booking, or just chit-chatting. From the perspective of the dialogue strategy, there is a big difference between providing information or simple services and trying to, for instance, provoke behavioural changes. In the latter there is no rush to complete any task; it is more important to calmly converse with the user and make them aware of their problems, obstacles and potential goals they may want to achieve. The objective of the health-related conversational agents being so different and delicate, the employed methodologies are also significantly different. Some works propose simple user interfaces such as multiple choice inputs for their system or just question-answering systems. Even among those which allow a dialogue via (spoken or text-based) natural language interface, the dialogue strategy is almost always implemented by a hand-crafted strategy or at least non fully data-driven approaches, such as finite state or frame-based management (Laranjo et al., 2018). On the other hand, some of the most promising chatbots in open-domain dialogue modelling are solely based on machine learning and are fully data-driven (Bao et al., 2022; Adiwardana et al., 2020; Roller et al., 2020), which is radically different from the aforementioned dialogue management approaches. In this chapter, we aim at bridging this gap, applying state-of-the-art AI techniques to develop a conversational agent capable of carrying out coaching sessions. We propose several improvements to adapt open domain dialogue modelling techniques to the needs of behavioural change models and being able to effectively apply a dialogue strategy from the perspective of planned behaviour (Ajzen et al., 1991).

Unlike rule-based conversational agents which are often implemented taking into account the consideration of experts, such as the one presented in Chapter 3, this time we try to learn and model directly their professional coaching strategy. To this end, we use the data acquired within the EMPATHIC project in Spanish, French and Norwegian, and also the English translations[1]. The fact that the corpus is multilingual already poses a major challenge. The deep learning system we propose in this chapter is only word-based, i.e., we try to model professional coaches without using any type of symbolic turn representation like dialogue acts or name entities. While this ensures that our approach can be easily replicated in other contexts and that it does not require expensive labelling, it hinders our task, especially when working with very low-resource languages like Norwegian. The second challenge to overcome is to build a conversational agent based on this technology that is capable of modelling complex conversations with long-term dialogue strategy like coaching sessions. This is especially difficult because deep learning approaches similar to the one proposed in this chapter have been mainly employed in very short dialogue tasks (Wolf et al., 2019), or in open domain dialogue modelling where the long-term structure of the dialogue has been completely ignored (Komeili et al., 2022; Adiwardana et al., 2020), even though even social dialogues have an underlying structure (Gilmartin et al., 2018).

To address these challenges we build upon a transfer learning approach, which has been adopted and proved to be successful in many dialogue modelling tasks (Wolf et al., 2019; Gunasekara et al., 2020; Komeili et al., 2022). This

[1]The EMPATHIC corpus is available at: http://catalog.elra.info/en-us/repository/browse/ELRA-S0414/.

methodology has turned out to be very handy and attractive in NLP in general, mostly due to big research teams releasing very large and pretrained neural LMs such as T5 (Ni et al., 2022), PaLM (Chowdhery et al., 2022), GPT-3 (Brown et al., 2020), GPT-2 (Radford et al., 2019) or BERT (Devlin et al., 2019). These transformer (Vaswani et al., 2017) neural networks have shown to have a great generalization ability, and can be fine-tuned and converted into up-and-running generative conversational agents. In fact, experts in coaching have pointed out that it is necessary to research the applicability of these giant neural network models in well-being-related tasks (Zhang et al., 2020a). However, these models are mainly developed for the English language. Thus, we propose to pretrain such neural LMs on big open domain text corpora available in many languages, such as OpensSubtitles or Wikipedia, and then fine-tune them on our smaller and multilingual EMPATHIC coaching corpus.

On the other hand, the main point to be taken into account is that the target dialogues are coaching sessions. These, in contrast to open domain conversations, have a long-term structure that cannot be ignored, and therefore needs to be learnt. The open-domain dialogue systems that we take as baselines often take a local dialogue history only as input, and therefore, are unable to keep long-term coherence. Thus, we propose two substantial methodologies to further adapt the models to our task. Our first improvement comes in the fine-tuning stage of the generative model. We propose to learn embeddings that indicate the model at which dialogue phase it is and which kind of coaching session is being carried out, so the generated responses are more coherent. Second, we propose to build an additional deep learning system that is used to take into account the whole history of the conversation, i.e. the dialogue history. We name it the Whole Dialogue History (WDH) system. The two models, i.e. the fine-tuned neural LM and the WDH system, cooperate to produce a response as suitable as possible in the coaching environment.

The fine-tuned neural LM acts as a generative model which produces a set of candidate responses given the partial dialogue history. Thus, we also refer to it as the short-term generative model. Ideally, if the training process has been successful, these candidates should be coherent short-term. They should take into account the current topic of the dialogue and the last information the user has provided. However, it may well happen that not all of the candidates are coherent long-term too. For example, the user and the agent might be talking about the user's dinner routine. Only taking into account that context, it might be reasonable to ask the user whether they take fruit at dinner time. However, the agent and the user might already have discussed about the fruit intake earlier in the dialogue in a way that it makes no sense to select this candidate as the final response. This is where the WDH system comes into play. It analyses (the contextual sentence embedding corresponding to) each turn in the dialogue history and computes a score measuring how suitable each generated candidate is. Following our example, this system would see that the agent and the user have already been discussing about fruit, so it would assign a very low score to that candidate, whereas other, more relevant and coherent candidates would be ranked much higher. Moreover, not only does the WDH system avoid repetitions, but it should also select, in general, candidates that follow more precisely the coaching dialogue strategy appearing in the corpus.

Additionally, we also show that the WDH system can be a powerful tool to understand and explain on what basis the decisions of the dialogue system are taken, which is an emerging concern in neural network-based systems. In fact, we show that the unsupervised representations learnt by the WDH system are closely related to conventional dialogue acts, but with the advantage that no costly annotations are needed to develop them.

Finally, we measure the impact of each of our proposals in terms of automatic metrics and human evaluation of the generated responses. We also provide an analysis of interaction experiments with our system in the four languages.

Thus, in summary, these are our contributions:

- We develop a novel coaching conversational agent by directly modelling professionals. Our proposal is trained purely on text, no dialogue acts are used, which makes it more general and applicable in other domains. Additionally, it is multilingual, i.e., it is capable of carrying out coaching sessions in English, Spanish, French and Norwegian.
- We describe a novel approach to improve the quality and relevance of the candidates the fine-tuned neural LM generates. On the one hand, we use scenario embeddings to specify which scenario the model should carry out. On the other hand, we explain how to build dialogue phase embeddings, a simple and powerful resource to enhance a more fluid dialogue flow.
- We propose and validate a novel mechanism, the Whole Dialogue History system, to take into account the whole dialogue context to ensure the coaching model is coherent long-term.
- We also show that this system can be a valuable tool in terms of explainable Artificial Intelligence; it allows to visually analyse on what basis the system takes its decisions. To this end, we compare the learnt representations with dialogue acts too.
- Finally, we discuss the potential impact and acceptance the described system would have on real users, based on automatic and human evaluation of the system.

The rest of the chapter is organised as follows. Section 4.2 presents the related work. Section 4.3 provides information about the dialogues to be modeled and gives a top-level overview of the proposed system. Section 4.4 describes the short-term generative model. There we present our proposals for the fine-tuning stage, i.e. how to train scenario and dialogue phase embeddings. Section 4.5 describes the WDH system in depth. In Section 4.6, we give more details about the experimental setup; including information about the pretraining and fine-tuning of the generative model and training details of the WDH system's modules. We also describe the automatic metrics and human evaluation procedures. In Section 4.7 we report the results of these evaluations. Finally, in Section 4.8 we present the visual analysis to better understand the decisions taken by the system and present a comparison with dialogue acts. We conclude with a discussion of our findings and with some final remarks in Section 4.9. The research presented in this chapter has been published in López Zorrilla and Torres (2022).

4.2 | Related Work

4.2.1 | Coaching conversational agents

Many diverse machine-assisted coaching systems, conversational agents and apps have been proposed in the last few years, forming a wide spectrum in terms of the employed technologies, implemented coaching methodologies and their area of application. In fact, besides healthcare and well-being, coaching systems with AI (coaching AIs, in short) have recently targeted other domains such as leadership (e.g. PocketConfidant[2]) or employee training (Luo et al., 2021). On the other hand, the coaching strategy also varies greatly. In this regard, it is important to mention that not all the coaching AIs in the market or the literature make use of an NLP interface, and even fewer incorporate a conversational agent. Some, like HabitBull[3] or Remente[4], just track the user progress in one or many habits, and provide them with data analysis, motivational videos or interactive guides to motivate them to reach their goals. Others, such as Quenza[5] or Coach.me[6], also act as mediators between users and professional human coaches, allowing face-to-face online coaching sessions. However, since our work involves the design of a conversational agent, we are most interested in coaching AIs that approach coaching as a conversation between the coach and the coachee, or that at least contain a dialogue module inside them. We first discuss some of the coaching chatbots that can be found in the market and then the works in the literature.

[2] https://pocketconfidant.com/
[3] http://www.habitbull.com/
[4] https://www.remente.com/
[5] https://quenza.com/
[6] https://www.coach.me/

4.2.1.1 | Coaching chatbots in the market

The so-called leadership bots, which aim at strengthening leadership skills, improving communication and developing self-confidence, have gained interest from many companies. Among this kind of coaching chatbots, we can find PocketConfident[7], ROCKY[8] or LEADx Coach Amanda[9]. According to their websites, PocketConfidant *engages individuals in personal, private and meaningful conversations to get unstuck, develop and reinforce human competence*; ROCKY provokes reflection routines asking *questions to help you reflect or prepare on your day*, which *vary every morning and evening and get more personalised over time thanks to machine learning behind*; and the LEADx Coach Amanda can provide leadership tips and answers to employee problems. It seems to perform some kind of user customization too: *because the Coach Amanda HR chatbot knows your personality, she'll personalise your manager training down to the sentence level.*

Naturally, there are also coaching chatbots designed for health care and well-being-related matters. Wysa[10] is one of the most notable chatbots in this regard. It has been awarded as the best health care app by ORCHA, and its effectiveness has been validated through clinical studies (Inkster et al., 2018; Sinha et al., 2022). Wysa is able to keep relatively long dialogues with a mix of natural language and multiple-choice input, and uses cognitive-behavioral techniques to reduce the levels of depression and stress; fight frustration, loneliness, or isolation; and improve mental health in general. It has also been the first AI mental health app to meet clinical safety standards, more precisely, the NHS UK's DCB 0129 Standard of Clinical Safety. Youper[11] is another app for mental health that includes a conversational agent. It is designed to help the users overcome anxiety and depression, *applying behavioral coping skills, and monitoring mental health symptoms.* Youper has been listed among the top ten behavioral apps in terms of real-world stickiness and engagement (Carlo et al., 2020). Last, it is interesting to mention Replika[12], which acts more as a companion chatbot than an actual coaching system. It is most popular among young people (its main users are aged between 18 and 25), and the authors claim that it can help to manage emotions, and reduce anxiety and sleeping issues.

[7] https://pocketconfidant.com/about/
[8] https://www.rocky.ai/chatbot
[9] https://leadx.org/hr-ai-chatbot-coach/
[10] https://www.wysa.io
[11] https://www.youper.ai/
[12] https://replika.ai/

4.2.1.2 | Coaching conversational agents in the literature

On the other hand, similar coaching systems have also been proposed in the literature. Even if they include a conversational agent, most of them are more focused on the tracking and goal-setting parts of the coaching rather than on motivational conversations. Additionally, the described dialogue engines are not end-to-end. For example, Fadhil et al. (2019) implement a chatbot, CoachAI, which acts as a task scheduler and tracker to promote physical activity. To this end, it includes a dialogue engine that guides the user through a series of steps to achieve their daily goal. However, in contrast to the dialogue model presented in this work, theirs is not a data-driven one. Instead, its core is a structured finite state machine. Another work that presents a coach AI to promote regular aerobic exercise is Mohan et al. (2020). The users can set their weekly goals, and the system keeps track of them, schedules exercises, and offers future goals depending on their progress. However, this chatbot does not use any complex conversational system to interact with the user: it relies on rule-based heuristics to drive the coach's reasoning. Gaydhani et al. (2020) present ongoing work on building a conversational agent to perform conversations about daily living to determine their degree of independence and assess them. While they mention their intention of building end-to-end dialogue models in the future, the described conversational agent relies on rule-based dialogue policies due to the lack of training data. Beun et al. (2017) describe an interesting coaching system for insomnia therapy made of two modules, a conversational agent and a module in charge of data acquisition, analysis and visualization. The dialogue system, however, is rather simple and it is based on multiple-choice inputs. Abdulrahman et al. (2022) present an embodied conversational agent to help students manage their study stress. They focus on understanding the users' goal and provide explanations about one of four predefined behaviour change procedures.

Closer to our domain of interest, nutrition, Casas et al. (2018) describe a coaching chatbot to help people improve their food lifestyle. It offers two goal possibilities to the user: reduce their meat consumption or increase the amount of vegetables and fruit they take. Besides tracking the user's situation with respect to their goal, it also offers the possibility to have guided conversations with the coaching chatbot about some predefined topics. The agent can provide the user with relevant images, videos and links to illustrate its remarks. The dialogue manager is built with the Chatfuel[13] service, which allows to manually design dialogues using a graphical interface. Interestingly, this chatbot was deployed in French rather than English. Maher et al. (2020) describe the results of a single-arm pre-post study carried out to test the efficacy of a virtual health coach focusing (Mediterranean) diet and exercise. They show that

[13] https://chatfuel.com/

the use of the Paola chatbot was able to reduce the weight of the participants and highly increase their Mediterranean diet score (Schröder et al., 2011). IBM Watson Virtual Assistant artificial intelligence software was used to design and implement the dialogue system, which allows the chatbot to converse with a natural language interface. This module, in contrast to our approach, is based on intent and entity detection to provide an appropriate response from a set of predefined options. Kettle and Lee (2021) deploy a SMS-based conversational agent using Rasa conversational AI software. This chatbot performs daily wellness checks, asks the participants to choose daily goals, and also allows them to freely talk to the system, which records their concerns but without responding. Last, Yan and Nakashole (2021) present a system that performs medical grounded question answering using machine learning. This is, they focus on solving users' doubts and concerns rather than on making them reflect through conversations.

Thus, our proposal is one of the very few works that describes a conversational agent capable of carrying relatively long dialogues with natural language input. Moreover, to the best of our knowledge, this is the first attempt to build a fully data-driven end-to-end coaching conversational agent.

Last, we would like to note that, after developing our research, some similar or related works have also been published. Mainly, Saha et al. (2021) describe an end-to-end model based on sequence-to-sequence neural networks to carry out motivational conversations for patients with depression.

4.2.2 | Multilingual or non-English end-to-end dialogue systems

There have been diverse attempts to build multilingual or non-English dialogue systems, yet the number of works describing end-to-end[14] dialogue models based on neural networks is rather scarce. Due to the lack of conversational data in many languages, some authors tackle this problem using automatic translation systems to convert the input message into English, then use an English chatbot to generate a response and finally translate it back into the original language (Ralston et al., 2019). Nonetheless, there are also a few examples of end-to-end neural dialogue systems trained directly in other languages. For example, (Chen et al., 2019) presented a chatbot in Chinese and a multilingual version of it in Chinese and English based on memory networks. Generative Adversarial Networks have been used to train multilingual response selection

[14] The term *end-to-end* is used with slightly different connotations by the machine learning community. In this work, with *end-to-end* we mean dialogue systems which produce a response based solely on the text corresponding to the dialogue history without using any kind of turn representations like dialogue acts or name entities.

systems (Sano et al., 2018) or response generation models in very low resource languages like Basque (López Zorrilla et al., 2020). Closer to our transfer learning approach, (Lin et al., 2020b) built a multilingual transformer capable of interacting in six languages other than English, trained on a multilingual version of the Persona-chat database.

Nonetheless, multilingual end-to-end dialogue systems are definitely a growing area of research. For example, many authors have targeted some cross-lingual and dialogue-related tasks, such as dialogue breakdown detection (Lin et al., 2020a), intent detection and slot filling (Bhathiya and Thayasivam, 2020), topic classification (Montenegro et al., 2019b), or language understanding (Müller et al., 2021).

More recently, there has been a notable rise in the amount of transformer-based LMs pretrained in many languages, such as GPT-3 (Brown et al., 2020) or mT5 (Xue et al., 2020). These allow to apply transfer learning (or even zero-shot learning) for dialogue modelling in other languages, Arabic for example (Fuad and Al-Yahya, 2022). Nonetheless, it is still worth mentioning that such models still underperform compared to their English counterparts, especially for very low-resource languages (Ebrahimi et al., 2021).

4.2.3 | Mechanisms to strengthen the long-term coherence of end-to-end dialogue systems

The task of keeping track of the dialogue context has been tackled since the early task-oriented dialogue systems. When the objective of the dialogue is to fulfill a goal of the user, it is necessary to know how close to that goal the dialogue is. To this end, goal-oriented dialogue systems have since then used a dialogue state tracking module. At first, a set of hand-crafted rules were normally used to track the dialogue state. Afterwards, with the advent of POMDPs (Williams and Young, 2007), probabilistic methods, such as dynamic Bayesian networks or attributed bi-automata (Serras et al., 2019b), gained popularity also for dialogue state tracking. Since the revolution of deep learning, a variety of approaches to track the dialogue state and/or to take into account the whole dialogue history have been proposed in task-oriented settings. Hybrid Code Networks (Williams et al., 2017), dialogue policies to specify actions plans (Hedayatnia et al., 2020), or, in general, pipelines that include a dialogue state tracking module have been proposed (Wang et al., 2022; Ham et al., 2020; Goel et al., 2019; Liu and Lane, 2018), among others. However, in all these cases the dialogue state and flow are controlled mainly or at least partially via dialogue acts extracted from the previous system and/or user turns. Therefore, all the methodologies require an annotated corpus (or hand-crafted rules) at some point to predict the dialogue acts. Our proposal does not.

Other works have tried to make use of the whole dialogue history in a similar manner to our approach, but often with different goals. Bayer et al. (2017) used a recurrent neural network on top of turn embeddings to improve the dialogue act prediction. Ganhotra et al. (2021) employ BERT embeddings of the dialogue history's utterances for Spoken Language Understanding. Tomashenko et al. (2020) integrate the whole spoken dialogue history using a variety of sentence embeddings for a semantic slot-filling task. Ortega et al. (2019) present a dialog act classification system on automatically generated transcriptions that combines convolutional neural networks and conditional random fields for context modelling. Liu et al. (2017); Wu et al. (2021b) also perform a dialogue act classification via a hierarchical deep learning model that takes into account the dialogue context. Wang et al. (2020) keep track of the dialogue history with a dual dynamic memory network and use it to make queries to a knowledge base in a task-oriented setting. In the context of the Alexa prize, Chi et al. (2021) use a module to smoothly switch from one topic to another, and also employ a neural entity linker to keep coherence throughout the dialogue. The work presented in Rodríguez-Cantelar et al. (2020) has been particularly inspiring for us. They propose to model the dynamics of turn embeddings to automatically evaluate the quality of the dialogue in the long run.

Last, we would like to underline that processing the whole (or a very large) dialogue history was previously explored with sequence-to-sequence (but not transformer) neural networks, such as hierarchical sequence-to-sequence networks (Serban et al., 2016; Li et al., 2022c) or deep RL chatbots (Cuayáhuitl et al., 2019). However, this is much harder to do with novel transformer-based dialogue systems, since their high (GPU) RAM memory requirements makes it infeasible to input long sequences of text, unless expensive dedicated hardware is available. Thus, our contribution alleviates this drawback of the powerful pretrained transformer networks.

4.2.4 | Conditioning the output of generative networks

In respect of our proposal to learn scenario and dialogue phase embeddings, we can find related works in the literature that condition the output of generative networks in several ways, and with several purposes. Some examples include using Reinforcement Learning to control the repetitiveness (Saleh et al., 2019) or politeness (Mishra et al., 2022) of a model or different approaches to assigning the model a fixed personality (Zhang et al., 2018; Huang et al., 2022). Conditioning the output of a generative model is very related to the area of stylised response generation too. For instance, (Gao et al., 2019) propose a chatbot that generates responses in a similar style to a non-conversational corpus. A similar approach has also been used to ensure a certain level of politeness in the system responses (Niu and Bansal, 2018; Firdaus et al., 2022), and to generate system turns that express given emotional status (Zhou et al., 2018; Firdaus et al., 2021). However, none of these works use a strategy similar to the one presented in this work, to the best of our knowledge.

4.3 | Overview

4.3.1 | GROW coaching dialogues and corpus

The corpus used to fine-tune the neural LMs is the EMPATHIC WoZ corpus previously described in Chapter 3 (Section 3.4). It is made of WoZ conversations with elderly people in Spain, France and Norway. All the dialogues are translated into English and into the rest of the target languages. Thus, the amount data is the same for the four languages. The dialogues are GROW coaching sessions (Whitmore, 1992) (see Section 3.2 in Chapter 3), which present a clear long-term structure. Mind that, in this work, we refer to coaching as behavioural change model which tries to make the coachees realise how they could improve their habits, not as an instructor or trainer who explicitly tells them what to do. Two different scenarios were designed for the WoZ interactions. First, we designed an introductory scenario, which was used to engage the user and make them feel comfortable in the interaction with the system. Secondly, a (partial) GROW session on nutrition was simulated.

A summary of the big numbers of the corpus are shown in Table 4.1 (more information about how this corpus was acquired can be found back in Chapter 3). Each dialogue was approximately 10 minutes long, which resulted in an average of roughly 29 turns per dialogue. The corpus is the same in the four languages, but some values differ due to the differences across languages. Even if the corpus has been annotated in terms of semantics and emotions, we will be using none of these in this study, since we are most interested in working with unlabeled data and developing end-to-end neural dialogue systems. Hence, our research should also be potentially more general and helpful to others too, because not always corpora are labeled neither do the labeled ones use the same label taxonomy.

Mind that the coaching dialogue structure and strategy do fall into any of the two broad categories conversational agents are often classified in: task-oriented, and open-domain. In the field of open-domain dialogue modelling there is no topic to talk about or task to carry out, the only goal is to generate appropriate and meaningful responses given a dialogue context. On the other hand, task-oriented dialogue systems are often developed to provide the

Table 4.1.: General statistics of the corpus of WoZ (translated) dialogues.

	Total (same for every language)
Number of dialogues	272
Number of system turns	7913
Avg. turns per dialogue	29.2

	Spanish	French	Norwegian	English
Avg. nb. of words per system turn	8.9	9.5	9.1	9.5
Avg. nb. of words per user turn	17.7	19.9	18.1	18.7
Total running words	208K	228K	213K	221K
Vocabulary size (system)	2.9K	3.1K	2.7K	2.4K
Vocabulary size (user)	7.9K	7.7K	7.4K	6.2K
Vocabulary size (aggregated)	8.7K	8.6K	8.1K	6.6K

user with some information or service they request as soon as possible, such as hotel booking. On the contrary, GROW coaching dialogues have peculiarities that do not allow us to easily classify them as transactional nor social. They are somehow task-oriented because there are some tasks to be completed, such as getting the user's objective or identifying which obstacles are not letting them fulfill their goal. However, the dialogue is not carried out in a conventional task-oriented manner. There is no rush to complete the task, and it is more important to calmly converse with the user and make them aware of their problems, obstacles and potential goals they want to achieve. In this sense, coaching is also related to open-domain dialogues, where there is no task and the only objective is to converse about different topics. However, coaching dialogues follow a clear and well-structured strategy. These differences in the properties of the dialogues are the main reason why novel approaches and techniques are needed to model them.

4.3.2 | System overview

We propose a dialogue system that can effectively model the described long-term dialogue strategies while dealing only with unlabeled text. A diagram of the system is shown in Figure 4.1. The proposed system is made of two important parts: a short-term generative model which creates some response candidates given a local dialogue history, and a global module that ranks the candidates according to their relevance given the whole dialogue history. We name this module the WDH system. Before getting into the details of both parts of the system and proposed novelties, we provide a top-level view of the system's functioning.

The short-term generative model is a fine-tuned neural language model, a

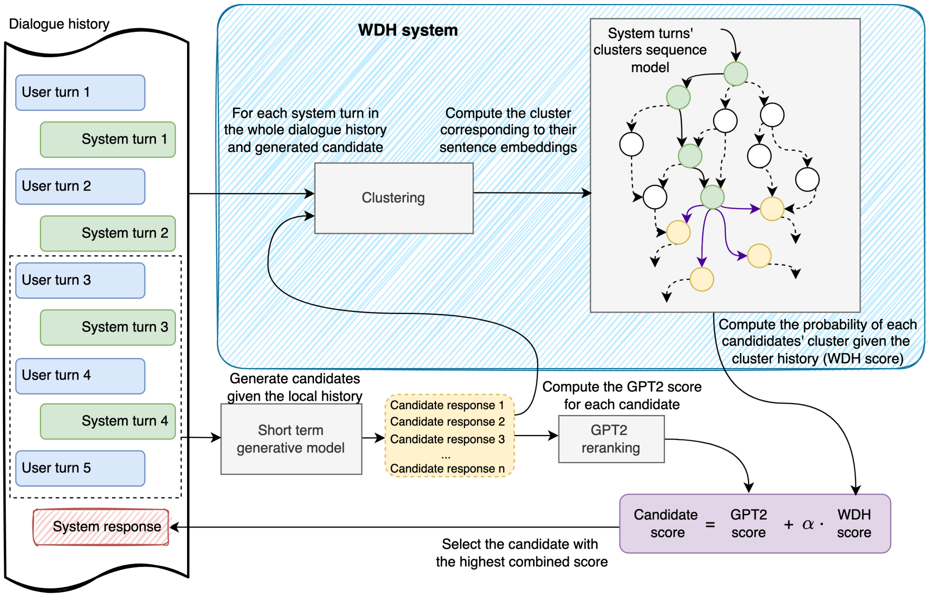


Figure 4.1.: The diagram of the proposed conversational agent.

GPT-2 transformer more precisely. It is trained in a transfer learning fashion to produce responses similar to what a coach would, given the local dialogue history made of the last turns of the conversation. The responses are generated via a top-K sampling decoding, which allows the generation of many different candidates given the same local context. However, since the local history that the model sees is not large enough to take into account the coaching strategy we aim at modelling, some of the generated responses are likely to be non-relevant or inappropriate. In Section 4.4, we propose some control mechanisms that can be included in the fine-tuning stage to alleviate this problem.

Nonetheless, we firmly believe that, in any case, it is necessary to take into account the whole dialogue history to successfully carry out complex dialogues like coaching sessions. If the model responses are produced only given the local context, repetitions might occur and in general non-consistent turns can appear very easily. Since with the current hardware it is not possible to include all the dialogue in the generative model's input due to memory limitations, we propose to build another system, the aforementioned WDH system. This evaluates how coherent each of the candidates proposed by the short-term generative model is, given the whole dialogue history. The main idea behind the WDH system is to model the long-term dynamics of the dialogue and include them in the decision-making stage. More precisely, we model the path the dialogues follow in the abstract semantic space of sentence or turn embeddings. To this end, the embeddings are grouped into clusters, with the assumption that turns inside each cluster should share some semantics and their role in the dialogue should not be too different. In fact, we later show (in Section 4.8.2) that there is a strong correlation between the cluster a turn has been assigned to and the corresponding dialogue act. Figure 4.2 shows a bidimensional projection of the turn embeddings and the resulting clusters. For the sake of simplicity, the number of clusters shown in the image is lower than the actual one. For instance, the purple cluster in the figure contains introductory turns, such as greetings or system presentations; the black cluster turns about food routine; the green one is travelling related, and the light blue contains turns about music.

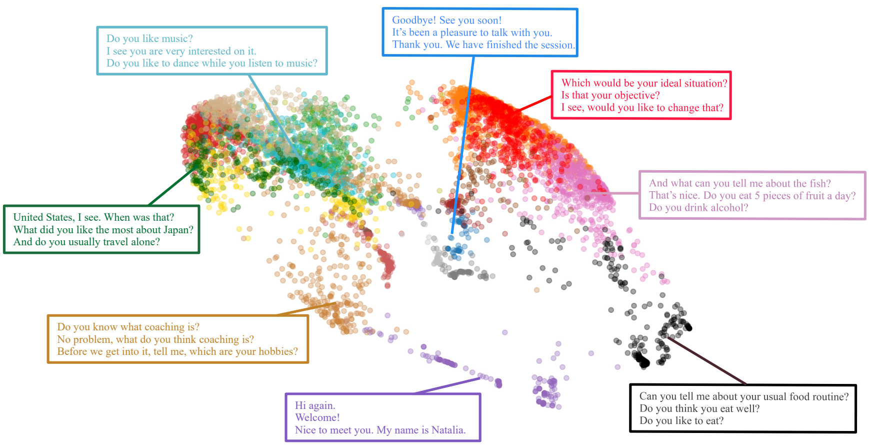


Figure 4.2.: A bidimensional projection of turn embeddings, coloured by the cluster they have been assigned to.

The turn embedding space seems to be organised enough to provide valuable information in the decision-making stage. We discuss this space more in-depth in Section 4.8. Note that, if we group each turn into a cluster, the dialogues in the corpus can be represented as sequences of clusters. Since the dialogues in the corpus follow certain patterns and strategies, these sequences should follow them too. We try to model the sequences of clusters and produce a system response that belongs to a cluster that is likely given a certain cluster sequence. A diagram of the whole system is shown in Figure 4.1. In the diagram, the GPT-2 score represents the score that the generative model assigns to each candidate via a reranking procedure (more about this score in Section 4.4.4).

In addition to its relevance in the decision-making stage, the WDH system can be employed to analyse and visualise the dialogues in the corpus. It also helps to explain and understand the system's decisions. We show it at the end of this study, in Section 4.8.

4.4 | Adding embeddings to the short-term generative model

In this section, we focus on the short-term generative model that can be seen in Figure 4.1. The neural network trained to produce candidate responses given a local dialogue history is a sequence-to-sequence transformer model (Vaswani et al., 2017). We start with a pretrained GPT-2 LM (Radford et al., 2019), and convert it into a response generation model applying transfer learning. In order to apply this methodology most effectively, it is key to exploit all the capabilities of the pretrained model. In the case of the GPT-2 transformer models, (Wolf et al., 2019) have already proved that adding information in form of additional embeddings to the input representation can be very useful.

Thus, taking their work as a baseline, we introduce two modifications to the input representation to improve its performance and adapt it to the needs of a motivational conversation model. As we explain in Sections 4.4.1, 4.4.2 and 4.4.3, this proposal consists in learning different embeddings to control the behaviour of the network in one way or another. A diagram of an example of a complete input to our transformer can be found in Figure 4.3, where only two input turns are shown for simplicity. In Section 4.4.1 we explain our baseline model (BL), in Section 4.4.2 the scenario embeddings (SC), and in Section 4.4.3 the dialogue phase embeddings (PH). Finally, in Section 4.4.4, we give further details about the generative model.

4.4.1 | Baseline (BL)

In our baseline, the input is represented with two parallel sequences of embeddings[15]. Since the first layer of the transformer takes only one sequence of vectors as input, the embeddings corresponding to each time step are added before being fed into the transformer. Let us describe the task of each embedding.

The first sequence corresponds to the word embeddings of the word/tokens of the last turns of the dialogues. In our experiments, we use the last five turns in the dialogue history, and concatenate them using special tokens as separators. This sequence of embeddings is the first row in the example shown in Figure

[15]Of course, positional encoding embeddings are also used throughout the whole work, but they are omitted here for the sake of simplicity, because they are common to almost all the transformer networks (Vaswani et al., 2017).

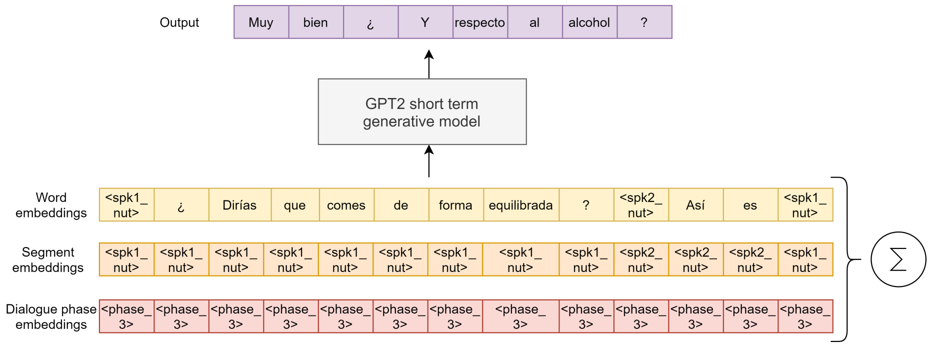


Figure 4.3.: An example of the proposed input representation to fine-tune the GPT-2 transformer network. The actual input to the transformer is the sum of all the embeddings in each time step. The segment embeddings (Section 4.4.2) indicate that the system is performing a nutrition dialogue, and the dialogue phase embeddings (Section 4.4.3) that it is the third phase of the dialogue.

4.3. The second one is used to segment the input into some categories. In our baseline, these segment embeddings indicate which input tokens correspond to the system's turns and which ones to the user's: <spk1> and <spk2>. This is the most straightforward way of applying transfer learning to convert a LM into a chatbot. While it might be interesting and appropriate for small conversations or just chit-chatting, in our case we need to ensure the overall robustness and coherence of the model if we want it to handle coaching sessions.

4.4.2 | Scenario segment embeddings (SC)

First of all, we have to take into account that our task requires the dialogue model to be able to carry out two different kinds of dialogues: an introductory dialogue and a partial GROW session about nutrition. Thus, we certainly need the option to specify which scenario to carry out to the model. It is also necessary that it does not arbitrarily jump from one scenario to the other. While we could train two different models for each scenario to avoid these issues, this approach would not allow each model to benefit from the conversational patterns appearing in the other half of the corpus. We consider that training a single model with the whole corpus in a multitask fashion is highly advantageous in this situation where the amount of data is not very high.

We propose to substitute the segment embeddings of the baseline with four different segment embeddings, in order to indicate which type of dialogue to carry out to the model: <spk1_int>, <spk1_nut>, <spk2_int> and <spk2_nut>. These now indicate who the user is, but also the scenario. For instance, in the second row of Figure 4.3 the embeddings indicate that the selected scenario is the nutrition one.

The scenario segment embeddings provide consequently a way of controlling the topic the system will talk about with the user: if at the beginning of a dialogue we feed the nutrition segment embeddings, the model will then talk about nutrition. If, conversely, we use the introductory segment embeddings, the machine will carry out an introductory conversation. Furthermore, note that this idea can be easily implemented in many other multitask frameworks other than ours.

4.4.3 | Dialogue phase embeddings (PH)

Finally, motivated by the empirical fact that the addition of (high-dimensional) embeddings is an appropriate technique to mix several pieces of information (Wolf et al., 2019), we decided to add a third set of embeddings: the dialogue phase embeddings. This is devoted to enhancing a dynamic progress of the conversation (without repetitions or loops) and a controlled ending. The phase embeddings tell the system at which point of the conversation it is, i.e., which proportion of the dialogue has been completed. For dialogues with lengths between 20 and 30 turns, we found that learning four dialogue phase embeddings was enough to lead to big improvements in terms of controlling the flow and limiting the length of the dialogue. Once a phase embedding is selected in function of the turn number and the desired length of the dialogue, it is added to all the input embeddings, as Figure 4.3 shows. Let us describe when each of the embeddings is used and which is its task, intuitively:

- The <phase_1> embedding is used in the first 20% of the dialogue. It tells the system that the conversation is starting, and thus when this embedding is added to each of the word embeddings, the system tends to produce opening sentences or greetings.
- The <phase_2> embedding is used from the 20% of the dialogue until the 50%. It corresponds to the rest of the first half of the dialogue, where the system tries to find an appropriate topic of conversation, asking the user some open questions.
- The <phase_3> embedding is used from the 50% of the dialogue until the 90%. In this phase, the system and the user mostly discuss about the topic they started in the second phase.
- Finally, the <phase_4> embedding is used within the last 10% of the dialogue. The system ends the discussion held in the previous phase, closes the conversation, and says goodbye to the user.

We also investigated and tested other smoother designs for these embeddings, such as using different embeddings per each turn. Nonetheless, we ended up discarding this option because our corpus (as many others) includes dialogues of very diverse lengths: sometimes the conversation ends at turn 15 whereas other times at turn 15 the user is still starting to talk about their nutrition habits. This can definitively lead to these embeddings not being trained precisely, and hence we opted for the relative phase embeddings approach.

This one, besides being more suitable in this case, also introduces the option of manually selecting the desired length of a conversation once the model is trained. This control, albeit not extraordinarily versatile, is enough to tweak the flow of the dialogue, which is very useful when dealing with end-to-end neural dialogue models, where controlling the system responses is often a very tough task.

4.4.4 | Decoding in the short-term generative model and GPT-2 candidate reranking

In this section, we give details about how the short-term model generates candidates, also referred to as decoding. We also explain how the GPT-2 score for each generated candidate is computed. In Figure 4.1, the described input embeddings can be associated to the input arrow to the short-term generative model block, the decoding refers to its output arrow. The GPT-2 score is shown in the bottom right purple block, where the total score for each candidate is computed.

Decoding details: Neural dialogue systems have been well known to generate too generic and repetitive. This problem has been tackled with many approaches, such as modifying the loss function (Li et al., 2016a) or using adversarial training (Li et al., 2017; López Zorrilla et al., 2021a). Lately, making use of a proper decoding procedure has proved to be essential for generative models to produce good quality non-generic responses (Kulikov et al., 2018; Golovanov et al., 2019). We adopt the nucleus sampling strategy (Holtzman et al., 2019) to prevent the system from generating dull or generic responses as much as possible. This technique consists of sampling only from a subset of tokens at each generation step. This subset is composed of the tokens whose cumulative probability is greater than or equal to a threshold. We set this threshold to 0.9. Additionally, before computing the aforementioned subset of candidate tokens at each generation step, we also apply some temperature (Ackley et al., 1985; Ficler and Goldberg, 2017) to the logits to control the diversity of the responses. For our application, we found that temperatures ranging from 0.65 to 0.8 led to very interesting responses. The value we set for the final experiments is 0.7.

Candidate reranking via the GPT-2 score: GPT-2 models are often trained both to generate candidates given a context and also to predict the next utterance given a set of possible ones (Wolf et al., 2019; Ham et al., 2020). More precisely, they are trained to predict whether a certain candidate is the correct response given the context or not. This binary prediction is done by a linear classifier that takes as input the hidden state of the transformer after processing the last token of the candidate. The output of this linear layer, i.e., the unnormalised probability of a candidate being the correct next utterance, is the GPT-2 reranking score. Intuitively, this score should be high when a candidate is informative and coherent with the local context; whereas non-relevant candidates or candidates containing grammatical errors should be assigned a low GPT-2 score. Mind that this idea of training a response generator and a response selector/discriminator jointly is very close to the philosophy of GANs (previously employed in the research presented in Chapter 2). In this case, however, the generator and the discriminator share all the parameters except for the last layer, which is much more stable, and the fake responses are not generated by the generator but sampled from the corpus instead.

4.5 | Reranking using the whole dialogue history (WDH)

Let us now present the WDH system in depth. It is composed of four modules. The first one's function is to produce sentence embeddings of each system turn. The second one carries out a dimension reduction of the previously computed sentence embeddings. The third one is a clustering module which assigns a cluster to the lower-dimensional embedding. These first three modules correspond to the *clustering* block in Figure 4.1. Finally, the last module produces an (unnormalised) probability distribution over all the possible clusters given the sequence of clusters that represents the dialogue history. This probability is the WDH score. In Figure 4.1, this fourth module is the block shown in the top right, and the resulting WDH score can be found in the bottom right.

4.5.1 | Contextual turn embeddings

There are several techniques to produce sentence embeddings, and each of them has shown strengths and weaknesses depending on the NLP task they have been employed in. In preliminary experiments, we compared generic sentence embedding methods, such as multilingual universal sentence encoders (Yang et al., 2019), sentence transformers (Reimers and Gurevych, 2020), or a weighted average of word vectors (Arora et al., 2019). However, the embeddings produced by the short-term generative GPT-2 models performed better in our experiments. The embeddings are the hidden state of the transformer after processing the last token of the sentence.

Using the short-term model for computing these sentence embeddings not only simplifies the system's pipeline, but it also provides additional benefits. First of all, since the model takes as input a partial dialogue history, the embedding it outputs contains information about both the user and system turns. This should definitely be considered an asset, because it allows packing the information of the user's turn in the system's contextual embedding. Otherwise, if a non-contextual embedding method were to be used, it would require computing and processing two different embeddings, one for the user and another one for the system. The second benefit of using a fine-tuned model is that the resulting embeddings are domain specific too, which is key for better performance.

Since we want the sentence embeddings to include the scenario and dialogue phase scenario information mentioned in Section 4.4, we include this information in the dimensionality reduction stage, described next.

4.5.2 | Dimensionality reduction

We apply a dimensionality reduction technique prior to the clustering method to avoid the curse of dimensionality (Bellman, 1966), since it is known to improve the quality of the clustering methods when these are distance or similarity-based (Steinbach et al., 2004).

We tried many methodologies such as PCA to carry out this dimensionality reduction, but we ended up building an autoencoder (Kramer, 1991). The reason for this is that, as aforementioned, we can easily incorporate supervision in the dimension reduction process, in a similar fashion to Le et al. (2018). The most straightforward way of training autoencoders is to optimise a recovery loss from a space with a lower dimension than the original space. In addition to the recovery loss, we minimise two classification losses, computed after a linear transformation of the low dimensional space. These correspond to the scenario and dialogue phase classification.

A summary of the structure of the autoencoder can be found at Figure 4.4. It takes as input a sentence embedding x, and after applying some non-linear layers with successively less output size, it ends up transforming it into $\boldsymbol{h}$, the low-dimensional representation of $\boldsymbol{x}$. This is the vector used at the clustering stage. Then additional layers transform $\boldsymbol{h}$ into $\boldsymbol{x}'$, the reconstructed version of $\boldsymbol{x}$. Thus, $\boldsymbol{h}$ contains as much information of $\boldsymbol{x}$ as possible. Furthermore, two linear layers perform two classifications from $\boldsymbol{h}$. After respective softmax normalizations, $\boldsymbol{y}_{scenario}$ and $\boldsymbol{y}_{phase}$ are produced, the probability distributions over the possible scenarios and dialogues phases. These two classifications ensure that the low-dimensional representation of the turn embeddings maintains as much information as possible about the scenario and dialogue phase, which are key properties of the turns.

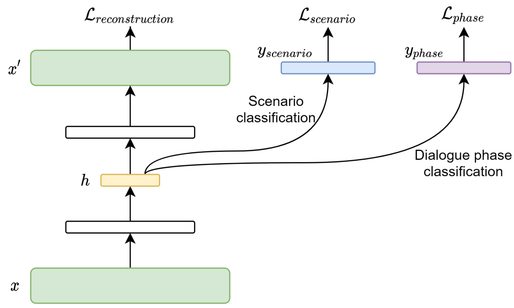


Figure 4.4.: A diagram of the proposed supervised autoencoder to reduce the dimension of turn embeddings.

Therefore, the training objective for this autoencoder is a combination of three losses, as shown in Equation 4.1. We tried some weighted sums of the losses instead of the unweighted one, but we found no improvement. The reconstruction loss is the euclidean distance between $\boldsymbol{x}'$ and $\boldsymbol{x}$. On the other hand, $\mathcal{L}_{scenario}$ and $\mathcal{L}_{phase}$ are cross-entropy losses for classification.

$$\mathcal{L}_{autoencoder} = \mathcal{L}_{reconstruction} + \mathcal{L}_{scenario} + \mathcal{L}_{phase} \tag{4.1}$$

4.5.3 | Clustering the turn embeddings

After the sentence embeddings corresponding to the system turns are computed and dimensionally reduced, we propose to group them into clusters, in an unsupervised fashion. Intuitively, system turns that are close to each other in the low dimensional embedding space should be semantically close, and they should also share key dialogue information, such as the scenario and dialogue phase.

There are many techniques to perform unsupervised clustering, for which the superior one is often a matter of the use case (Saxena et al., 2017). We tried and compared various methods, such as DBSCAN (Ester et al., 1996), Birch (Zhang et al., 1996), OPTICS (Ankerst et al., 1999) and K-Means (MacQueen et al., 1967). After an inspection of the turns inside each cluster, we decided to stick to the K-Means, because we found no improvement with the more sophisticated methods. Additionally, the K-Means algorithm provides two substantial benefits in our case. First, it takes the number of clusters as a parameter, which is very valuable for our application: we want enough clusters so that each of them represents a different state in the dialogue; but if the number is too large compared to the number of dialogues in our corpus, the task of learning the probability of the next cluster would not be feasible. A detailed analysis of the effect of the number of clusters is provided in Section 4.8.2. The second benefit is that, in contrast to many other clustering algorithms, it allows us to predict the cluster corresponding to a new sample in a very simple way. This is necessary when interacting with the system, because it is not possible to know the cluster a given turn corresponds to beforehand. Instead of having to train an additional classifier that learns to map from turn embeddings to clusters, the distance from the new sample to the cluster centroids can be measured, and the argument of the minimum will be the corresponding cluster.

4.5.4 | LEARNING THE NEXT CLUSTER PROBABILITY DISTRIBUTION

We cast the task of learning the next cluster probability as a sequence modelling problem. Given a set of vocabulary of m clusters $\mathcal{V} = \{v_1, v_2, ..., v_m\}$, and a sequence of clusters corresponding to a dialogue history $c_1, c_2, ..., c_n$, the objective is to compute the discrete probability distribution of each cluster being the next one in the sequence (Equation 4.2).

$$P(c_{n+1} = v_i \mid c_1, c_2, ..., c_n), \ \forall v_i \in \mathcal{V} \tag{4.2}$$

This task is very similar to a language modelling task, but having clusters instead of words. Therefore, we considered classical language modelling methodologies to tackle this problem. Even though N-gram models are simple models and have broadly been used to this end, recurrent neural networks, Gated Recurrent Units (GRUs) (Cho et al., 2014) more precisely, were our final choice. The main problem with N-grams is that they are based on the Markov Assumption, which assumes that the probability of the next cluster can be computed based only on the last few clusters. We really want the WDH system to take into account the whole dialogue history, so the N-gram models were finally discarded. On the contrary, GRUs process the whole cluster sequence. We show at the end of Section 4.8.3, indeed, taking into account the whole sequence is highly beneficial, since GRUs outperform the N-gram models in terms of accuracy and top N accuracy.

The objective function used to train the GRU was the negative log-likelihood at the cluster level:

$$\mathcal{L}_{GRU} = \frac{1}{|\mathcal{C}|} \sum_{\boldsymbol{c} \in \mathcal{C}} \frac{1}{|\boldsymbol{c}|} \sum_{i=1}^{|\boldsymbol{c}|} -\log P(c_{i+1} \mid c_1, c_2, ..., c_{i-1}) \tag{4.3}$$

where $\mathcal{C}$ is the training corpus made of sequences of clusters $\boldsymbol{c}$, each of them corresponding to a dialogue. The probability of c_{i+1} being the next cluster given the partial sequence $c_1, c_2, ..., c_{i-1}$ is computed by a softmax normalization on top of a linear classifier given the last hidden state of the GRU after processing the cluster sequence.

4.5.5 | Computing the total score for each candidate

The GPT-2 score and the WDH system's score are fused in a simple way. The total score is a weighted both scores, as shown in Equation 4.4. This Equation is also shown in the bottom right of Figure 4.1. A detailed analysis of the chosen value for the hyperparameter α and its role in the system's performance is shown in Section 4.7.1.

$$\text{Total score} = \text{GPT-2 score} + \alpha \cdot \text{Cluster score} \tag{4.4}$$

4.6 | Training details and experimental setup

In this section, we give more details about our implementation and introduce the experimental setup that was used to produce the results we present and discuss in Section 4.7. According to our proposal, we train and compare six models:

- The baseline model (BL). This refers to the model presented in Section 4.4.1, without any reranking. I.e., here we only use the short-term generative model, without the WDH system. This generates just one candidate, which will be used as the system response.
- The baseline model with scenario embeddings (BL+SC). In this model, we add the scenario embeddings (Section 4.4.2) to the baseline model.
- The baseline model with dialogue phase embeddings (BL+PH). Here we add the dialogue phase (Section 4.4.3) to the baseline model, but without the scenario embeddings.
- The full generative model (FM). This one includes both the scenario and

dialogue phase embeddings. But still, there is no reranking, i.e., it outputs the first utterance it generates.

- The full model with just GPT-2 reranking (FM+RR). In order to check the influence of the WDH system, we first include a reranking process with only the GPT-2 score. We generate and rank 10 candidates.
- The full model with both the GPT-2 reranking and the WDH reranking (FM+WDH). This model includes all our proposals. The number of candidates is the same as in the FM+RR model, 10.

We first explain the process of pretraining the GPT-2 neural LMs in Spanish, French and Norwegian. Then we get into the fine-tuning details of these models with the EMPATHIC corpus. Finally, we give details about the WDH system and introduce the experiments and the evaluation procedures.

4.6.1 | Pretraining procedure

We are dealing with a multilingual corpus in Spanish, French, Norwegian and English. However, most of the big pretrained neural LMs are only available in English. After some preliminary experiments using multilingual pretrained transformers such as XLM (Conneau and Lample, 2019), we found that fine-tuning these did not result in great dialogue models. Thus, we ended up pre-training GPT-2 models from scratch in Spanish, French and Norwegian, and using the pretrained and freely available GPT-2 models in English.

There are four different GPT-2 architectures (Radford et al., 2019), which mainly differ in the number of layers and their size. We selected the *small* GPT-2 transformer architecture for all of our experiments in Spanish, French and Norwegian, which has 124 million parameters. This selection was made to meet two important criteria: the model should be large enough to be capable of learning our task, but small enough to fit into standard GPUs and be pretrained in a reasonable amount of time. As for English, we compared the *small* model with the *medium*, which has 324 million parameters. The latter one worked much better already since the first experiments, as shown in Section 4.7.1. Therefore, unless it is mentioned explicitly, the results for the English system were achieved with the *medium* GPT-2.

We used the Spanish, French and Norwegian versions of Wikipedia and OpenSubtitles (Lison et al., 2019) to pretrain each language model. The reason for choosing these corpora is that both are available in the target languages, and that both include valuable information which could improve the final performance of the coaching dialogue model. Wikipedia contains information about millions of topics, and OpenSubtitles is made of conversations mainly, which hopefully helps the model learning dialogue skills. Since the amount of data in Norwegian was much lower, we also included a fraction of the Norwegian version of the OSCAR text corpus (Ortiz Suárez et al., 2020). OSCAR is a subset of Common Crawl, and thus it is made of web-scrapped text from the Internet. Mind that, therefore, this corpus may not be as related to our task as OpenSubtitles or Wikipedia. Table 3.13 in Chapter 3 shows a summary of the data used for pretraining after cleaning lines containing irrelevant characters, urls, and so on.

We first trained a BPE tokeniser (Sennrich et al., 2016) in each language with this data. This tokeniser is used during the pretraining and fine-tuning steps. We selected a vocabulary of 10K subwords in each language. This number is lower than the pretrained tokeniser in English, which has a vocabulary of around 50K subwords. Using a reduced vocabulary size also reduces memory consumption and training time.

We then trained each GPT-2 model from scratch, throughout two complete epochs on each dataset. We set the maximum number of input tokens to 512, which we consider enough since it allows us to afterwards include 5 turns of dialogue history in the fine-tuning step. We used the ADAM optimiser (Kingma and Ba, 2014) with a linearly decaying learning rate from 1e-5 to 5e-4, and a batch size of 4, the maximum that fitted in our GPU. We clipped the gradients at a maximum absolute value of 1. Each training procedure took around 2-3 weeks in total to be completed in a single Nvidia Titan Xp GPU. Besides in this application, the trained LMs were also used in the EMPATHIC NLG (Section 3.5.3, Chapter 3).

4.6.2 | Fine-tuning the GPT-2 generative model on the EMPATHIC corpus

After pretraining the LMs, we fine-tuned them on our dialogue corpus to convert them into dialogue models. We fine-tuned each model with combinations of the three input representations explained in Section 4.4, for comparison purposes. We trained the baseline, the baseline with scenario embeddings, the baseline with dialogue phase embeddings, and finally the full generative model with both scenario and dialogue phase embeddings. All the systems were also trained to predict the end of the dialogue. To this end, an end-of-dialogue token was inserted in the last system turn of every dialogue. The number of turns selected for the local dialogue history was five: three user turns and two system turns. To measure the effect of not pretraining, we also trained a GPT-2 model from scratch with our corpus in Spanish.

We split the data into train (85%) and test (15%) partitions. These proportions were kept when splitting the original dialogues in each language and also the dialogues translated from the remaining two languages. Each partition also contains the same number of introductory and nutrition dialogues. Since most of the users interacted with the system in both scenarios, we also made sure that all the dialogues corresponding to a given user only appeared in one of the partitions.

Training details: Following previous works, we optimise a linear combination of two loss functions during the fine-tuning step (Radford et al., 2018; Wolf et al., 2019; Budzianowski and Vulic, 2019): the language modelling loss and the next turn prediction loss. The second one also enables the possibility of using the GPT-2 score described in Section 4.5.5. We set the weight of the LM loss to be the double of the next turn prediction one. We used 10 candidates for the next turn prediction loss, the actual ground truth, 3 system turns from the previous dialogue history (but not appearing in the local history), 3 system turns that occurred later in the dialogue, and 3 random turns sampled randomly from the training set. The combined loss function was minimised throughout 4 epochs via the AdamW optimiser (Loshchilov and Hutter, 2017). The learning rate was linearly decreased from 6e-5 to zero, gradients were clipped at their absolute value of 1 and a weight decay of 0.01 was used. We could only fit one training example at a time in the GPU during the training process, but we still experimented with greater virtual batch sizes, accumulating the gradients for some steps. We found that a virtual batch size of 4 led to the most consistent results.

Additional details: The desired length of the dialogues was fixed to 20 system turns. As for the systems that use reranking (BL+RR and BL+WDH), in the decoding step we generated and ranked 10 candidates.

4.6.3 | WDH system details

As mentioned in Sections 4.5.1, 4.5.2 and 4.5.3, we considered many strategies and algorithms to compute the sentence embeddings, dimension reduction and clustering. The final choice for each module in the pipeline was decided after an inspection of the resulting clusters. We checked that the turns grouped in the same clusters were in fact semantically close, and that it would make sense to use them in similar dialogue contexts. Finally, the baseline short-term generative model was used to produce sentence embeddings, a supervised autoencoder for dimension reduction and the K-Means algorithm for clustering. In Section 4.8, we provide a more insightful analysis of the results of the clustering pipeline.

Let us now give the details about the architecture and hyperparameter selection in the WDH modules. The turn embeddings were computed with the BL model. As for the autoencoder, its input and output size is the same as the turn embeddings. In the case of the Spanish, French and Norwegian systems this was 768, and in the case of English 1024, due to the use of the *medium* GPT-2 architecture instead of the *small* one. The autoencoder's encoder and decoder are symmetrical. They are made of three fully connected layers of sizes 200, 50 and 5. The hyperbolic tangent was used as the activation function. Thus the low dimensional embedding space is of dimension 5. The two classification layers take as input this embedding and linearly classify the scenario and dialogue phase. The autoencoder was trained with the sentence embeddings of the training set of the corpus during four epochs via the Adam optimiser. A batch size of 4 and a learning rate of 1e-4 were used.

As for the clustering, the Elkan's variation of the K-Means algorithm was used (Elkan, 2003), with the euclidean distance in the low dimensional embedding space. After analysing its impact on different metrics, the number of clusters was set to 60. As shown in Section 4.8.2, this value represents a nice compromise between a balanced number of turns per cluster and the performance of the WDH system at the next utterance classification task (which we introduce next in Section 4.6.4). Additionally, it is also a value that permits a good mapping from cluster index to dialogue act, as explained in Section 4.8.2, where the correlation between the clustering and dialogue act classification is explored.

Once the clustering pipeline was fixed and trained, we proceeded with the cluster sequence modelling experiments via GRUs. The hyperparameters of the recurrent neural network were tuned in a development partition within the training set to preserve the train-test independence. The input size of the cluster sequence modelling GRU was set to 5. Namely, each cluster was represented by a five-dimensional vector. We tried initialising them in terms of the turn embeddings but got no improvement, so they were randomly initialised and learnt in the process. Two GRU layers of hidden size 60 were then used, followed by a softmax layer of size 60 to output the probability distribution over the possible 60 clusters. The GRU was trained during 3 epochs via the Adam optimiser, with a batch size of 4 and a learning rate of 1e-4.

4.6.4 | Automatic and human evaluation

Once the models were trained on the train partition of the corpus in all the languages, we evaluated each of them according to their responses in the test partition. On the one hand, we computed some automatic metrics to measure the similarity of the generated response to the ground truth and the accuracy of the reranking methodologies. On the other hand, experts in coaching compared the responses of different models and selected the most appropriate one. Finally, these experts also interacted with the best model and evaluated the resulting dialogues.

Automatic metrics: Automatic evaluation of dialogue models is a very active and complex research area. In the last few years, many authors have been seeking metrics that measure the quality of the responses, and that correlate as much as possible with human evaluation in terms of, e.g. relevance, semantical appropriateness or informativeness. On the one hand, there are word overlap metrics such as BLEU (Papineni et al., 2002), which measure how the generated response and the ground truth resemble at the word level. More recently, this similarity has also been measured via word or sentence embeddings (Zhang et al., 2019). There are even authors who propose unsupervised metrics which do not rely on ground truth responses (Mehri and Eskenazi, 2020; Nedelchev et al., 2020).

In this chapter, we use two of the official metrics proposed in The Conversational Intelligence Challenge 2 (Dinan et al., 2020): the accuracy at selecting the correct next utterance among a set of 10 candidates or next utterance classification accuracy, and the F1 score between the set of words of the response generated by the system and the ground truth. Additionally, we include the BLEU score as an additional measure of how similar the ground truth and the generated response are.

The next utterance selection accuracy is particularly interesting in our case, since much of our work focuses on improving the selection of good responses given a set of candidates. Note that this metric does not directly evaluate the response generation task. Instead, it focuses on the ability of the different models on selecting the correct response from a set of candidates sampled from the corpus. This selection is done via the aforementioned GPT-2 reranking modules (Section 4.4.4), and also with the WDH system in the case of the FM+WDH model. In any case, this metric should be a nice indicator of the systems' performance when interacting with real users. The only difference is that in that case the set of candidates is not sampled from the corpus, but generated by the generative model. In the original metric of the Conversational Intelligence Challenge 2 (Dinan et al., 2020), the set of candidates is made of randomly sampled responses entirely. However, in our case, 6 out of the 10 candidates are system turns that are part of the same dialogue, which makes the task more challenging since many candidates are probably closer semantically. Among the remaining candidates, 3 are randomly sampled from the corpus, and the last one is the ground truth.

Human evaluation: On the other hand, we carried out two series of human evaluation processes to validate our methodologies in the task of coaching. Since coaching is not a simple topic and expertise is needed to evaluate how good a system would be in this area, the selected human evaluators were the same professionals that carried out or participated in the Wizard of Oz experiments to acquire the corpus. This is very important, because it may well happen that a non-expert human considered that the interaction with the system has been good for example, but that would not ensure that the system is actually performing any type of coaching.

In the first series of evaluations, we evaluate the response quality of the different versions of our model. In order to measure the impact of each proposal, we incrementally compare them, through a sequence of pairwise comparisons. We start comparing the BL with the BL+SC to check the influence of the scenario embeddings. We do the same with the dialogue phase embeddings, comparing the BL with the BL+PH. Then the influence of adding both embeddings is measured via the comparison of the BL with the FM. Afterwards, we analyse the impact of the candidate reranking with two comparisons, FM vs. FM+RR and FM+RR vs. FM+WDH. Finally, to give a grasp of the absolute quality of the responses, we compare the FM+WDH with the ground truth (GT hereinafter), i.e., the human response found in the test set. Note that reason behind the choice of carrying out these incremental comparisons pairwise instead of, e.g. via a Likert-score based evaluation per model, is that the results would be harder to compare, due to the potential evaluator bias when selecting the score in the Likert scale, as discussed in (Li et al., 2019). Some annotators might be more generous and while others might tend to stick to more neutral responses. That bias is reduced in a pairwise setup, because the evaluators should only select which answer is better (or whether they are equal), but not to what extent. The biggest drawback of pairwise comparisons is that it might be difficult to aggregate the results if the comparisons are not incremental. In our case, this only happens with the BL vs. BL+SC and BL vs. BL+PH. This is why we also perform the BL vs. FM comparison.

In the second series of human evaluations, we focus on the usability and potential impact of the best system. We asked each coaching expert to interact with the model in each scenario and then to fill out two questionnaires. Even though the system is planned to be used with a spoken interface, it was tested on a text-based interface to avoid potential biases created by third modules. The first questionnaire is the Chatbot Usability Questionnaire (CUQ) (Holmes et al., 2019). This novel questionnaire is similar to the classical SUS (Brooke, 1996) for human-computer interfaces, but adapted to the particular domain of chatbots, taking into account their peculiarities. On the other hand, the second questionnaire is based on the standardised questionnaire AttrakDiff (Hassenzahl et al., 2003). AttrakDiff was designed to measure the user experience in human-machine interaction in four axes: pragmatic attractiveness and three hedonic qualities. Esposito et al. (2018) adapted this questionnaire for the evaluation of virtual agents. In this study, we use the questionnaire related to one of the hedonic qualities axes, to the hedonic quality stimulation or feelings, more precisely. It aims at identifying the feelings that may arise in the user when interacting with the system. This is particularly important to assess the usability and potential consequences of a health-care-related conversational agent. A system that gives rise to negative feelings in the user would never be acceptable, for instance. We refer to this questionnaire as Hedonic Feelings Questionnaire (HFQ) (Hedonic Feelings Questionnaire). Both questionnaires can be found in Tables C.1 and C.2 in Appendix C. The responses were arranged in a five-level Likert scale ranging from *Strongly agree* to *Strongly disagree*. Since both questionnaires ask about positive qualities of the system in even questions and about negative in odd ones, a score for each questionnaire can be easily calculated. A score of 100 would be obtained if an evaluator would *Strongly agree* with all the positive questions and *Strongly disagree* with all the negative ones, and a 0 in the opposite case.

4.7 | Results

In this section, we present and discuss the automatic and human evaluations of our proposal.

4.7.1 | Automatic evaluation

Let us now show the results of the automatic metrics. We start discussing the performance of the models in terms of the next utterance selection accuracy among 10 candidates (Table 4.2), since it provides the most consistent results across all languages. As aforementioned in Section 4.6.4, these three candidates are not generated by the short-term model, they are sampled from the corpus instead. Afterwards, we provide results about the quality of generated responses, in terms of F1 and BLEU scores.

Table 4.2.: Next utterance classification accuracy among a set of 10 candidates obtained by all the models in the four languages in the test partition of the corpus.

	Next utterance classification accuracy			
	English	Spanish	French	Norwegian
BL no pretraining	-	0.251	-	-
BL small	0.461	-	-	-
BL	0.482	0.374	0.404	0.350
BL+SC	0.488	0.379	0.402	0.343
BL+PH	0.488	0.388	0.417	0.366
FM	0.494	0.401	0.421	0.375
FM+RR	0.494	0.401	0.421	0.375
FM+WDH	**0.518**	**0.412**	**0.435**	**0.388**

Next utterance classification accuracy: First of all, we can see that there is a big gap between not pretraining the baseline and pretraining it in the Spanish model. Given this big gap, we did not consider trying with non pretrained baselines in the other languages. In English, there is also an improvement if we consider the medium GPT-2 architecture (BL) or the small one (BL small). Therefore, the rest of the experiments were carried out with the medium architecture.

Including the scenario embeddings does not seem to influence this accuracy as much as including the dialogue phase embeddings do. This is probably due to the nature of the candidates to be ranked: among the 10 candidates, 6 are system turns of the same dialogue. This can probably confuse the BL and the BL+SC more than the BL+PH model, because the candidates share the scenario and potentially the topic, but the phase embeddings might be able to capture that they are out of position given the status of the dialogue. The FM further improves over both BL+SC and BL+PH, proving that combining both embeddings leads to better performance. In respect of the FM+RR, note that the next utterance accuracy is the same as the FM model. This is because these models are essentially the same; they only differ in the decoding stage: the FM generates just one response, whereas the FM+RR generates some candidates and then selects the best according to the GPT-2 score. But in this case, since the set of candidates is given, there is no big difference. Finally, the full model with the WDH reranking method obtains the best results across all the languages. This clearly shows that the proposed reranking method helps to improve the candidate selection criteria. Consequently, it also reinforces our initial hypothesis that it is necessary to process the whole dialogue history to improve the overall quality of end-to-end neural dialogue systems. This is even more critical when no dialogue acts or dialogue state tracker are being used; and also when the application, such as coaching, requires the dialogues to be well structured.

Let us further show the influence of the WDH system in the next utterance classification accuracy. As explained in Section 4.5.5 and Equation 4.4, the total score for a response candidate is a weighted sum of the GPT-2 score and the WDH score, where α is the weight of WDH score. We performed a grid search with values of different orders of magnitude for this weight. The results are shown in Figure 4.5.

In general, the behaviour of the metric as a function of the cluster score is the expected one. When the next cluster score is 0, the model is equivalent to the FM+RR, and so the next utterance selection accuracy is the same shown in Table 4.2 for the FM+RR models. On the other hand, the accuracy peaks when the next cluster score is between 0.1 and 0.3. The maximum values (the ones corresponding to the optimal WDH score per language) are the ones shown in Table 4.2. If we increase the weight of the cluster score way more than its optimal value, the accuracy decreases drastically. This means that the WDH system should be used only as an addition to the GPT-2 score, not as a substitute. The reason for this is that the GPT-2 score takes into account properties that

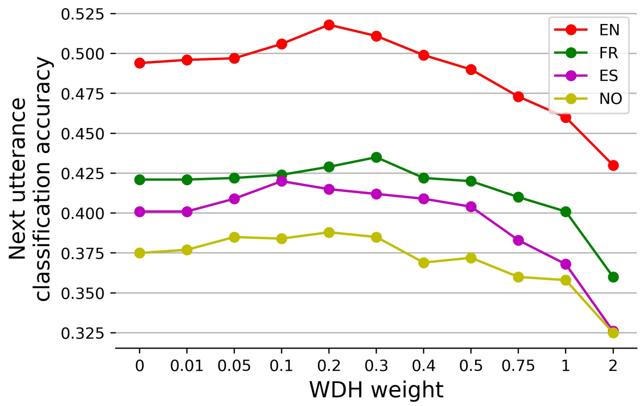


Figure 4.5.: Next utterance selection accuracy depending on the WDH score's weight. Note that the x-scale is equally spaced between the tested values.

the WDH does not, and vice versa. The GPT-2 score focuses on short-term coherence, but also on grammatical appropriateness, since it evaluates the turn embedding. On the other hand, the WDH is not aware of the system turn itself, only of the cluster it belongs to. Therefore it may assign a very high score to a candidate that belongs to a very relevant cluster given the dialogue history, but is grammatically incorrect, for example.

Not only does the next utterance accuracy reveal differences between models, but also across languages. There seems to be a big correlation between this accuracy and the quality of the pretrained language model. First, the English models outperform the models in lower-resource languages. If we then compare the remaining three languages, Spanish and French are one step ahead of Norwegian. The GPT-2 model in English was pretrained and released by Open AI. 40 GB of cleaned and processed data was used. In comparison, we only used 10GB, 7GB and 5GB used to pretrain the Spanish, French and Norwegian models, respectively. Additionally, we also believe that the OSCAR corpus used to increase the amount of data in Norwegian is not as beneficial for our domain as Opensubtitles and Wikipedia, due to its nature. Since it is made of web-scrapped text from the Internet, it may contain many sentences that are not related to our task at all, hindering the fine-tuning procedure.

Word overlapping metrics: While the next utterance classification score seems to be very aligned with the expected behaviour of our proposal, the F1 and BLEU score do not seem to be that correlated. Table 4.3 shows the obtained results. Nonetheless, there are still some conclusions to be made.

First of all, the two metrics behave in a very similar manner, which makes sense because both are measures of how the produced system response resem-

Table 4.3.: F1 and BLEU scores obtained by all the models in the four languages in the test partition of the corpus..

	English		Spanish		French		Norwegian	
	F1	BLEU	F1	BLEU	F1	BLEU	F1	BLEU
BL no pretraining	-	-	0.198	0.096	-	-	-	-
BL small	0.303	0.143	-	-	-	-	-	-
BL	0.305	0.139	0.229	0.109	0.250	0.119	0.297	0.147
BL+SC	0.299	0.140	0.248	0.122	0.257	0.124	0.274	0.123
BL+PH	0.283	0.132	0.259	0.130	0.241	0.111	0.297	0.141
FM	0.302	0.143	0.272	0.145	0.259	0.122	0.309	0.149
FM+RR	**0.315**	**0.150**	0.288	0.150	0.275	**0.135**	0.317	0.158
FM+WDH	0.303	0.142	**0.296**	**0.159**	**0.276**	0.131	**0.322**	**0.164**

bles the ground truth sentence found in the corpus. Second, if we compare the results of the different models, we can see that including the scenario or phase embeddings does not consistently yield better results. There does not seem to be any difference between the small and medium models in English. On the other hand, not pretraining the baseline in Spanish again produces worse results. Interestingly, applying a reranking process does improve the result in both metrics across the four languages. This shows that the reranking methodologies play an important role in our system, and that are capable of selecting the responses which are closer to the ones produced by human experts. Finally, we would also like to mention that in this case the results on different languages should not be compared too in-depth, because the four languages are morphologically different and therefore the differences might well be due to language particularities instead of performance discrepancies. In any case, many authors have argued that word overlapping metrics are not highly correlated with the actual quality of the responses (Liu et al., 2016), because a response that does not share any words with the ground truth reference could indeed be completely appropriate. Thus, we now provide a human evaluation to further validate our proposals.

4.7.2 | Human evaluation of the responses

The quality of the generated responses was measured by coaching experts. Four different experts per language participated in this evaluation. They compared pairs of responses from different models. Per each language and model pair, 40 pairs of responses were ranked twice. Every evaluator assessed the same number of instances per model pair, where each instance consisted of a local dialogue history made of the last 5 turns, and two possible continuations for the system. The dialogue histories were different for each model pair in order not to bias the evaluators. Four options were presented to the evaluators. According to their criteria in the context of coaching and the project, they had to select whether the first response was better, the second one was better, both of them were equally valid to continue the dialogue, or none of them was acceptable. We considered using more fine-grain level metrics, such as the ones used in Xu et al. (2017), but we decided to stick to the simpler approach because: 1) since the evaluators are experts, they should be able to weight the different aspects of the responses and reckon which is more appropriate for the task, 2) the models to be compared should not vary drastically in the style of the responses, because they are different versions of similar methodologies, and 3) it is, therefore, more cost-efficient; the additional costs would not compensate the potential benefits of a more detailed evaluation, owing to the aforementioned reasons. Additionally, in Section 4.7.3, we perform a detailed evaluation of the best model, which shows the strengths and weaknesses of our conversational agent in depth.

Before presenting the results, we would like to show some examples of the evaluation task to provide a feeling of the kind of dialogues that were carried out. Tables 4.4, 4.5, 4.6 and 4.7 show dialogue contexts and the continuation of the FM+WDH system and the ground truth (GT) in English, where, respectively, the two responses were valid, the GT response was better, the FM+WDH system's response was better, and none of the responses was good enough, according to the evaluators.

Table 4.4.: Evaluation example in English where both the GT and FM+WDH system's responses were valid, according to the evaluators.

Dialogue history	**U**: I'm from Poland. It was very interesting to spend time [laugh] especially in Warsaw [laugh] but a lot of time at the airport. **S**: All right. I'm really jealous. And when was that? **U**: On Sunday, a week ago. **S**: Do you usually travel accompanied? **U**: Yes, but sometimes alone too. But I almost always make friends there, yes.
FM+WDH response	Who organizes the trips?
GT response	Imagine that you are in Poland, in these busy streets. How do you feel? How do you imagine it?

Table 4.8 shows the results of the comparison of the models and the ground truth, combined in all the languages. Additionally, Tables 4.9, 4.10, 4.11 and 4.12 show these results divided by language; in English, Spanish, French and Norwegian, respectively. Binomial tests of significance were carried out in the global comparison of the models shown in Table 4.8, since it contains more samples and it is therefore more appropriate. The p-value was computed taking into account the decisive comparisons only: *A is better* versus *B is better*.

Table 4.5.: Evaluation example in English where both the FM+WDH system's response was better than the GT, according to the evaluators.

Dialogue history	**U**: Yes. **S**: and I detect that you intend to accomplish. **U**: That's right, because it's hard to lose weight over the years. It's very difficult, it costs a lot more. You have to be very constant, a lot of willpower, and that is also lost over the years. **S**: and create habits **U**: Create habits, that is, you have to create habits, once the habit is created, it's easier.
FM+WDH response	I understand you're talking about habits.
GT response	Unambiguous good.

Table 4.6.: Evaluation example in English where both the GT response was better than the FM+WDH system's continuation, according to the evaluators.

Dialogue history	**U**: Yes yes yes I like it. I'm rather sweet as they say that salty so I like pastries and desserts very much. **S**: Alright. It's good that you're trying to be careful at least at home. **U**: [laugh] That's right. **S**: And do you think you eat five fruits and vegetables per day? **U**: Five no, probably not, but three or four yes I think.
FM+WDH response	So you'd have to eat a lot of fruit to get it.
GT response	That's a good start.

Table 4.7.: Evaluation example in English where neither the GT or FM+WDH system's responses were appropriate, according to the evaluators.

Dialogue history	**U**: I think I'm a pretty healthy eater, so for me it's fine. **S**: Pefect! **U**: But maybe you'll teach me how to eat even better. **S**: I hope so! Do you pay attention to the quantities you take? **U**: Not especially. Well, yeah, no, it depends.
FM+WDH response	You think you're eating right?
GT response	It depends on what you like, you mean?

In general, the obtained results are coherent with our proposal and with the automatic evaluation, especially with the next utterance classification accuracy. While only including one of the proposed embeddings to control the dialogue not always results in a better model according to this evaluation, including both significantly improves the quality of the responses compared to the baseline.

Table 4.8.: Results of the pairwise response quality evaluation combined in the four target languages. Models in bold indicate that they are significantly better than their counterparts ($p<0.05$).

Model A	Model B	Neither A nor B	A is better	B is better	Both A and B
BL	BL+SC	17.50	26.25	30.00	26.25
BL	BL+PH	18.75	27.81	28.38	24.06
BL	**FM**	17.81	24.38	35.00	22.81
FM	**FM+RR**	20.63	18.75	31.56	29.06
FM+RR	**FM+WDH**	17.50	21.88	31.87	28.75
FM+WDH	**GT**	7.50	19.06	50.94	22.50

Table 4.9.: Results of the pairwise response quality evaluation in English.

Model A	Model B	Neither A nor B	A is better	B is better	Both A and B
BL	BL small	12.50	41.25	17.50	28.75
BL	BL+SC	11.25	33.75	23.75	31.25
BL	BL+PH	12.50	27.50	30.00	30.00
BL	FM	11.25	21.25	32.50	35.00
FM	FM+RR	13.75	13.75	38.75	33.75
FM+RR	FM+WDH	13.75	25.00	32.50	28.75
FM+WDH	GT	10.00	30.00	42.50	17.50

The effect of the reranking using just the GPT-2 score is particularly interesting. Even if, in general, it is significantly better than not using it, there are some differences if we compare the results across languages. It improves the quality of the responses in English the most, followed by French and Spanish. In Norwegian slightly worsens the quality of the responses. This could be closely related to the next utterance classification accuracy, which was shown in Table 4.2. In English, the next utterance accuracy is the highest of all languages, and therefore the model selects candidates which are often closer to what a human would select. French and Spanish are next, and so their improvement is not as big as in the English model in this case. Finally, the worst accuracy is ob-

Table 4.10.: Results of the pairwise response quality evaluation in Spanish.

Model A	Model B	Neither A nor B	A is better	B is better	Both A and B
BL	BL no pretrain.	26.25	37.50	18.75	17.50
BL	BL+SC	13.75	20.00	32.50	33.75
BL	BL+PH	13.75	31.25	30.00	25.00
BL	FM	16.25	23.75	33.75	26.25
FM	FM+RR	18.75	20.00	31.25	30.00
FM+RR	FM+WDH	15.00	26.25	33.75	25.00
FM+WDH	GT	5.00	10.00	52.50	32.50

Table 4.11.: Results of the pairwise response quality evaluation in French.

Model A	Model B	Neither A nor B	A is better	B is better	Both A and B
BL	BL+SC	22.50	27.50	25.00	25.00
BL	BL+PH	23.75	23.75	26.25	26.25
BL	FM	22.50	30.00	36.25	11.25
FM	FM+RR	25.00	13.75	36.25	25.00
FM+RR	FM+WDH	15.00	17.50	31.25	36.25
FM+WDH	GT	3.75	22.50	60.00	13.75

Table 4.12.: Results of the pairwise response quality evaluation in Norwegian.

Model A	Model B	Neither A nor B	A is better	B is better	Both A and B
BL	BL+SC	22.50	23.75	38.75	15.00
BL	BL+PH	25.00	28.75	31.25	15.00
BL	FM	21.25	22.50	37.50	18.75
FM	FM+RR	25.00	27.50	20.0	27.50
FM+RR	FM+WDH	26.25	18.75	30.00	25.00
FM+WDH	GT	11.25	13.75	48.75	26.25

tained in Norwegian, which may well indicate that the GPT-2 score by itself is not reliable for successfully selecting good candidates. Moreover, if we now focus on the influence of adding the WDH score instead of using only the GPT-2 score, we can see that it consistently improves the quality of the responses. This definitely makes sense since it already showed an improvement in terms of next utterance accuracy, as shown in Figure 4.5. However, it is important to remark that in this case the reranking is carried out over a set of model-generated candidates, while in the previous study of the next utterance accuracy, the candidates were human responses from the corpus. This indicates that the WDH system is robust no matter if the candidates are sampled from the corpus or generated by the model. Finally, our full proposal (FM+WDH) was compared with the ground truth responses of the corpus. As expected, the ground truth significantly outperforms our model in all languages. Additionally, we believe that in most of these cases where the models produced better responses than the ground truth might be due to the corpus containing a large number of translations. Even if these were automatic and manually corrected, there may still be cases where some translations are not completely accurate or grammatically correct, as previously shown in Table 4.5. In any case, the margin between the ground truth and the generated response is remarkably small in English, which shows that better pretraining is key to developing end-to-end dialogue models.

In this regard, an additional comparison was carried out to measure the effect of not pretraining the baseline model in Spanish (first row in Table 4.10). It underlines the fact that a pretrained LM is essential to enhance the posterior performance of the dialogue model. A similar study was carried out in English.

In this case, we compared the *small* and *medium* pretrained GPT-2 architectures (first row in Table 4.9). The medium architecture showed its superiority, as it has done in many other NLP tasks (Radford et al., 2019). We were not able to pretrain medium models for the other languages due to the lack of computational and corpus resources.

4.7.3 | Human interaction evaluation

Let us introduce the results of the human interaction with the FM+WDH system. The same four evaluators per language that evaluated the responses were the ones interacting with the system. Additionally, some of the non-English evaluators but fluent in English also tested the English system. Thus, the English system was evaluated by 12 experts, and the rest of the systems by 4. Each evaluator carried out two dialogues with the corresponding system, first the introductory dialogue into coaching, and afterwards the first part of a GROW nutrition session. On average, the dialogues were roughly 40 turns long (20 user turns + 20 system turns). This value was controlled via the dialogue phase embeddings. After interacting in the two scenarios, the evaluators filled out the aforementioned CUQ and HFQ questionnaires (Section 4.6.4, Appendix C). Table 4.13 shows the mean and 95% confidence intervals of the score achieved in these two questionnaires for each language.

Table 4.13.: CUQ and HFQ mean scores and 95% confidence intervals (in square brackets), per language.

Language	CUQ score	HFQ score
English	69.1, [61.0, 77.3]	63.1, [51.7, 74.4]
Spanish	62.1, [41.8, 82.4]	62.5, [37.1 ,87.9]
French	68.7, [56.6, 80.9]	61.9, [41.2, 82.5]
Norwegian	39.1, [18.8, 59.4]	43.8, [16.5, 71.0]

The English, Spanish and French models achieved am average score higher than 50 in both tests, which means that on average the evaluators tended to agree on the positive aspects of the system and disagree on the negative ones. On the contrary, this was not the case for the Norwegian system, which shows that there is still a significant performance gap to be closed for systems in languages with very few resources. The English model achieved the best results once again, but interestingly enough, the French and Spanish models were unexpectedly close in terms of HFQ score, and the French one was very close in the CUQ score too. This might be because the pretraining of the GPT-2 models affects mostly in the candidate generation stage, whereas the WDH system is only (except the turn embeddings) learnt on our coaching corpus. The WDH reranking is a key aspect of the whole pipeline, since it is the main responsible for keeping coherence in the dialogue, which greatly influences the user experience. This is even more important when dealing with coaching dialogues where the long-term strategy is so valuable. Thus, we hypothesise that future improvements in this direction would result in more structured, and therefore better-rated, dialogues. In the case of the Norwegian, however, the main issue might be that, overall, the generated candidates lack quality due to the worse pretraining of the GPT-2 generative model. If this were the case, improving the quality of multilingual transformers or better pretraining in low-resource languages would be essential to improve the usability and emotional influence of this kind of models in the future.

Let us now focus on the specific answers to each questionnaire. Figures 4.6 and 4.7 show the average results and their standard deviation for each question in the CUQ and HFQ, respectively. Even if some questions ask about negative aspects of the system, in the figures a standardised score between 0 and 1 is shown, where the higher the value is, the better the performance is too. The values have been computed with the combination of the questionnaires in the four target languages.

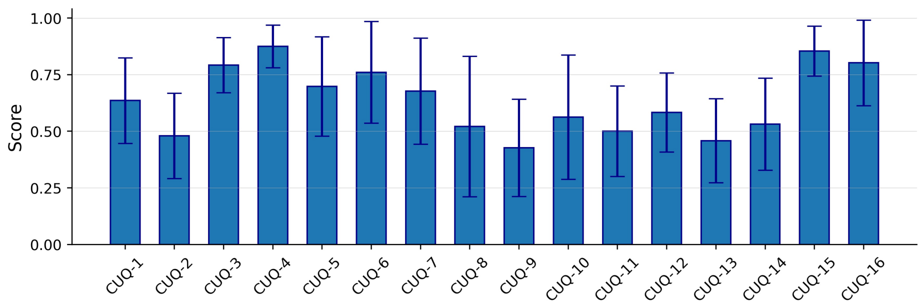


Figure 4.6.: Results of the Chatbot Usability Questionnaire.

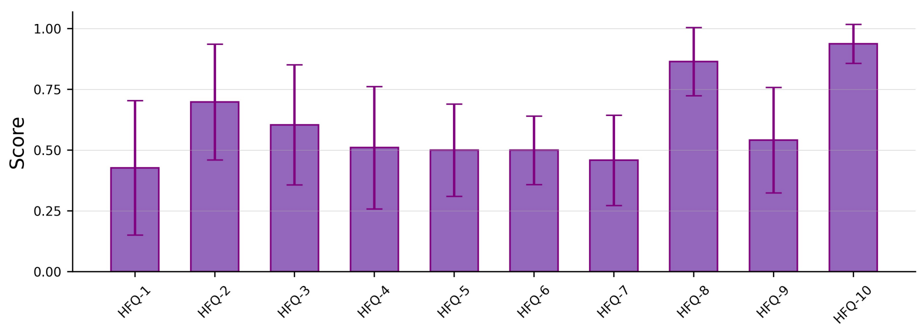


Figure 4.7.: Results of the Hedonic Feelings Questionnaire.

As for the usability, the responses to questions ranging from CUQ-3 to CUQ-6 indicate that the dialogue system presents itself correctly and that indicates well its purpose. More exactly, this means that the user understands that a session of coaching is about to be carried out, and that to this end they first talk about the user's hobbies before getting into the nutrition GROW session. Responses to CUQ-7, CUQ-8, CUQ-15 and CUQ-16 indicate that the interaction with the system is rather simple and easy, which is an important point for future interactions with real users. In general, the performance is not that great in terms of understanding the user and acting accordingly, as reflected in the results of questions from CUQ-9 to CUQ-12. In this regard, it is important to recall that the proposed methodology does not make use of any explicit knowledge representation like entities, ontologies or dialogue acts. It purely learns from the text transcription of dialogues. This makes it possible to develop a dialogue system more easily and affordably, but it also has its limitations. The system is less likely to react to user turns that contain some relevant information than if a Natural Language Understanding module was used, for example. CUQ-13 and CUQ-14 refer to the ability of the system to recover from errors. It seems that the system can recover from errors sometimes, but that other times it fails to do so. This is definitely an interesting and open topic of research, and we plan to use the WDH system to detect dialogue breakdowns, and avoid them if possible. Finally, responses to CUQ-1 and CUQ-2 indicate that the system is engaging to some degree, but that it is also quite robotic.

On the other hand, the HFQ provides useful information to measure the potential impact the system may have on the user, at least short term. Very importantly, experts strongly agree that the interaction with the system is neither depressing nor stressful (HFQ-8 and HFQ-10). This is a good starting point, because at least the system does not seem to give rise to very negative feelings. It does not seem to be boring either (HFQ-2). On the other hand, the HFQ also reveals that there is much progress to be done, since the system could be much more stimulant (HFQ-7). Coaching is about stimulating the user to help them to achieve their own goals. Thus we would really like to improve in this aspect. Nonetheless, experts do not think the communication is not stimulant either, which also means that we are not completely away from our objective. Apart from this, experts feel the system is quite innovative (HFQ-3) and do not agree nor disagree on the fact that the communication with the system is extraordinary (HFQ-1), disappointing (HFQ-4), thrilling (HFQ-5), trivial (HFQ-6) or reassuring (HFQ-9). Being able to produce dialogues even more coherent long-term would likely result in improvements on these aspects.

In summary, these results indicate that, in general, our proposals are heading in the right direction, but also that improvements are probably needed to systematically use our coaching system with end users.

4.8 | The WDH system as a tool to explain the behaviour of the conversational agent

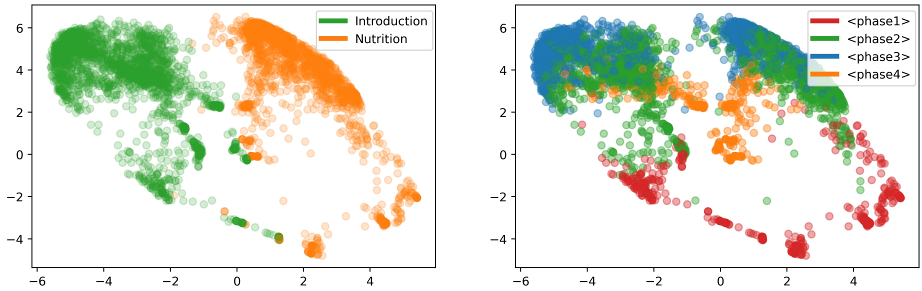
The WDH system has shown to improve the response quality of the system by integrating the whole dialogue history into the decision-making stage. Additionally, it can also be a powerful tool to understand on what basis these decisions have been taken. In this section, we first analyse the distribution of turn embeddings in the low-dimensional space. This can help us understand how the turns are clustered, and intuitively validate those, also by comparing them to dialogue acts. Additionally, we arrange the clusters and dialogue acts into graphs to visualise the paths the system is more likely to take and understand why. Moreover, we believe this kind of analysis could be taken one step beyond, and be used not only to analyse but also to improve the behaviour of the system. We leave this interesting research topic for future work.

4.8.1 | Low dimensional turn embedding space

In all the presented experiments the low-dimensional turn embedding space has been of size 5. Empirically, it has been a good choice to provide interesting results and to make the WDH system work. However, we can also choose to convert the high dimensional turn embeddings into bidimensional, and therefore visualizable, vectors. While this can be done by projecting the five-dimensional vectors into two dimensions with another dimension reduction technique, we have opted to train a second supervised autoencoder. We believe that this way the distribution of the points (system turns) in the bidimensional space should be more similar to the one in the five-dimensional one.

For example, this way we can see the clusters the turns are grouped in. We have shown this distribution back in Section 4.3, in Figure 4.2. There, the number of clusters is 20, lower than the actual 60 used in our experiments, for the sake of clarity. They correspond to the English version of the corpus. In that figure, some grouped turns have been highlighted. This manual inspection already suggests that turns clustered together share semantic information. Additionally, much more patterns can easily be detected. For instance, we can also group the system turns according to the scenario or dialogue phase they belong to, as shown in Figure 4.8.

The groups in this case make a lot of sense. For instance, we can see that the systems turns corresponding to the two scenarios are well separated. Nonetheless, there are two areas where these are much closer. If we check the dialogue



(a) Scenario. (b) Dialogue phase.

Figure 4.8.: Bidimensional turn embeddings grouped by the scenario and dialogue phase they belong to. These experiments were carried out with the English corpus.

phase distribution, we see that these areas correspond to the first and last dialogue phases. This seems very much reasonable, because the greetings and the goodbyes are similar in both scenarios, or at least much less different than the rest of the system turns.

There are also other properties of the corpus that become visible in this space. For example, we can take advantage of our corpus being labeled (Vázquez, 2019), even if we have not used these labels at any stage of the development of the dialogue system. We can check the distribution of the turns according to their labels. This is shown in Figure 4.9.

For sake of simplicity, we are only showing the turns corresponding to a subset of the labels, and some labels have been merged for better visualization (some different types of questions about nutrition are merged into just *Nutrition question*, for example). For example, Figure 4.9 shows that the turns labelled as *Hello* are placed in the same place as the ones corresponding to the first dialogue phase, and the same applies for the *Goodbye* and the last dialogue phase turns. Turns labelled as *System introduction* occupy the same space as the introductory scenario turns of the first and second dialogue phase, suggesting that the system presents itself at the beginning of the first scenario. *Travelling* and *Music/hobbies* are very close in the bidimensional space, roughly in the place of the second and third phase of the introductory scenario. These are actually two of the topics the system usually covers to make the user feel more comfortable.

On the right-hand side of the space, we can find the turns categorised as *Objective* and *Nutrition question*, which correspond to the GROW session about nutrition. They seem to be in a very similar region. This could be due to the dimension reduction being too drastic, and they could perfectly be better separated in a higher dimensional space. Additionally, the *Nutrition question* turns

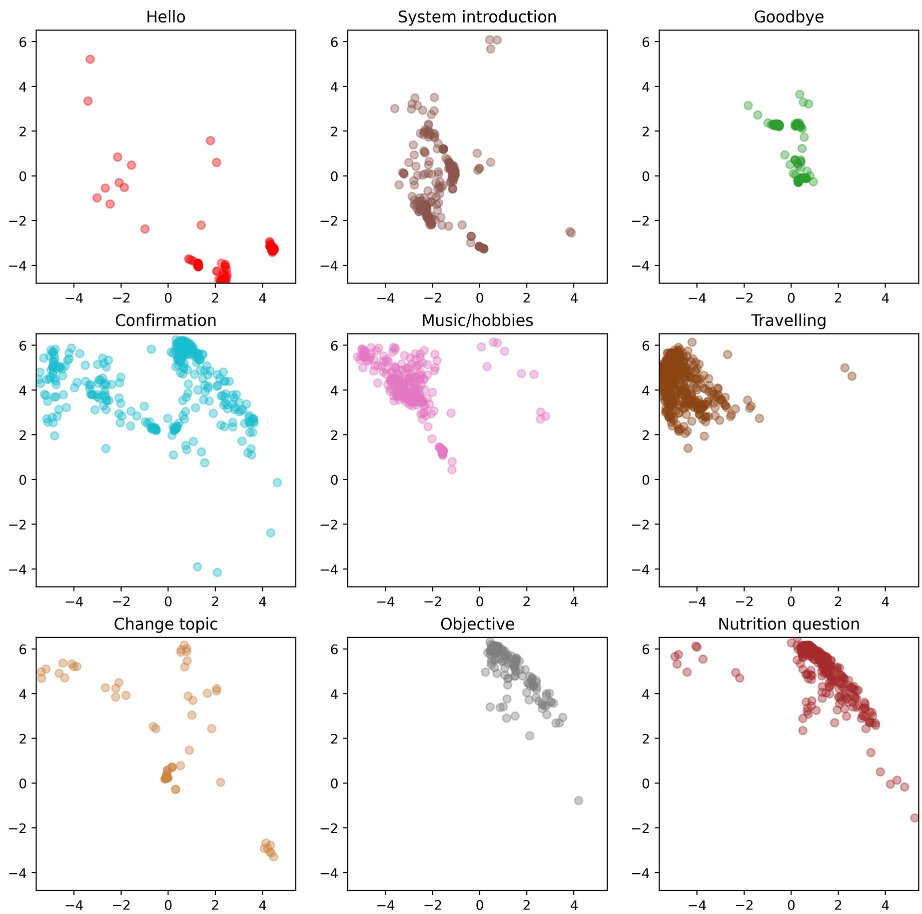


Figure 4.9.: Bidimensional turn embeddings divided according to the dialogue act labels. These experiments were carried out with the English corpus.

seem to be a bit more widespread than the *Objective*. This is coherent with the GROW coaching strategy: the system asks questions about nutrition in many situations, but only focuses on the objective once the user has confirmed that there is something they would like to achieve.

There are also turns labelled as *Confirmation*, which are to be found all over the place in both scenarios. This definitely makes sense; the system may have to confirm whatever the user asks at any point. Finally, we have *Change topic* turns, which are located in both scenarios but only in a few areas. These correspond to utterances where the system and the user finish talking about a given topic, and the system or the user suggests a new one.

4.8.2 | Clustering as an unsupervised way of learning dialogue acts

The fact that this low-dimensional turn embedding space is so structured also validates the proposed methodology from a more intuitive point of view: if the turns that are close in the low-dimensional space share semantic information, can often be labeled with the same dialogue act and are used in similar situations in the dialogue, then the resulting clusters should also represent that information. Therefore, might clustering be considered an unsupervised way of learning dialogue acts to a certain extent? To answer this question, we perform dialogue act classifications from turn embeddings and from cluster indexes. If clusters act as unsupervised dialogue acts, both classifications should produce similar or at least comparable results.

To this end, we employed a bigger set of dialogue acts than the one shown in Figure 4.9. For example, the *Nutrition question* label was subdivided *Motivational question*, *Resources or Obstacles question*, and so on. As a result, a set of 26 dialogue acts were finally used as the classification targets. These are listed and described in Appendix D. We perform three series of experiments. First, we attempt the dialogue act classification task from turn embeddings, via a simple two-layer feed-forward neural network. Second, we do the same, but from the low dimensional embeddings, which were the input to the clustering method in order to avoid the curse of dimensionality issue, as mentioned back in Section 4.5.2. Thus the comparison might be fairer, since the clustering module and the classifier have exactly the same input. Third, we try to predict the dialogue act only from the cluster a system turn has been assigned to. To do so, we learn a mapping from cluster indexes to dialogue acts in the training partition of the corpus, applying a (multi-start) local search heuristic optimisation to maximise the F1 score. Specifically, a first improvement heuristic was employed, and two mappings were considered neighbours if and only if they only differed in one value, i.e. if one and only one cluster was mapped to a different dialogue act. Table 4.14 shows the F1 scores of the three classification methods in the test partition of the corpus in the four target languages.

Table 4.14.: F1 scores of the three classification methods in the test partition of the corpus in the four target languages.

F1 Score at dialogue act classification	English	Spanish	French	Norwegian
From turn embeddings	0.505	0.498	0.483	0.473
From dimensionally reduced turn embeddings	0.328	0.293	0.299	0.292
From cluster index	0.287	0.279	0.285	0.285

In general terms, the F1 scores are reasonable, considering that this challenging task involves a classification between 26 quite imbalanced classes, where the majority class is around 26.4 more frequent than the minority class. Mind that a random classifier obtains an F1 score of around 0.03. The results follow the same pattern across the four languages. The best results are achieved, as expected, with the whole turn embeddings. Then the F1 score drops around two tenths if the lower dimensional embeddings are used. However, interestingly, the difference between the classification from the low dimensional turn embeddings and the cluster indexes is rather marginal. This comparison is very important, since both algorithms take as input the same dimensionally reduced embeddings. Therefore, it seems that clustering is able to extract almost the same information about dialogue acts from those embeddings as a classifier trained specifically to do so. Thus, it seems reasonable to say that, indeed, clustering works as an unsupervised way of learning dialogue acts. At least, there seems to be a strong correlation between the learnt clusters and the dialogue acts.

To gain a deeper insight into this correlation, it is especially interesting to analyze how the F1 score of the cluster to dialogue mapping changes depending on the number of clusters. This is shown in Figure 4.10, it is the purple line in the plot. The F1 score grows a lot from 10 clusters to 50, but from 60 clusters on it stabilises. Thus, the optimal number of clusters (around 60), is quite higher than the number of dialogue acts, the double approximately. This indicates that if the number of clusters is too low, some of the clusters contain turns with many different dialogue acts. After splitting them, when the optimal number of clusters is reached, there are multiple clusters mapped to the same dialogue act. The turns within these clusters probably differ in the context they are used: since the low-dimensional turn embeddings are learnt in a way that they contain information about the scenario and the dialogue phase, the clustering might make some distinctions where the dialogue acts do not. For example, if we consider the system turns "*I understand that you have a healthy eating routine*" and "*I understand, you really love travelling*", it may perfectly happen that they are assigned to different clusters, one that contains mainly similar sentences about nutrition and the other one into a cluster more related to travelling or to the introductory scenario. However, regarding the dialogue act, both would be labelled as *I understand.* Last, if we increase the number of clusters even more, the F1 score does not notably change anymore. This is probably due to some clusters being divided, but then being mapped to the same dialogue act. Thus, the classification results are very similar.

Figure 4.10 also shows the relation between the selected number of clusters and the coefficient of variation of the number of turns per cluster, the WDH system's accuracy at the next utterance classification task. The coefficient of variation of the number of turns per cluster (the ratio of the standard deviation to the mean) indicates how balanced the number of turns per cluster is; a lower coefficient of variation implies that the clusters are more balanced, i.e., that they

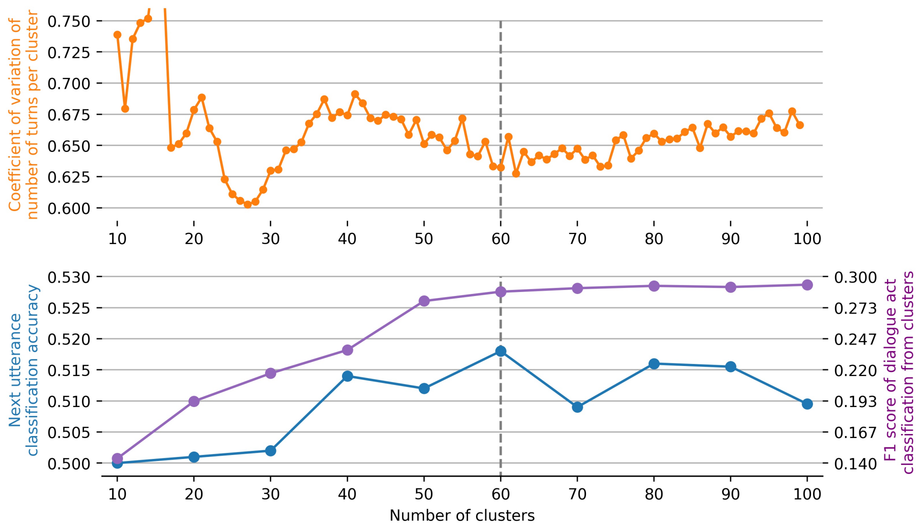


Figure 4.10.: Different metrics in terms of the selected number of clusters. On top, the coefficient of variation of the number of turns per cluster. At the bottom, the next utterance classification accuracy in blue (the scale is on the left), and the F1 score of dialogue act classification from clusters in purple (the scale is on the right).

include a more similar number of turns each; whereas larger values indicate that some clusters are very populated while others contain very few turns inside. Lower values are therefore preferred, since they should allow better modelling of the cluster flow, as this data set would be more balanced. In our case, there is a first and very notorious local minimum at around 25-30 clusters. Then the values go up at 40 clusters and they are reduced again, even though slightly at around 60-70 clusters. We decided to select 60 instead of 25 or 30 due to the behaviour of the other metrics.

Last, the next utterance classification accuracy is the noisiest metric. Anyway, it is worse with fewer clusters (10-30), and then it improves after 40, with the maximum at 60. Thus we believe that the choice of 60 clusters represents a good balance between all these three metrics.

4.8.3 | Cluster and dialogue act dynamics

This relation between dialogue acts and clusters is also visible if we analyse the dialogue flow. This can be done by arranging the clusters or dialogue acts into a graph that shows the number of transitions between each of them. We show such graphs for clusters in Figures 4.11 and 4.13, and for dialogue acts in Figures 4.12 and 4.14. These have been built with the English version of the

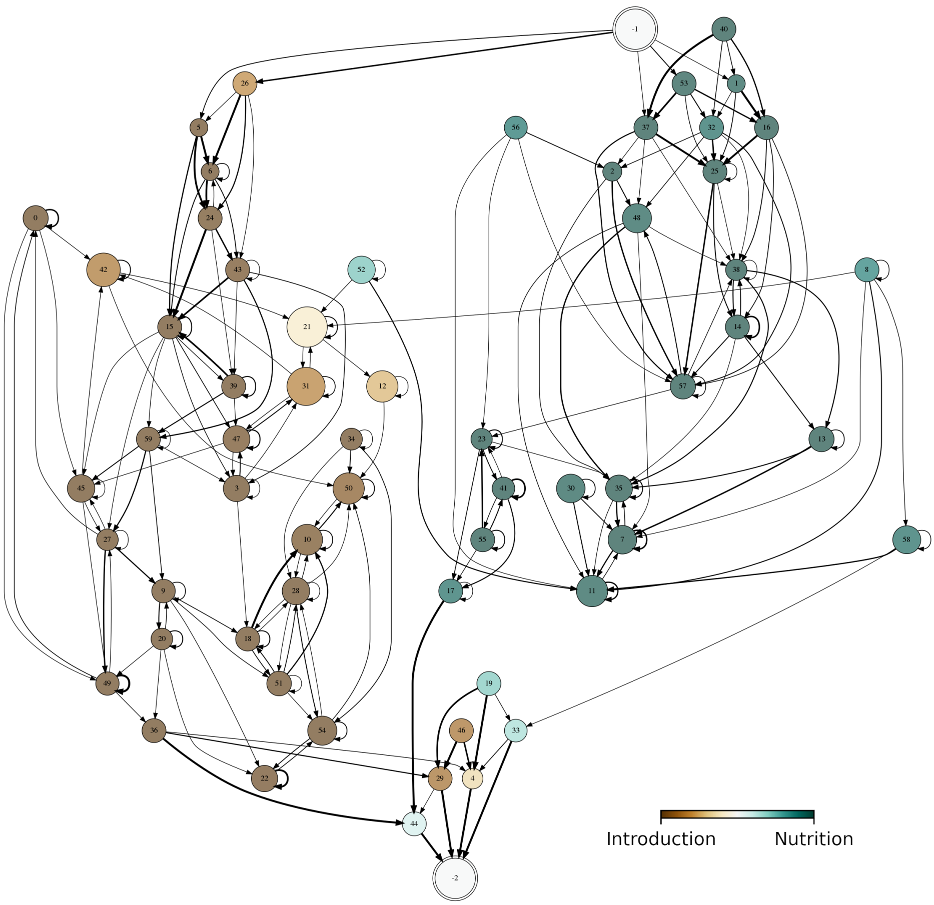


Figure 4.11.: A graph where nodes represent clusters, and their colours the scenario of the system turns they gather. These experiments were carried out with the English corpus.

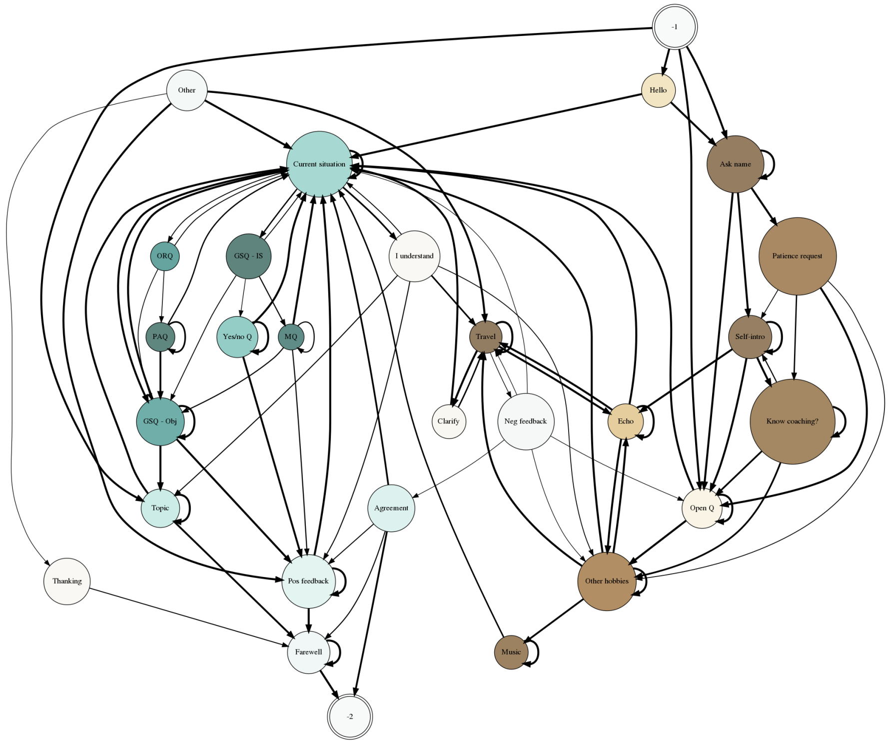


Figure 4.12.: A graph of system dialogue acts, coloured according to the scenario they were used in. These experiments were carried out with the English corpus.

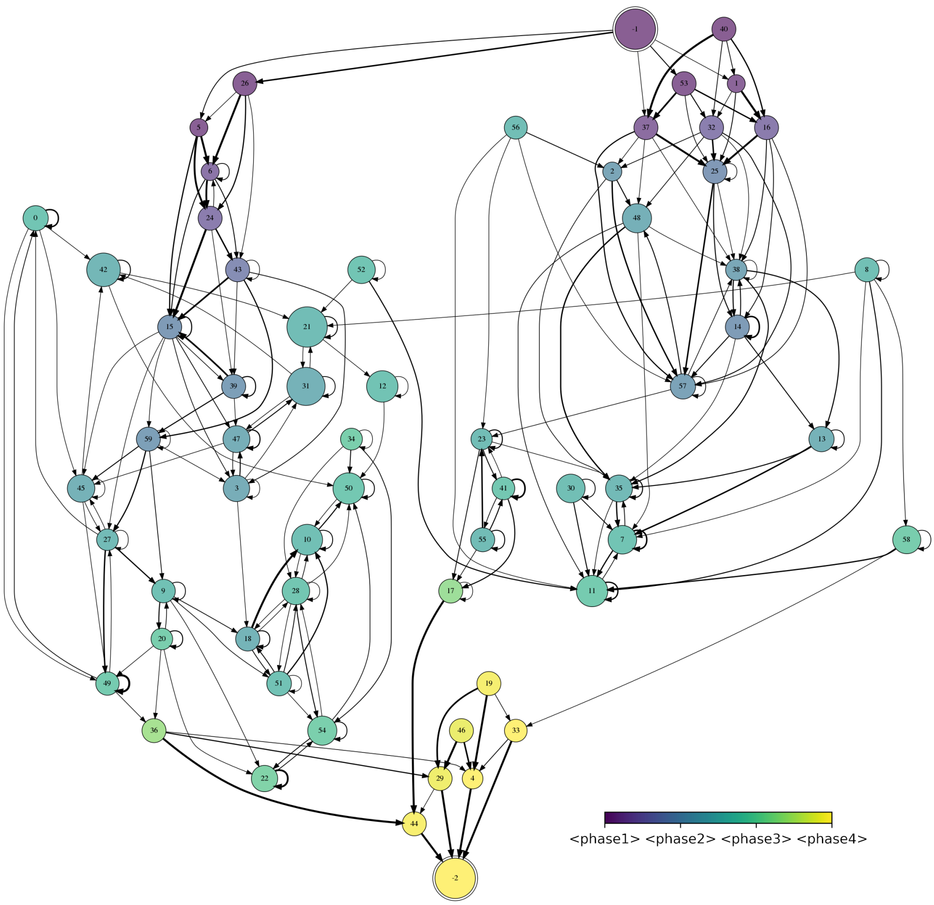


Figure 4.13.: A graph where nodes represent clusters, and their colours the dialogue phase of the system turns they gather. These experiments were carried out with the English corpus.

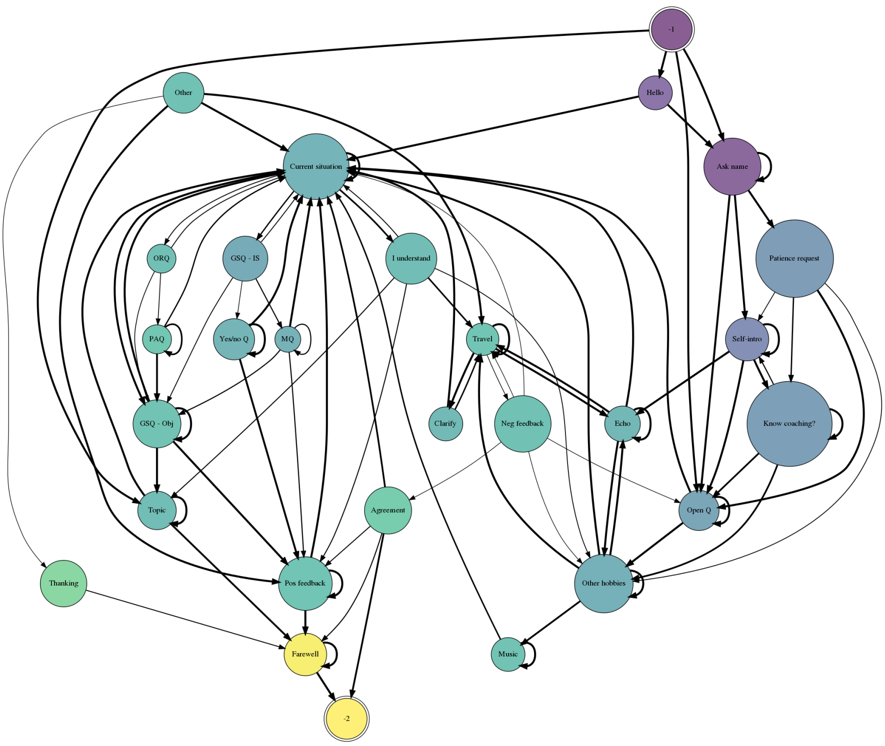


Figure 4.14.: A graph of system dialogue acts, coloured according to the dialogue phase they were used in. These experiments were carried out with the English corpus.

corpus. In all the diagrams, node -1 is the source, i.e. the nodes that come after it represents the cluster/act of the first system utterance in a dialogue. On the other side, node -2 is a sink; it represents the end of a dialogue. To keep the graphs as informative as possible, we skip some minor transitions: we do not show edges that correspond to less than five percent of the total transitions from one cluster/act to another. This is the reason why some nodes have no edges in their direction in the figures.

In Figure 4.11, the clusters have been coloured according to the scenario of the turns they gather. The brown nodes refer to clusters that mainly contain turns used in the introductory scenario. Alternatively, the greener ones correspond to clusters related to the GROW session about nutrition. The same colour scheme has been applied in Figure 4.12, but for dialogue acts. If we focus on Figure 4.11, it is interesting that, while most of the clusters are one-sided, there are a few that share introductory and nutrition turns. These often include generic turns like confirmations or backchannels. In general, we can see that the graph can be split into two major regions: the browner one that corresponds to introductory dialogues, and a greener one that unravels the structure of the coaching sessions about nutrition. The two regions merge almost exclusively at the end of the dialogues, when the system bids farewell to the user. However, the dialogue act graph in Figure 4.12 is not so split. Even though many dialogue acts clearly correspond to one scenario, many others are coloured in white, such as, *Thanking*, *I understand*, *Neg. feedback* or *Clarify*. As aforementioned, the clusters corresponding to these acts have probably been broken into several different clusters. Another dialogue act that is probably divided into many clusters is *Current situation*, which is a very central node in the graph, meaning that it is used in different contexts. This makes sense, since it is necessary to analyse the user's current situation to establish a goal and carry out the coaching session accordingly.

On the other hand, the graphs shown in Figures 4.13 and 4.14, where the nodes are coloured in terms of the dialogue phase, show similar patterns. For example, in this case we can deduce that the *Topic* label has also been divided into multiple clusters. On the one hand, it is coloured in blue/green which means that it contains many turns used when the dialogue is quite advanced; but there is also an arrow from -1 to it, denoting that there are many dialogues that start with that dialogue act. This makes sense, because the *Topic* dialogue act groups utterances that open, close or choose a new topic. This distinction has probably been learnt in the clustering, but is not shown in the dialogue act graph.

Last, as expected, the flow of the clusters is best modeled with GRUs as opposed to N-gram models. Table 4.15 shows the accuracy and top-N accuracy (with N=3) on the test set for each language.

Table 4.15.: Accuracy and top-N accuracy (with N=3) obtained by the cluster sequence modelling models across the four languages on the test set.

	English		Spanish		French		Norwegian	
	Acc.	Top-N acc.	Acc.	Top-N acc.	Acc.	Top-N acc.	Acc.	Top-N acc.
GRU	0.350	0.581	0.346	0.567	0.327	0.592	0.356	0.616
2-gram	0.243	0.479	0.247	0.475	0.248	0.535	0.231	0.522
3-gram	0.183	0.352	0.188	0.376	0.177	0.413	0.173	0.396
4-gram	0.147	0.304	0.155	0.299	0.151	0.349	0.146	0.323

4.9 | Conclusions

Let us summarise the most notable findings of our research and their implications for developing intelligent conversational agents.

Bridging the gap between state-of-the-art Artificial Intelligence techniques and current coaching models. If we compare the dialogue technologies used in coaching agents found in the literature and the market and the ones employed in the most novel and prominent chatbots, there is a big disparity. This is valid for most of the health-care-related conversational systems too. In a nutshell, professional dialogue strategies in health-care-related conversational agents are often implemented, at least partially, via hand-crafted policies. On the other hand, state-of-the-art dialogue models are fully data-driven, and thus do not require carefully designed policies, these are learnt from the data. We have shown that it is possible to adapt and modify these novel technologies to develop complex coaching conversations. This provides major benefits. First, it simplifies the whole design process. Second, the resulting dialogue models might potentially perform better and for more domains than rule-based models, which can only be programmed for a limited amount of situations. Nonetheless, there are still limitations to this attractive approach. Its main drawback is a consequence of the models being fully data-driven. It may happen that, for instance, the system makes an error at some point in the dialogue, as a result of a non-completely successful training; or there might also be some inconsistencies with the name entities, because the system is not able to automatically coherently keep track of them. While those errors could be solved in a rule-based system easily, they have no direct solution in a fully data-driven model. Our proposals help alleviate this issue by enhancing coherent responses, but they do not ensure errors will not happen. Other research ideas, such as neural entity linking (Chi et al., 2021), could be used to further improve in this area.

These methodologies are language agnostic. Classical modular dialogue systems require the development of some very language-dependent modules, such as the NLU or NLG modules, which often multiply the effort needed to develop the conversational system in an additional language. In contrast, our models in English, Spanish, French and Norwegian require just the same engineering effort, which is encouraging. The only limitation is the pretraining step of the generative model. In this regard, we have shown that pretraining the GPT-2 model on languages like French or Spanish with open domain corpora leads to only slightly worse results than the ones obtained with the official pretrained English model. However, the experts evaluated the Norwegian system as much poorer, due to fewer data being available for the pretraining. In any case, with more and more research targeting non-English languages, we believe that the difference in performance of conversational agents and other NLP models in English and other languages should attenuate soon (Jiang et al., 2022), and that therefore language-agnostic approaches like ours might gain popularity and perform even better.

Improved response generation by conditioning the generative network using scenario and dialogue phase embeddings. Our first methodological contributions are the scenario and dialogue phase embeddings, which have led to better response generation, as shown in Table 4.8. Learning this kind of embeddings is very simple and very flexible too. As aforementioned, scenario embeddings could be used in multi-task or multi-domain environments, which have gained a lot of interest from the dialogue community (Eric et al., 2019; Rastogi et al., 2020). Apart from enabling the use of a single system for all the domains, the learnt embeddings could also provide information about each task and serve as a tool for comparing them. We have not carried out such an analysis here because there are only two domains in our corpus, but it would be interesting to research in this topic in the future. As for the dialogue phase embeddings, for the moment we have predefined when a dialogue phase starts and when it ends, based on a manual inspection of our data. However, we believe that this approach could be further enhanced, probably with mechanisms that learn the beginning and the end of a phase in an unsupervised way. We think that the WDH system could be useful to this end.

Improved long-term coherence via the WDH system and unsupervised dialogue act learning. The proposed WDH system has shown great potential. It has improved the performance of our baseline models in automatic and human evaluation in the four target languages, showing that dialogue models actually require long-term context information to keep more coherent conversations. Therefore, this approach could also improve the performance of many dialogue models, especially in tasks where dialogues are longer and more structured. Tasks requiring to process shorter dialogue histories (like the one presented next in Chapter 5) would not benefit much of the WDH system, especially if the whole dialogue history can be used as input to the generative model.

Additionally, we have analysed the clustering process inside this long-term context system, and a strong correlation with dialogue acts has been found. More precisely, as the experiments carried out in Section 4.8.2 indicate, the clusters the system turns have been grouped in share to a certain extent the dialogue act they were assigned in a manual labelling. In other words, clustering system turns and then mapping the corresponding cluster into a dialogue act is almost as effective as directly applying supervised learning from the low-dimensional turn embeddings, and not exceedingly worse than classifying the whole turn embedding. Thus, we hypothesise that building a similar system that uses dialogue acts would not outperform our proposal by a big margin. This is important, since many conversational agents rely on dialogue act representations, which involve costly and time-consuming annotations. We hope that our efforts to find alternatives will trigger other researchers' interest in alternative (and potentially unsupervised) turn representations, which could simplify the process of building and designing conversational systems.

Chapter 5

Speech-aware spoken dialogue management

5.1 | Introduction

Recent advances in self-supervised speech representation learning have opened the door to new ways of including acoustic information in AI systems. Motivated by the success of similar approaches in NLP, these speech representations are learnt by transformer-based neural networks using unlabeled data only. They have demonstrated to be really powerful. State-of-the-art results (or close to that) can be obtained relatively easily in a variety of audio-related tasks using them, even if small domain specific data is available (Baevski et al., 2020; Pepino et al., 2021; Seo et al., 2021).

Nonetheless, the application of such audio embeddings in SDSs remains yet largely unexplored. SDSs are inherently devoted to process the users' audio signal and to provide the most convenient response given the dialogue context. But due to the difficulties of working directly with audio signals, these are often mapped into words using an ASR, and then NLP techniques are applied to understand the user and act accordingly. This approach is very dependent on the ASR providing a correct transcription, which might not be the case in noisy environments, or if the user is non-native or has an uncommon accent (Litman et al., 2018). Moreover, this approach ignores important information in the users' speech, such as their emotional mood, prosody, or the noise level of the environment, which could be key to carry out a better dialogue strategy. This argument is supported by previous studies with young adults that have compared video chat, audio chat and text-based chat, where the latter has shown lower levels of bonding than the other forms of interaction (Sherman et al., 2013).

In this chapter, we study how audio embeddings can be used to include this kind of information in dialogue policies, and yield better dialogues policies. We propose a transformer-based DM capable of processing both the text dialogue history and the audio signal of the last user's turn. We compare it against a version of itself that does not explicitly process audio, in a variety of conditions and with different learning algorithms. We also compare three of the latest audio embedding models (Wav2Vec2 (Baevski et al., 2020), HuBERT (Hsu et al., 2021) and UniSpeech-SAT (Chen et al., 2021)) and two different methodologies to extract the speech representations from the user turns. Automatic metrics, human evaluation and manual inspection in the DSTC2 dataset are in favour of our hypothesis: audio embeddings help to learn better dialogue policies.

We analyse the effects of adding audio embeddings to dialogue policies combined with text representations obtained from two ASRs' output (of different qualities) and manual transcription. Regarding training algorithms, we experiment with SL and two policy gradient-based RL algorithms. Consequently, we identify under which conditions audio embeddings help to learn better dialogue policies: they help the most with noisy ASRs, especially when the policies are learnt via Supervised Learning. Whilst speech representations allow a better user understanding in many occasions (e.g. identify what kind of information is being requested), they are also able to indicate the system that a turn has been noisy and that the ASR transcription might not be very reliable in some cases. We have also found that the improvements are higher when learning policies via SL as opposed to RL, because RL policies adapt better to the uncertainty in the ASR output.

Additionally, and in order to carry out this experimentation, we extend the conventional pipeline for dialogue simulations. The simulations are needed to train RL policies and also to evaluate the performance of the policies in terms of task completion metrics. Such pipelines include a User Model (UM) module that simulates the behaviour of the users, by outputting dialogue acts or generating text. In those cases, the effect of the dialogue being spoken is usually simulated by introducing artificial errors to the UM output. However, for our research the audio signal corresponding to the user's turn needs to be fed into the DM. Thus, we propose an additional contribution: a novel User Audio Sampler. This module is capable of sampling audio turns that correspond to the output of the UM from the corpus. Sampling probabilities are adjusted depending on the last user dialogue act, the turn number in the dialogue, and the number of repetitions of the corresponding dialogue act.

The rest of the chapter describes related works in Section 5.2, our proposed approach for audio-based policy learning in Section 5.3, the User Audio Sampler in 5.4, our experimental framework (corpus, simulation pipeline, evaluation metrics and learning algorithms) in Section 5.5, experimental results and analysis (automatic evaluation, audio embedding comparison, human evaluation and a manual inspection of the models) in Section 5.6, and Section 5.7 presents our conclusions. A preliminary version of the research described in this chapter has been presented in a conference paper at ASRU 2021 (López Zorrilla et al., 2021b), and the complete version has been be published in the TASLP journal (López Zorrilla et al., 2022).

5.2 | Related Work

The inconveniences caused by relying only on ASR outputs to make decisions have been previously treated in different ways in SDSs and related areas. Some classical approaches to deal with this problem have focused on extracting as much information as possible from the ASR at hand. For example, a conventional methodology to build more robust SDSs consists of processing the top N hypotheses of the ASR rather than just the main output (He and Young, 2003). Some other alternatives make decisions based on ASR word confidence scores (Hakkani-Tur and Riccardi, 2003) or word confusion networks (Hakkani-Tür et al., 2006; Justo et al., 2011), which were proposed around two decades ago and are still in use nowadays (Swarup et al., 2019) in SDSs and in Spoken Language Understanding (Weng et al., 2020; Ganesan et al., 2021). In the same vein and though hard to scale up, POMDP-based dialogue managers (Williams and Young, 2007) were developed to cope with the uncertainties related to SDSs, including ASR outputs. Other efforts to include information present in the users audio signals but absent in the ASR transcription can be found in the area of emotion aware dialogue systems (Pittermann and Pittermann, 2006; Olaso et al., 2021). This kind of systems often include a module devoted to emotion recognition from audio, whose output is employed by the dialogue manager in the decision making step. However, none of the methods presented in these works explicitly process speech representations and make decisions based on them.

Closer to our work, we can find the research area of end-to-end spoken language understanding, where an audio is mapped into semantic labels directly. The encoder-decoder approach was the first way to tackle this problem (Haghani et al., 2018). Lately Wav2Vec2, one of the audio embedding networks that we use in this study, has also been used to this end (Seo et al., 2021), showing the potential of this transformer network. But these studies focus on classifying audio signals, not on making decisions based on them, which is a notable difference when comparing them to our investigation.

The number of previous works describing dialogue systems that process the users' audio directly (without an ASR) is rather scarce. Nguyen et al. (2018) and Le et al. (2019) present sequence-to-sequence models that process audio features in the context of audio visual scene-aware dialogue (AlAmri et al., 2019, 2018), where the system has to answer a number of questions related to an audio visual scene. However, the audio to be analysed is not the users' audio, but the scenes' one. Closer to our approach is Shi and Yu (2018), who explore the inclusion of user sentiments in end-to-end dialogue systems. They train a dialogue policy that takes some audio features as input via SL. They found however that using the output of an external sentiment classifier worked better than the raw features. They also fine-tuned their DM using RL, but without including audio features.

The work presented in Young et al. (2020) is probably the closest to ours. They investigate the inclusion of users' audio in an LSTM-based encoder-decoder network for response generation in open-domain dialogue using SL only, not RL. To this end, they first train word-level audio embeddings in a response selection task and then concatenate those to traditional word embeddings to form the input to the network. In contrast to their work, our approach is simpler in terms of implementation, we use novel audio embedding networks which require no further pretraining, and we apply it to task-oriented dialogue data. In addition, the audio representations used in Young et al. (2020) were trained without taking into account the word order in the turns, which could miss valuable audio information such as prosody.

In this chapter we also show that our dialogue policies can be easily fine-tuned using both SL and RL. Even though (deep) RL has established methodologies to train different types of dialogue systems (Casanueva et al., 2018; Cuayáhuitl et al., 2017; Takanobu et al., 2019; Williams and Zweig, 2016), the user simulators employed in previous works only generate either words or dialogue acts—not audio. In contrast, we present a novel User Audio Sampler capable of providing audio-based dialogue turns. We refer the readers to (Latif et al., 2021) for a more-in-depth analysis of RL in spoken dialogue systems.

5.3 | Audio-Aware Dialogue Management

To measure the impact of including speech representations in DMs, we compare policies that make decisions based on text-based dialogue history only against policies that process the history in the exact same way but also include audio embeddings to represent the last user's turn audio signal. We use a simple but contemporary architecture for our dialogue managers, see Figure 5.1. First, fixed-length representations from the dialogue history and the last user's turn audio are obtained with different architectures of transformer networks. Then, a linear predictor is used to compute the unnormalised probability distribution of the system's next dialogue act. Dialogue acts (as meanings of utterances) are used as output because they facilitate the integration of a user model and the policy optimisation with RL.

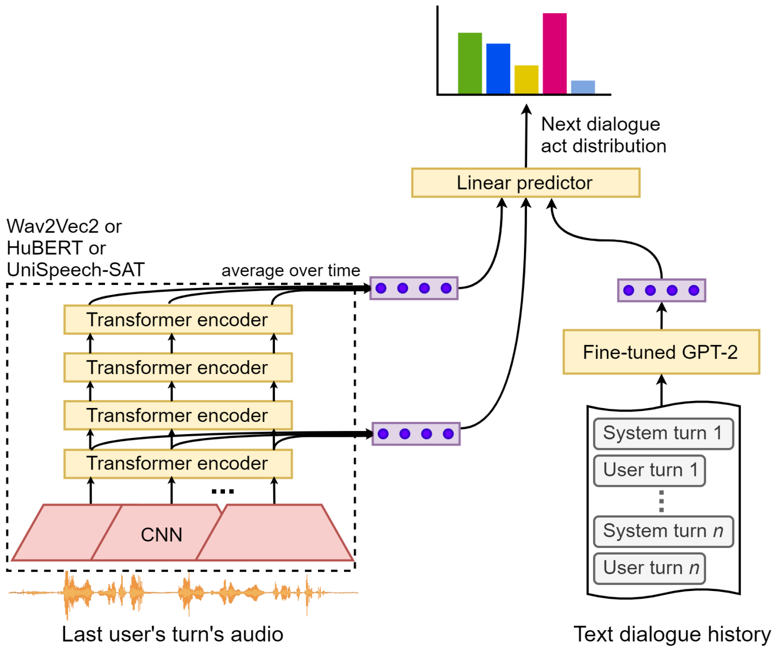


Figure 5.1.: Proposed dialogue manager architecture using audio-textual features.

Text dialogue history. A pretrained GPT-2 transformer network (Radford et al., 2019) is used to process the text dialogue history and is fine-tuned during the training process. This approach has shown great success in both open domain (Roller et al., 2020) and goal oriented (Ham et al., 2020) dialogue management. Each turn in the dialogue history is represented as raw text, i.e. no dialogue acts or named entities are used as input to the policies—to keep our approach as simple as possible. The sequence of turns are processed with a pretrained BPE tokeniser before being fed to the GPT-2 network. We employ a strategy similar to Wolf et al. (2019) to build the input to the GPT-2 network; three sequences of embeddings are added before being fed to the transformer, as represented in the example of Figure 5.2. First, the sequence of token embeddings is generated by concatenating the text of the turns in the dialogue history and processing it with a pretrained BPE tokeniser (first row in Figure 5.2). Dialogue turns are separated with special tokens (<sys> or <user>) that indicate when system or user turns start. The second input sequence is made of segment/speaker embeddings, and is devoted to underline whose turn is (second row in Figure 5.2). The aforementioned <sys> and <user> tokens are used to this end. Last, the position embeddings provide the notion of order, as in most transformer networks (Radford et al., 2019) (third row in Figure 5.2). A <DA_pred> token is appended to the token and segment embeddings to indicate that the input sequences are complete and the dialogue act prediction should be made. In the example, the <API_call> and <no_DB_result> are used to log database searches in the dialogue history, as explained later in Section 5.5.2.

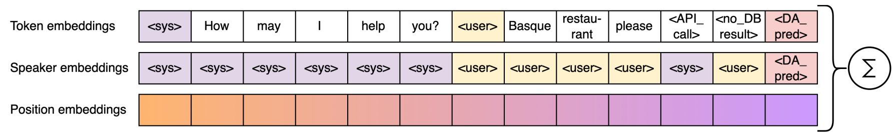


Figure 5.2.: Example set of inputs as part of the 'text dialogue history' in Figure 5.1 showing how it is represented in the proposed dialogue manager architecture.

Last user's turn audio. We combine text with speech-based representations of the last user's dialogue turn. We compare three audio embedding models trained with self-supervised learning: Wav2Vec2 (W2V2 in short) (Baevski et al., 2020), HuBERT (Hsu et al., 2021) or UniSpeech-SAT (also referred to as 'UniS.') (Chen et al., 2021). Even though each model is trained in a particular manner and has unique features, they all share a similar neural network architecture: a Convolutional Neural Net to digest the raw audio signal, and a multi-layer transformer on top of it to produce representations at different levels of abstraction, depending on the layer.

We keep the audio embedding models frozen during training, as recent studies (Yang et al., 2021) have shown that great success can be achieved in a number of tasks via linear predictions from the audio embeddings only, without any need of fine-tuning. The three audio embedding models employ a 12-layer transformer with a hidden size of 768. Thus, they output 768×12 values per time frame. They output 50 sets of vectors per second, and so the total is too high to directly perform predictions from them. In order to reduce the size of the representations to enhance our training procedure, we average the output of each layer in the time dimension, as suggested by Yang et al. (2021) and Chen et al. (2021). We further reduce the dimensionality of the speech representations by selecting the output of a subset of layers. We do not just use the last layer because its representations might well not be the best (Yang et al., 2021), depending on the task. In Section 5.6.2, we study which are the best layers for each model in our case.

Furthermore, we also explored the option of fine-tuning the audio transformers instead of keeping them frozen, in Section 5.6.2. However, we obtained poorer results, and therefore the experiments presented in this chapter are carried out without fine-tuning the audio embedding models.

5.4 | User Audio Sampler

User simulations often output a dialogue act corresponding to the next user turn, which is then converted to text via a user NLG. This approach, however, is not appropriate in the proposed framework because of the requirement of an audio signal corresponding to that text.

In order to optimise the chance of finding an audio corresponding to the output of the UM, we do not use any NLG. Instead, we directly search for turns labeled with the same dialogue act and associated slots in the corpus of real dialogues. Multiple user turns are found in most cases, unless the UM generates dialogue act-slot combinations not appearing in the corpus. From the set of candidates, any turn should already be valid and its audio and transcription (if needed) could be provided to the next module in the dialogue system. However, we consider a couple of factors that make some candidates potentially more suitable than others.

First, we take into account the turn number of the current simulated dialogue compared to the turn number of a given candidate. The justification is as follows. Assume that a dialogue is taking too long and the user starts to feel tired of the interaction. The user may speak in a different way than in the first few turns (when the user first met the system). We assume that selecting a turn that occurred in a similar situation (turn number-wise) of the dialogue to the one the simulation is in should lead to more realistic simulated dialogues, and with potentially more relevant audio information.

The second factor is the number of repetitions of the dialogue act output by the UM in the dialogue, and its reasoning is the following. Assuming that a given dialogue act/slot combination has been used more than once in a dialogue, it is probably due to the system not understanding it correctly and requesting the same information again. When such a situation happens in a dialogue, users tend to get upset on the one hand, and to speak louder and slower on the other hand. This information should also be reflected in the audio signal and could be exploited to improve dialogue policies.

Thus, the sampling probability of the candidate user turns is computed as follows. First, we compute a score for each candidate in terms of the aforementioned two criteria according to:

$$s_i = \frac{1}{|t_i - t_d| + \epsilon} + \frac{1}{|r_i - r_d| + \epsilon} \,,$$

where s_i is the score obtained by the i-th candidate, t_i is the turn number of the candidate in the original dialogue in the corpus, t_d is the turn number of the simulated dialogue, r_i is the number of repetitions of the dialogue act/slot combination in the original dialogue until the appearance of the candidate, r_d is the number of repetitions of the dialogue act/slot combination in the simulated dialogue so far, and ϵ a small constant to prevent divisions by zero.

The scores are then converted into probabilities by dividing them by the sum of all the scores.

5.5 | EXPERIMENTAL FRAMEWORK

5.5.1 | CORPUS

Recent dialogue corpora released in the last few years (e.g. MultiWOZ (Budzianowski et al., 2018), STAR (Mosig et al., 2020) or SGD (Rastogi et al., 2020)) have focused on text based dialogue modelling and do not include audio. Some of the largest spoken dialogue corpora are the DSTC 1, 2 and 3 datasets (Williams et al., 2013; Henderson et al., 2014a,b). Among these, the DSTC2 dataset (Henderson et al., 2014a) is by far the most used corpus for research in spoken dialogue technology and therefore we use this corpus in this work, which also allows us to more easily develop the whole dialogue pipeline due to the publicly available modules for this task.

DSTC2 contains 3235 human-machine dialogues in the domain of restaurant search, acquired with three different dialogue systems. The corpus makes use of 8 slot types: *area*, *food type*, *restaurant name*, *price range*, *address*, *phone*, *postcode* and *signature*. All the slots are requestable, which means that users can ask for information about any of those. For example, they may ask about the phone number of a given restaurant. In contrast, only the first 4 slot types are informable, i.e. they can be used to constraint in the restaurant search. This means that users can look for restaurants by area, food type or price range, but not by postcode for example. The corpus is split into three partitions: *train*, *dev* and *test*, which contain 1612, 506 and 1117 dialogues respectively. We merge the original *dev* and *test* partitions to build our testing data. In that way the training and test partitions have the same amount of data. This is important because the module in charge of sampling user audios—the User Audio Sampler presented in Section 5.4—is sensitive to the number of available audio turns to sample from. This is done to prevent biased user behaviour.

5.5.2 | Dialogue pipeline for simulations

We use dialogue simulations to evaluate the performance of our dialogue policies and to train them using RL. The simulation pipeline is illustrated in Figure 5.3. The remaining of this subsection describes its components.

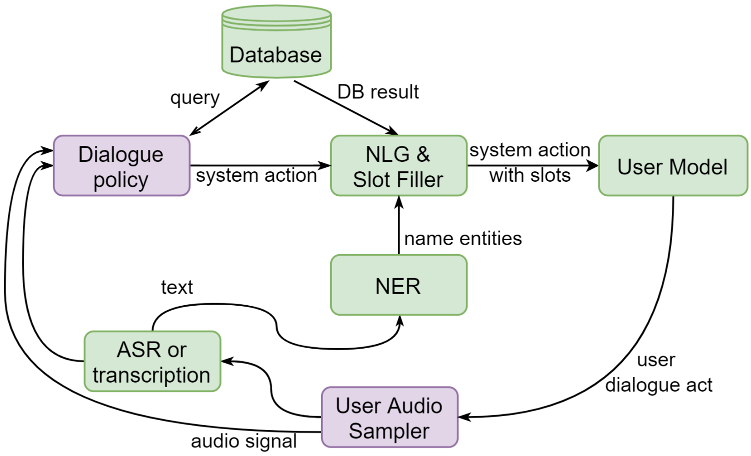


Figure 5.3.: Complete simulation pipeline. Our contributions are related to the modules highlighted in purple.

Dialogue Manager Details. Our dialogue polices use the publicly available *small* pretrained GPT-2 architecture and the so-called *Base* one for the Wav2Vec2, HuBERT and UniSpeecch-SAT networks. Our dialogue states take into account a dialogue history truncated to the last 9 turns to avoid excessive GPU memory consumption. Our dialogue actions use composite dialogue acts to support multiple dialogue acts in a single dialogue turn (e.g. `confirm | area + request | food`), similarly to the procedure in DeepPavlov DSTC2 (Burtsev et al., 2018). The reward function and learning algorithms used for selecting the best dialogue act in each state are described in Section 5.5.4.

Database. Although no database was released as part of DSTC2, database calls can be inferred from the data to form a large enough dataset to perform dialogue simulations with it. Our policies are thus able to make database queries. In order to log this activity in the text dialogue history, every time a database query is made, the token <API_call> is added to the dialogue history. If the query is successful, the token <DB_result> is added. The token <no_DB_result> is concatenated otherwise. When multiple restaurants are retrieved, only one is selected (randomly). Thus, all information that the system may provide in subsequent dialogue turns would correspond to that result. If new user constraints are detected and the dialogue manager makes a new successful API call, the information retrieved from that point onwards would correspond to the latest search result.

Named Entity Recogniser. Since our dialogue policies output dialogue acts containing one or a few slots, we use a Named Entity Recogniser (NER) to extract named entities from user turns to fill the slots of the dialogue acts. Our Name Entity Recogniser (NER) component, based on fuzzy matching, is a slightly improved version of DeepPavlov's NER for this task.

Slot Filler and NLG. We use a rule-based slot filler to select the slot values associated to a dialogue act. As a dialogue progresses, we keep track of the recognised named entities by the NER module and the output of database searches. Depending on the dialogue act, we fill the slots with the last values produced by the NER or database modules. Our NLG module produces text corresponding to the system turns given a pair of dialogue act and selected slots using predefined templates. Since the user model works at the dialogue act level, the generated text is only used to fill the dialogue history. The slot filler is also in charge of selecting the search criteria for the database searches, based on the last entities recognised from the user. For instance, in the previously shown example of Figure 5.2, the only recognised entity would be *Basque*. Therefore, the only condition in the consequent database search would be that the food type is Basque, and there would not be any constraint regarding the area or the price range.

User Model (UM). Our UM is based on Attributed Probabilistic Finite State Bi-Automata (Serras et al., 2019a; Serras, 2021; Torres, 2013). It is data-driven and works at the dialogue act level, and its goal is selected at the beginning of the simulations according to the goal probability distribution found in the corpus. We built a UM with the training data to learn RL policies, and a UM with the test data to evaluate the performance of all policies. It may happen that the dialogue act/slot combination output by the UM is not present in the corpus. In this case no audio signal can be sampled by the User Audio Sampler, and the simulation ends prematurely. This happens in 20% of the dialogues, but fortunately the sampling errors occur in the very first user turn almost exclusively (96% of the times), due to the constraint combination of the user goal not appearing in the dataset. This means that only very rarely computation time is wasted without adverse effects, in 0.8% of the simulations.

ASR/Transcription. Our experiments use three types of textual inputs: Manual transcriptions (TRSs), and two automatic speech recognition systems of different quality. The noisiest ASR (ASR 1) is based on a Wav2Vec2 network, where the `wav2vec2-base-960h` checkpoint was fine-tuned using 960 hours of Librispeech (Baevski et al., 2020). It achieved a Character Error Rate (CER) of 24.5 in the DSTC2 corpus, and a WER of 45.8. For the second and better performing ASR (ASR 2) we chose the best English model provided in the Vosk toolkit[1], the `vosk-model-en-us-0.22` model. The CER and WER errors were lower for this ASR, 10.1 and 21.0 respectively.

5.5.3 | Automatic evaluation metrics

Our dialogue policies are evaluated automatically via simulated dialogues using the test user model. Their performance is measured with three common task-completion metrics bounded between 0 and 1 (Serras, 2021; Kreyssig et al., 2018).

User Request Score (URS) indicates whether the system answers to the user in focus. It is the ratio between user informs answering a user request and user requests. For example, this score is high if the system provides an address after the user has requested it. This metric does not take into account, however, whether that address is correct or not, i.e., if it corresponds to the restaurant they are talking about or not. Whenever the user does not explicitly request any information, this score is not computed. This typically happens when the system provides information without the user requesting it.

System Offered Valid Venue (SOVV) indicates the correctness of system informs. It is the ratio between system informs that satisfy the constraints of the user and the total informs.

Can't Help Score (CHS) is only computed in a small fraction of the dialogues, about 20% approximately. Sometimes the UM has unreachable goals; for example, a user may want to find a Basque restaurant in the west of town, but there is none. In that case, the system should inform that there is no way to find such a restaurant. This score is 1 if the system provides this information, and 0 otherwise.

For simplicity and completeness, we use a combination of the three scores, which we call *Evaluation score*. Instead of defining it as the average of the three scores, we perform a weighted average with a lower weight for URS because it is a simpler task in which all the policies achieve close to perfect (>0.95) results. Lowering the URS weight gives more importance to the other two scores, which differ more across policies. In this way, the evaluation score aims to reflect more clearly the differences between policies. It is defined as:

$$\text{Evaluation score} = 0.2 \cdot \text{URS} + 0.4 \cdot \text{SOVV} + 0.4 \cdot \text{CHS}.$$

If any of the three scores above is not computed, the weights of the remaining

[1] https://alphacephei.com/vosk/models.

scores are increased proportionally to keep the score bounded between 0 and 1.

5.5.4 | Experiments overview

We carry out three sets of experiments. First, a detailed study of the effects of adding different audio embeddings to dialogue policies in different setups. We experiment with two ASRs and manual transcription for the text part of the network, and with three learning algorithms. We report the results in terms of automatic metrics in Section 5.6.1. Second, we compare different ways to add audio embeddings to dialogue policies, in Section 5.6.2. Third, in Section 5.6.3 we perform a human evaluation to further validate the results obtained in the first experiment, particularly the models that benefited the most by the inclusion of audio embeddings. Last, in Section 5.6.4, we inspect some of the resulting dialogue policies to further understand in which cases the decisions led by the audio part of the networks result in more successful dialogues.

5.5.4.1 | Training procedure

The automatic metrics are computed and averaged after a number of independent training runs to provide statistically meaningful results. The procedure followed to train and evaluate the different SL and RL policies is as follows:

I. We start by training text-only baselines for each input type (2 ASRs and TRS). For each input type, we train 6 different models and provide average results. Each model is evaluated with 5K dialogues with the test UM.

II. For each text-only SL model, we train each audio embedding part 5 independent times using SL. Thus, 30 (6×5) models are trained for each input type and audio embedding transformer (Wav2Vec2, HuBERT and UniSpeech-SAT). The GPT-2 network is kept intact in this stage to directly measure the impact of audio embeddings. Each model is evaluated with 1K dialogues with the test UM.

III. For each text-only SL model, run REINFORCE (Williams, 1992) and Actor-Critic (Konda and Tsitsiklis, 1999) RL algorithms 5 times without including any speech representations. As a result, 30 models are trained per text input type and RL algorithm. Each model is evaluated with 1K dialogues.

IV. Finally, re-train every output model of step two using REINFORCE and Actor-Critic and evaluate its performance with 1K dialogues. In this case, both the text and audio parts are trained jointly.

Thus, for each combination of learning algorithm, text input type, and audio embedding model or just text input, 30K test dialogues are obtained in total (6 models $\times$ 5K dialogues for the text-only baselines, 30 models $\times$ 1K dialogues otherwise). We also attempted training the RL policies from scratch without a SL baseline, but it was much harder to make them converge and the results were a lot poorer.

5.5.4.2 | Supervised Learning details

We use 4 epochs of SL training for the text only baselines, and 2 epochs when training the audio part only. A batch size of 4 is used throughout all the experiments, and the cross-entropy loss at the dialogue act level is minimised using the Adam optimiser with a learning rate of 5e-5.

5.5.4.3 | Reinforcement Learning details

REINFORCE and Actor-Critic are policy gradient RL algorithms that learn a set of weights θ in order select action a in state s according to policy $\pi_\theta(a|s)$. We designed three reward functions, two sparse ones (R_{s1} and R_{s2}) and a dense one (R_d), as follows:

$$R_{s1} = \begin{cases} 100 \cdot \textit{score} & \text{if end of dialogue,} \\ -1 & \text{otherwise,} \end{cases}$$

$$R_{s2} = \begin{cases} 100 \cdot \textit{score} & \text{if end of dialogue,} \\ -0.1 & \text{otherwise,} \end{cases}$$

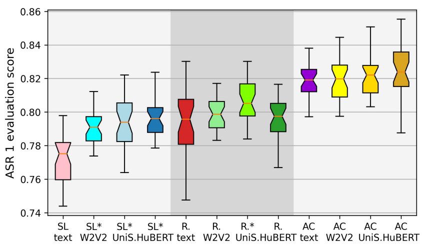
$$R_d = \begin{cases} 100 \cdot \textit{score} - 50 \cdot (1 - \textit{score}) & \text{if end of dialogue,} \\ 50 \cdot \ \textit{score} - 25 \cdot (1 - \textit{score}) & \text{if } \textit{score} \text{ is updated,} \\ -0.1 & \text{otherwise,} \end{cases}$$

where $score$ is the evaluation score described in Section 5.5.3. While the justification of R_{s2} is due to dialogue optimisation with less weight on dialogue length, R_d is motivated by using denser rewards as opposed to sparse ones. Preliminary experiments presented in Section 5.6.2.4 R_d clearly indicate that R_d is the most suitable reward function. We therefore used this one in almost all the experiments, and we will refer to it just as the *reward* unless otherwise stated.

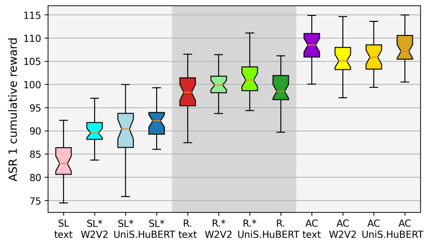
A discount factor of 0.95 and the Adam optimiser were used with a learning rate of 5e-6 in all the RL experiments. In the case of the Actor-Critic algorithm, the Actor (policy) and the Critic (estimated value function) use separate networks initialised with the resulting weights after the SL stage, except the linear predictor. We experienced some convergence problems with the actor-critic algorithm, which were solved by implementing two separate losses (one for the actor and the other for the critic). In addition to that, we used gradient clipping to prevent gradient exploding.

5.6 | RESULTS

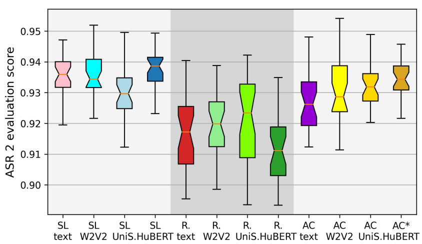
5.6.1 | AUTOMATIC EVALUATION OF THE DIALOGUE POLICIES



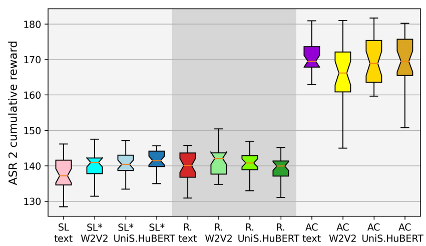
(a) ASR 1, evaluation score.



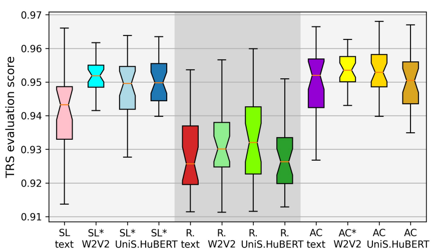
(b) ASR 1, cumulative reward.



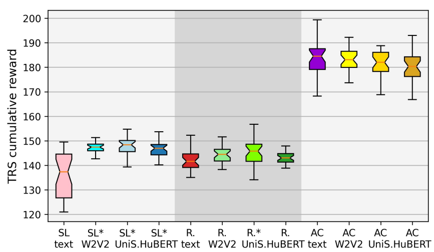
(c) ASR 2, evaluation score.



(d) ASR 2, cumulative reward.



(e) TRS, evaluation score.



(f) TRS, cumulative reward.

Figure 5.4.: Performance of dialogue policies on the test UM after Supervised Learning, REINFORCE and Actor-Critic with different audio embedding models.

Table 5.1.: Averaged evaluation metrics using the test UM after SL, REINFORCE and Actor-Critic, with different text inputs and audio embedding models. The policies with results in purple were part of the human evaluation.

	SL				REINFORCE				Actor-Critic			
	Text	+W2V2	+UniS.	+HuBERT	Text	+W2V2	+UniS.	+HuBERT	Text	+W2V2	+UniS.	+HuBERT
Evaluation score												
ASR 1	0.771	0.790*	0.792*	**0.795***	0.792	0.796	**0.805***	0.796	0.818	0.820	0.822	**0.823**
ASR 2	0.934	0.935	0.932	**0.937**	0.916	0.918	**0.920**	0.911	0.927	0.930	0.931	**0.934***
TRS	0.940	**0.951***	0.947*	0.948*	0.928	0.931	**0.932**	0.928	0.947	**0.953***	0.953	0.950
Cumulative reward												
ASR 1	83.2	89.5*	90.4*	**91.4***	97.3	100.5*	**101.4***	98.3	**109.2**	105.7	106.1	108.1
ASR 2	137.8	140.3*	141.1*	**141.6***	138.9	**141.0**	139.7	139.5	**169.1**	165.3	168.6	168.3
TRS	135.4	147.0*	**147.4***	146.1*	142.9	144.4	**146.3***	144.0	**183.9**	182.9	182.1	180.6
User Request Score (URS)												
ASR 1	0.945	0.962*	**0.975***	0.969*	0.958	0.964	**0.971***	0.967*	0.987	**0.988**	0.987	0.988
ASR 2	0.984	0.988*	**0.991***	0.991*	0.982	0.984	**0.988***	0.979	0.991	0.993	0.993	**0.993**
TRS	0.974	0.986*	**0.989***	0.987*	0.975	0.978	**0.981**	0.979	0.991	0.992	**0.992**	0.992
System Offered Valid Venue (SOVV)												
ASR 1	0.750	0.766*	0.762*	**0.768***	0.773	0.774	**0.783**	0.775	0.791	0.793	0.795	0.796
ASR 2	0.917	0.920	0.912	**0.921**	0.894	0.901	**0.902**	0.896	0.909	0.911	0.913	**0.917***
TRS	0.880	**0.938***	0.932*	0.935*	0.912	**0.919**	0.919	0.918	0.936	**0.943**	0.942	0.938
Can't Help Score (CHS)												
ASR 1	0.668	0.701*	**0.721***	0.703*	0.629	0.643	**0.651**	0.629	0.674	0.703*	**0.713***	0.695
ASR 2	0.967	**0.968**	0.965	0.966	**0.922**	0.906	0.905	0.895	0.940	0.942	**0.954***	0.695
TRS	**0.989**	0.988	0.978	0.983	**0.925**	0.915	0.915	0.887	0.943	0.963*	**0.965***	0.959*

Table 5.1 and Figure 5.4 show the performance of our learnt policies using the test UM according to evaluation scores and cumulative rewards. The bottom half of Table 5.1 shows results broken down into the three task completion metrics used to compute the evaluation score. The star symbol (*) indicates that values obtained using audio embeddings are significantly better than the ones corresponding to text only policies. More specifically, they mean that p-value$\leq$0.05 using the Welch's t-test, which tests whether two populations have equal means, without assuming equal variances. We use such a statistical test and p-value threshold in all the comparisons in this chapter. In addition to the above, Table 5.1 shows results of our dialogue policies using manual transcriptions (see values in grey)—n.b. those policies do not compete against the ones processing ASR outputs because they do not make decisions based on noisy inputs. The values in purple in this table correspond to the policies in the human evaluation described in Section 5.6.3.

These metrics show that including speech representations can help to learn better dialogues policies. But that depends on the learning algorithm and the quality of the text input. Regarding learning algorithms, SL policies clearly benefit the most by the inclusion of audio embeddings. This can be noted in SL policies including speech representations generated with either of the three audio embedding models, which significantly improve their performance. This is especially accurate for ASR 1, which suggests that audio embeddings are more beneficial in the case of noisier ASRs. Analysing the results per task completion metric, URS consistently improves significantly when using speech representations, SOVV in the case of ASR 1 and manual transcription, and CHS only with ASR 1-based text input.

RL-based policies do not benefit as much from audio embeddings as SL policies. The biggest improvement happens with REINFORCE and ASR 1 text inputs, where the evaluation score improves significantly by adding the speech information generated by UniSpeech-SAT, and so do the cumulative reward (with both Wav2Vec2 and UniSpeech-SAT embeddings) and URS (with UniSpeech-SAT and HuBERT). The improvements in Actor-Critic policies are much more scarce. Despite some exceptions, adding audio embeddings does not seem to help much. In fact, the absolute better results in terms of cumulative reward are obtained by polices that do not use audio embeddings—but the differences between policies using and not using audio embeddings are not significant. This is in contrast with REINFORCE, where the better results were always obtained with policies processing audio embeddings in addition to the text input (with some exceptions in the CHS metric), even though the differences in performance were not always statistically significant.

Why do audio embeddings help more when learning policies via SL instead of RL, especially with the noisier ASR? A dialogue policy trained via SL mimics the behaviour of the system in the corpus. Its performance is thus highly influenced by the information available at the decision making stage. If the training data contains a decision based on information that the learning policy lacks, it would hardly be able to imitate that action. Information can be lost, for instance, due to a poor ASR performance. Additionally, the ASRs employed to acquire the DSTC2 provided a top N list of hypotheses, which could also provide some information about the environmental noise (if the hypotheses differ greatly, for example). This lost information could be provided by the audio embeddings, and that could be the reason of the better performance of the SL policies that use them. In fact, as shown later in our study (see Table 5.7), the speech representations contribute to a better user understanding, which is reflected in a significantly lower amount of user requests per dialogue. This can also be seen in Table 5.1, which shows big improvements in the URS score in SL. Similarly, audio embeddings also encode information about the noise present in the user's audio signal, which can lead to relevant system requests of repetition that avoid misunderstandings.

In contrast, policies trained via RL learn their behaviour through interaction with the UM. This allows them to develop alternative strategies to avoid misunderstandings and deal with the input's noise. In general, such strategies, particularly the Actor-Critic ones, are more conservative than the SL policies. Briefly, the RL policies perform more confirms and ask the user to repeat their constraints more (one way or another) before trying to look for a suitable venue. Thus, these polices are less sensitive to the input noise (whether environmental or introduced by the ASR), and therefore benefit less from the inclusion of speech representations. We discuss this topic further in Section 5.6.3.

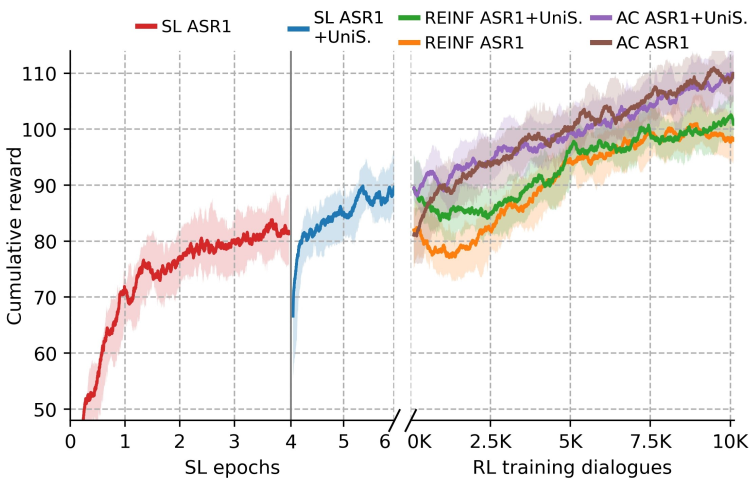


Figure 5.5.: Learning curves of dialogue policies with/without audio embeddings.

Figure 5.5 shows the evolution of the rolling average (over windows of 300 dialogues) of cumulative reward throughout the training process. It shows the learning curves corresponding to the ASR 1 text input and the UniSpeech-SAT audio embedding model—the same policies used in the human evaluation. Note that during SL the cumulative reward is not optimised explicitly. Instead, we minimise the dialogue act level cross entropy loss. But we show it for clarity and completeness. On the left hand side of the figure, we can clearly see the impact of adding audio embeddings during SL (blue vs. red curve). After the first two epochs of SL with only text input, the performance of that policy only improves slightly, indicating that there is not much more room for improvement. The audio part of the dialogue manager is then added after the fourth epoch is finished. Since the audio part of the linear predictor is randomly initialised, a drop in performance can be seen at the beginning of the fifth epoch, the first with audio embeddings. Shortly after that drop, the benefits of adding speech representations appear. The policy recovers its performance and improves much quicker than in the third or fourth epochs. At the end of epoch six, the cumulative reward was $\sim$10 points higher than epoch four. This is worth noting because the text part of the policy was kept untouched during the last two epochs. Thus, the improvements obtained in this period are due to the inclusion of audio embeddings only.

On the right hand side of Figure 5.5, we can see that RL policies on top of SL policies improve steadily their performance. But the differences between policies using and not using audio embeddings are largely reduced in RL. After several hundreds dialogues, the differences in the Actor-Critic policies vanish—this effect is not so sudden with REINFORCE. It can be seen that REINFORCE is more unstable than Actor-Critic, and only in the middle of the training process the policies using ASR 1 output only level up, on average. In the second half of training, the policies combining this input with the UniSpeech-SAT embeddings keep improving, though slightly, whereas the text only policies seem to have converged.

5.6.2 | Audio embedding and reward function comparison

5.6.2.1 | Which audio embedding model is best?

To answer this question Table 5.2 summarises the results shown in Table 5.1, but averaged over the algorithms and input types. It can be seen that UniSpeech-SAT performs slightly better than HuBERT and Wav2Vec2 respectively, as one could expect from previous comparison studies between these networks (Yang et al., 2021; Chen et al., 2021) applied to other tasks. But there are no statistically significant differences across these models in our task.

Table 5.2.: Performance comparison of audio embeddings in our task based on average results from Table 5.1.

	Wav2Vec2	UniSpeech-SAT	HuBERT
Evaluation score	0.892	**0.893**	0.892
Cumulative reward	135.1	**135.8**	135.4

Nevertheless and even if the three models perform similarly in our task, the best way to extract the audio dense vectors from those models can be further explored. To address that, we compared the cumulative reward obtained using the test UM when selecting different layers (or set of layers) from those models. As aforementioned, the output vectors of each selected layer were averaged over the time dimension to obtain easy-to-handle fixed-length vectors. We compared the output of each of the 12-layer transformers individually, and two combinations of 2 and 4 layers. The results are shown in Figure 5.6 with average rewards over three text input types on SL policies.

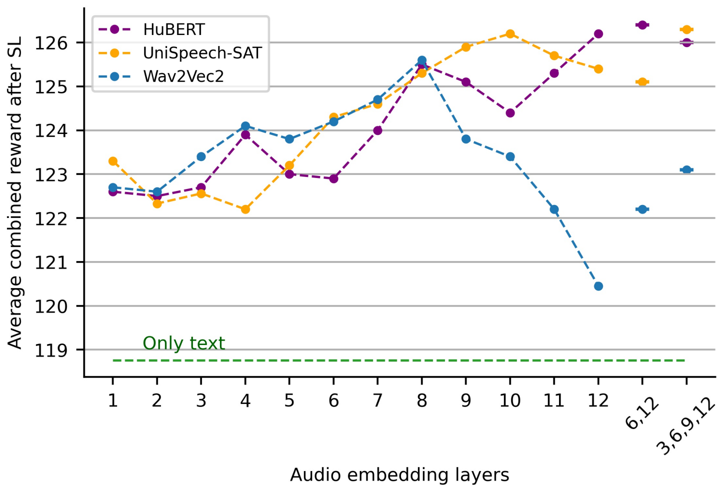


Figure 5.6.: Dialogue reward per neural layer of three audio embedding models.

It is worth mentioning that every combination of output layer and embedding model outperforms the results obtained with text only. Besides, HuBERT and UniSpeech-SAT follow a similar pattern, which is reasonable since UniSpeech-SAT was trained inspired by the methodology used to create HuBERT (Chen et al., 2021). In both cases the worst results are obtained with shallow layers, and the best ones with the last five layers. Moreover, the tested combinations of layers work quite well, presumably because they include layers that worked well individually in both cases. In contrast, Wav2Vec2 has a clear drop in the performance from layer 9 onwards. Consequently, the combinations of layers we tested did not perform too well because they include the last layer. This comparison was performed at the beginning of our experimentation to select the output layers for each model according to Figure 5.6, and kept them unchanged during the rest of experiments presented in this work. Specifically, the 8th layer was selected for Wav2Vec2; the combination of the 6th and 12th layers for HuBERT and the set of the 3rd, 6th, 9th and 12th layers for UniSpeech-SAT.

Table 5.3.: Performance of SL dialogue policies using the test UM after fine-tuning the audio embedding models.

	Wav2Vec2		UniSpeech-SAT		HuBERT	
	Fine-tuned	Frozen	Fine-tuned	Frozen	Fine-tuned	Frozen
Evaluation score						
ASR 1	0.780	**0.790***	0.778	**0.792***	0.779	**0.795***
ASR 2	0.933	**0.935**	**0.933**	0.932	0.936	**0.937**
TRS	0.940	**0.951***	0.940	**0.947***	0.941	**0.948***
Cumulative reward						
ASR 1	85.6	**89.5***	85.0	**90.4***	85.5	**91.4***
ASR 2	137.8	**140.3***	137.0	**141.1***	138.0	**141.6***
TRS	140.3	**147.0***	140.4	**147.4***	140.8	**146.1***

5.6.2.2 | How about fine-tuning the audio embedding models instead of keeping them frozen?

Throughout this chapter so far, the audio embedding models have been kept frozen and only the linear predictors on top of them have been trained. As an additional experiment, also using SL only, we explored an alternative methodology that consists of using the last output vector (in the time dimension) of the last transformer layer as a summary of the whole input. Since this vector should contains less information than over all the time steps, the transformer is fine-tuned while training. This way it should learn to include all the relevant information in that final vector. Table 5.3 shows performance results compared to those described in Table 5.1. It can be observed that transformers with frozen layers outperform the fine-tuned ones. Nonetheless, we should remark that even if this alternative strategy seems to be worse than the main one, the results obtained with it are still better than with a text input only. This means that the alternative strategy can be valid, but the main one seems to be better in this experimental setup. We hypothesise that the better performance of the frozen audio embedding models might be due to the following reasons: 1) the last output vector in the time dimension contains notoriously less information than the average over time, and a simple fine-tuning with a small amount of data is not enough to train the network effectively to encode all the necessary information in that vector; and 2) fine-tuning involves the training of an exponentially larger amount of parameters, which could cause training problems such as overfitting, especially due to the limited amount of training data.

5.6.2.3 | Audio embeddings versus ASR confidence

We hypothesise that dialogue policies benefit from speech representations in two main ways. First, they allow a better semantic understanding of the user in many occasions, i.e. they can help to recognise what kind of information is being requested. This is supported by the experiments and examples we discuss below in Section 5.6.4 (Figure 5.9a). Second, audio embeddings also provide information about the intelligibility of audios, and thus should be able to inform the system when a turn has been noisy and the ASR transcription might not be very reliable. This information can also be introduced in the system via ASR confidence scores. In fact, if we substitute audio embeddings by the average and standard deviation of the character level ASR confidence (for ASR 1) in SL policies, an evaluation score of 0.780 can be obtained. This value is higher than for the text only baseline (0.771, Table 5.1), but is still far from the best results obtained with audio embeddings (0.795, Table 5.1). This suggests that speech representations not only include information about the potential ASR uncertainty, but also additional semantic information that allows dialogue policies to perform even better. Future works could confirm this result in other scenarios, tasks or datasets.

5.6.2.4 | Reward function comparison

We also explored three different reward functions for RL, as aforementioned in Section 5.5.4.3. Table 5.4 shows the evaluation score obtained in the test user model for 4 policies trained with REINFORCE and using the three reward functions. Since this experiment was carried out in an early stage of the research project (López Zorrilla et al., 2021b), the policies take as input the ASR 1 output or the manual transcription, and use (or not) the Wav2Vec2 embeddings (fine-tuned). The values were computed using one text baseline only. As expected, the best results were obtained with the dense reward function, and therefore this function was the one used in the rest of the experiments.

5.6.3 | Human evaluation

We further validated the results obtained in Section 5.6.1 via a human evaluation. Since these results indicate that audio embeddings help the most with the noisiest ASR, ASR 1, we compared policies using ASR 1 transcriptions without

Table 5.4.: Evaluation score obtained after training REINFORCE policies with three reward functions.

	ASR 1	+W2V2 (fine-tuned)	TRS	+W2V2 (fine-tuned)
R_{s1}	0.769	0.780	0.895	0.892
R_{s2}	0.770	0.786	0.895	0.925
R_d	**0.787**	**0.807**	**0.922**	**0.935**

and with audio embeddings. The latter are based on UniSpeech-SAT due to better performance, see Table 5.2. We thus compare 3 pairs of policies: a policy processing only the ASR 1 versus another that also uses UniSpeech-SAT speech representations after training them via SL, REINFORCE and Actor-Critic. We do not compare other combinations because a human evaluation is much more costly than an automatic one.

Six judges (knowledgeable in the area of SDSs) evaluated 82 dialogues evaluated for each of our six policies—resulting in 492 dialogues each judge, 2952 dialogues in total. The evaluation was carried out using the CrowdZientzia platform (Justo et al., 2016). Both the manual and ASR 1 transcriptions were shown in each of the users turn to allow the judges to assess the dialogues properly, similar to the examples in Figure 5.9, which we analyse in Section 5.6.4. The judges were not aware of which policy had carried each dialogue to avoid any bias. After reading and analysing a dialogue, judges were asked to fill the multiple-choice 3-question questionnaire shown in Table 5.5, adapted from (Keizer et al., 2021). In the questionnaire, Q1 is related to the SOVV and CHS scores described in Section 5.5.3, Q2 to the URS score, and Q3 is the most subjective question regarding dialogue naturalness. Table 5.5 also shows the possible answers to each question, as well as their conversion to scalar ratings.

Table 5.6 shows the averaged results of the human evaluation. We measured the inter-rater reliability with the Krippendorff's Alpha coefficient (Krippendorff, 2018), with the interval metric (Krippendorff, 2011) as the difference function. The values were $\alpha_{Q1} = 0.715$, $\alpha_{Q2} = 0.802$, and $\alpha_{Q3} = 0.742$, which indicate a high agreement among the judges. Overall, the human evaluation supports and complements the conclusions drawn from the automatic evaluation. First, the greatest improvements come after SL, as expected from previous analysis: the policy using UniSpeech-SAT speech representations obtains a significantly higher score in the three questions, and also on average. Second and in the case of policies trained via REINFORCE, the policy using audio embeddings improves too—but in this case the differences are not significant. Something similar happened with the automatic metrics, where only in some cases (with some audio embedding models) the cumulative reward or the evaluation score improved significantly. This indicates again that audio embeddings help in REINFORCE, but not always. Last and unsurprisingly, the gap is even nar-

Table 5.5.: Questionnaire used by judges in the human evaluation.

Q1: The system offered a restaurant satisfying the user constraints, or correctly informed that there were no such restaurants.
- Yes. (1)
- No. (0)

Q2: The system provided the information the user was looking for (phone number, post code, address...).
- Yes. (1)
- Partially. (0.5)
- No. (0)
- None–if there are no user requests.

Q3: The conversation felt natural.
- Strongly agree. (1)
- Agree. (0.75)
- Neither agree nor disagree. (0.5)
- Disagree. (0.25)
- Strongly disagree. (0)

Table 5.6.: Human evaluation results.

#	Algo.	Input	**Q1**	**Q2**	**Q3**	**Avg.**
1	SL	ASR1	0.656	0.848	0.535	0.629
2	SL	ASR1+UniS.	**0.760***	**0.902***	**0.601***	**0.716***
3	REINF	ASR1	0.730	0.892	**0.637**	0.716
4	REINF	ASR1+UniS.	**0.762**	**0.901**	0.632	**0.721**
5	AC	ASR1	0.761	**0.919**	**0.605**	0.718
6	AC	ASR1+UniS.	**0.789**	0.907	0.585	**0.719**

rower when using Actor-Critic as the learning algorithm. In this case, the differences are rather marginal, as happened when measuring their performance with automatic metrics.

Table 5.6, on the other hand, also helps to gain a deeper insight into some other aspects of the policies, especially if we focus on Q3. While Q1 and Q2 focus on task completion, Q3 has more to do with how naturally they complete the task. If we rank the policies only in terms of Q1 and Q2, the ranking is very similar to the one obtained with the automatic evaluation:

- the Actor-Critic policies are the best,
- then REINFORCE with audio embeddings,
- then the SL policy with audio embeddings and the REINFORCE policy processing only text, and finally
- the SL policy based only on the ASR 1 output.

But in terms of naturalness, Actor-Critic policies are worse than the REIN-

FORCE ones, and AC+ASR1+UniS. is even less natural than SL+ASR1+UniS. This is due to Actor-Critic being a better RL algorithm in this task. The resulting policies thus exploit the UM as much as possible, leading to dialogue strategies that are not perceived as so natural. Some of these behaviours can be inferred from Figure 5.7, which shows the average and standard deviation of the frequency of each dialogue act per dialogue. Note that some dialogue acts have been grouped to make the figure clearer. For example and since ASR 1 is noisy, Actor-Critic policies learn to extract as much information as possible from the UM, making it repeat the constraints multiple times. The more frequent use of the `Repeat` dialogue act also leads to less (but more accurate) API calls.

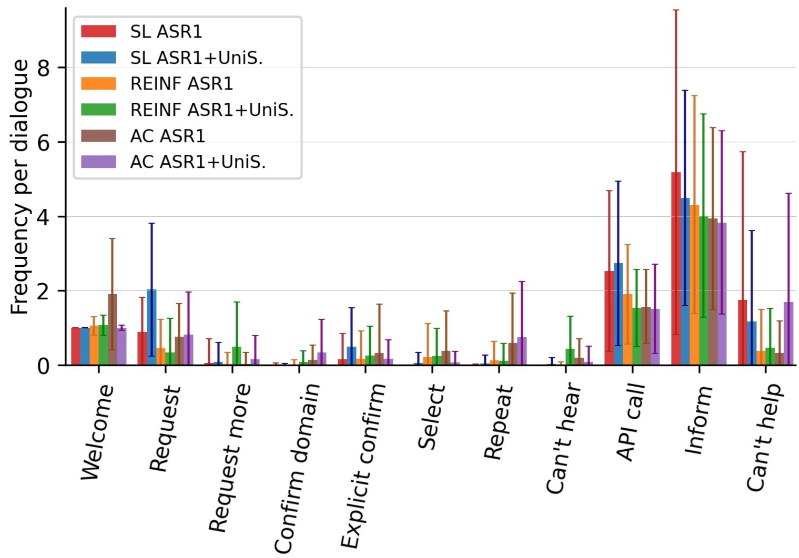


Figure 5.7.: Dialogue act histogram (with std) comparing six dialogue policies.

The fact that Actor-Critic dialogue policies confirm information provided by the user multiple times to reduce misunderstandings (even with text input only), suggests that other factors (such as amount of repetitions) should be considered in the employed reward function—or the use of learnt rewards. These suggestions could help to realise the full potential of audio embeddings for RL-based dialogue policies in the future.

5.6.4 | Manual inspection

In this section, we aim at identifying how and when audio embeddings lead to better performance. To this end, we generated and analysed a number of simulated dialogues with policies that share the text processing part. Therefore, if they select a different dialogue act given the same context, it is only due

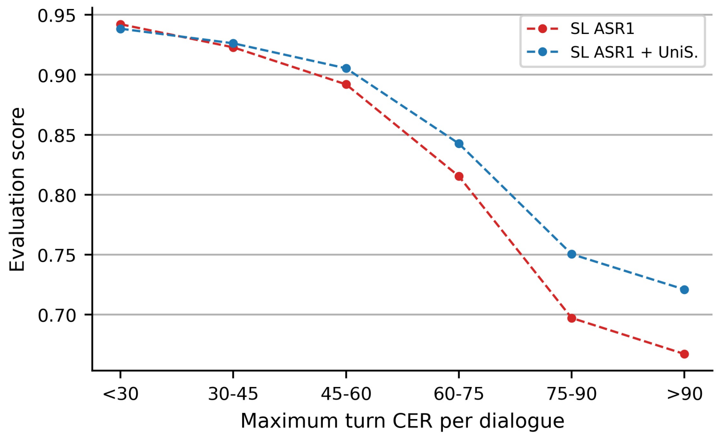
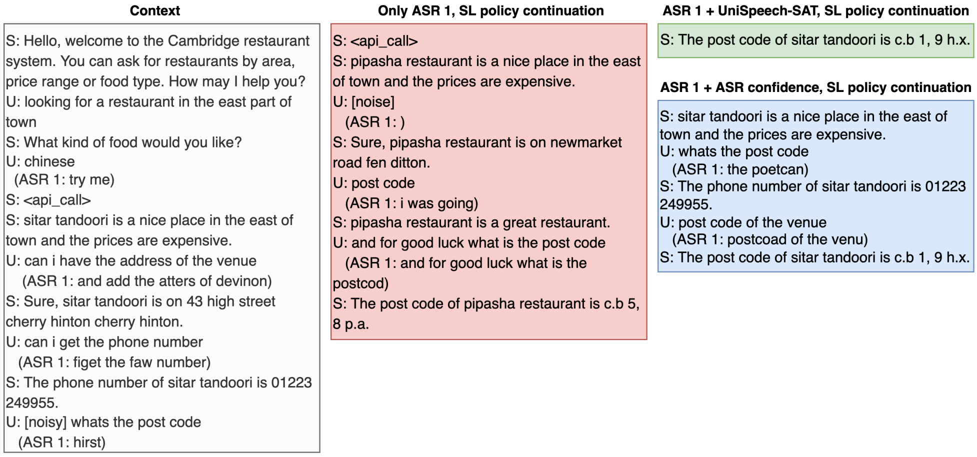


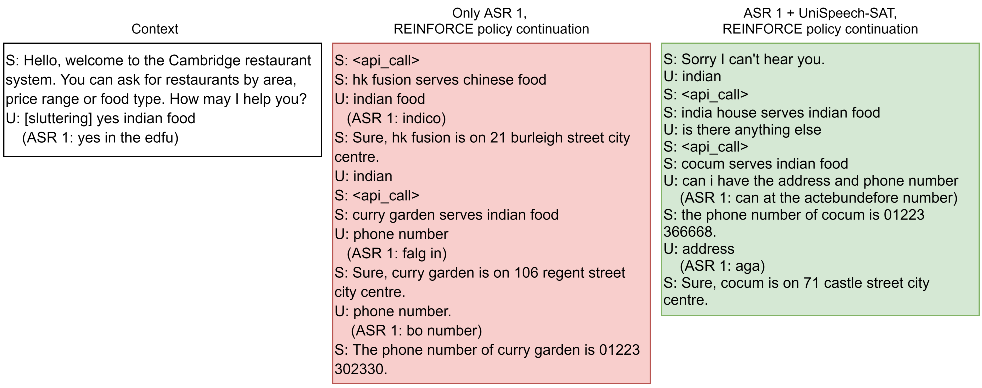
Figure 5.8.: Evaluation score of SL policies depending on the maximum turn CER (Character Error Rate) per dialogue.

to the audio embeddings. In many cases, dialogue strategies develop similarly whether they use policies with or without speech representations. This happens especially when the ASR transcriptions are more accurate. But in cases where the ASR output is poor, audio embeddings provide crucial information absent in the ASR transcription, allowing the policies perform better. This can be seen in Figure 5.8, where the correlation between the evaluation score of SL policies and the maximum turn CER per dialogue is plotted. The higher the CER, the more the policies benefit from audio embeddings. Figure 5.9 shows two simulated dialogues with poor ASR outputs where audio embeddings help to perform better actions.

The first example (Figure 5.9a) is particularly representative, where we can see a typical conversation between the UM and the dialogue manager. The dialogue goes quite smoothly until a breaking point occurs when the UM requests the post code of the offered venue, in a rather noisy turn where the ASR 1 outputs "*hirst*". The text only policy (red box) performs an additional API call, and after some repetitions finally provides a post code, but it corresponds to the second restaurant it searched. Conversely, the policy processing the user's audio via the UniSpeech-SAT network (green box) is able to understand the user's intent even after the "*hirst*" turn, successfully providing the post code of the first venue it had offered. Thus, the dialogue ends in a much more natural manner. Additionally, Figure 5.9a shows the continuation of a policy that uses the ASR confidence as input (commented in Section 5.6.2.3). The low ASR confidence in the noisy turn prevents the policy from performing an API call, and it performs a safe inform instead. After another two post code requests, the system finally retrieves the desired information.



(a) Two SL policies, with exactly the same text processing part.



(b) Two REINFORCE policies, based on the same text SL baseline.

Figure 5.9.: Sample dialogues where the policy including speech representations carries out a more successful dialogue. The context is the same for both policies.

The second example compares the two REINFORCE policies judged in the human evaluation. Although both policies were trained on top of the same SL baseline, the two policies do not share completely the text processing part (because both the text and audio processing parts of the dialogue managers were trained jointly in RL experiments with audio embeddings). The example is still illustrative nonetheless. The initial user's message is not clear, due to a stuttering. After that, the text only REINFORCE policy performs an API call, but the found venue does not satisfy the user's requisites, because that first turn was not clear enough. Eventually the system corrects itself and finds a suitable venue, but the conversation is more messy than the continuation of the policy using audio embeddings. This one does not make an API call immediately, instead, it asks the user to repeat the sentence, because it is probably aware that the user turn was not understood. In addition to that, the text only policy is not able to understand that the user requests the phone after the ASR 1 transcription "*falg in*", and the user is forced to ask for it again. The policy with audio embeddings, on the other hand, is able to provide the address after the turn with ASR 1 output of "*aga*", which further consolidates the hypothesis that the speech representations lead to a better understanding.

Such a better understanding can be measured by the average number of requests by the UM per dialogue. As shown in Table 5.7, including audio embeddings leads to less number of requests per dialogue, especially after SL, where the difference is statistically significant. This stays on-track with the rest of the results obtained in our work: including audio embeddings helps the most when training the policies via SL.

Table 5.7.: Request repetitions by the UM per dialogue with the policies used in the human evaluation, averaged over 1K dialogues.

	ASR 1	+ UniSpeech-SAT
Supervised Learning	1.420	**1.258***
REINFORCE	1.411	**1.315**
Actor-Critic	1.183	**1.175**

Last but not least, we analyse the audio embedding layers' contribution to select the dialogue act in the examples in Figure 5.9. Since UniSpeech-SAT was used, the output of four layers is taken into consideration: the 3rd, 6th, 9th and 12th, as explained in Section 5.6.2. We can easily compute how much each layer contributed to take the final decision. To this end, we take the output of the linear predictors that process the averaged embeddings of each layer, and select the value corresponding to the predicted dialogue act. This value is the layer's contribution to the unnormalised probability of taking that action. The contributions are shown in Figure 5.10.

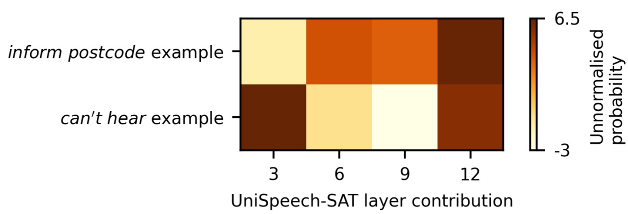


Figure 5.10.: Layer contribution to the decision taken in the example conversations.

In the *first example*, where the system correctly understands that the user is requesting a post code, the last layer has the biggest contribution, the two intermediate ones contribute less, and the shallower 3rd only influences the action barely. The layer contribution is different in the *second example*, in which the system informs the user that it cannot hear correctly. In this case, the biggest contributions come from the 3rd and last layers. This makes sense since shallow layers operating closer to the audio signal are known to learn mostly speaker and environmental information, while the last layers contextualise more to learn content and semantic information (Chen et al., 2021). Therefore and whilst a greater contribution of the intermediate and last neural layers are related to a better understanding, a greater contribution of the shallow neural layers can be interpreted as the system being aware of some anomalies at the signal level such as those exhibited by noise or stuttering (among others).

5.7 | Conclusion and future work

We present an in-depth study to analyse under which conditions speech representations (via audio embeddings) help to learn better dialogue policies in the context of the DSTC2 corpus. They help to understand the user better or to inform the system when the user might not be well understood—especially with the noisier ASR prone to providing inaccurate transcriptions. This effect is clearer when training the policies with supervised learning, because reinforcement learning algorithms are able to exploit the UM better and learn strategies to deal with the uncertainty in the text input more successfully.

We hypothesise that our approach could be very helpful in other demanding spoken dialogue tasks where the user is difficult to understand, even with very high quality ASRs. Some examples include noisy industrial environments (Aceta et al., 2022), SDSs integrated in cars (Schmidt et al., 2019), and also systems that interact with non-native users or users with strong local accents (Litman et al., 2018). These are target domains for the dialogue community, and we hope that our findings can help to develop SDSs of higher quality in the future in these areas.

Finally, other potential future works could take advantage of the latest advances in Speech Synthesis to continue our research with modern—and only text-based—dialogue corpora. The proposed User Audio Sampler could be replaced by a high-quality TTS module. This has been successfully attempted in end-to-end Spoken Language Understanding recently (Lugosch et al., 2020). Not only would it allow to test our approach on more challenging dialogue tasks, but it could also further validate our conclusions if user responses could be simulated taking into account specific background noise, local accents, backchannels, and/or emotions. In those cases, the speech representations should contain information absent in the ASR output and substantially boost the performance of SDSs.

Chapter 6

Conclusions and future work

We have presented several improvements to alleviate some of the issues of neural dialogue models, in different tasks and frameworks. At the end of each chapter, we have discussed the implications, benefits, limitations and potential future work related to each proposal in depth. Next, we summarise our main contributions and conclusions.

6.1 | Conclusions

In open-domain dialogue generation, we have addressed lack of variety in the generated responses in Chapter 2, with a novel methodology to train text-based GANs, and extension of conventional response-level GANs to the batch level. The idea of batch-level discriminators should be relevant to other GAN architectures and tasks, not only to dialogue-related GANs. In general, many NLP systems that use any kind of discriminators could also benefit from our proposals, such as GPT-2 based dialogue managers (see Chapter 4) or BERT-based question answering systems (Devlin et al., 2019).

We have taken part in the EMPATHIC project (Chapter 3), where a modular VC capable of carrying out coaching sessions has been developed. Even if data-driven methodologies for the DM and NLG were considered initially, these modules were finally developed with mostly rule-based approaches. This highlights that the lack of control inherent to statistical models is something that needs attention in the future, especially for commercial or industrial applications.

The WDH system presented in Chapter 4 has shown to be a nice method to provide neural dialogue models with a higher capability of implementing long-term dialogue strategies. Even if we have carried out our experiments in the context of coaching, this strategy is generic enough to be implemented for any task where dialogues follow a specific structure. The positive effects of applying this methodology in other tasks should be clearer the more structured and longer the target dialogues are. Additionally, our proposal could be easily combined with the newest transformers for language (and therefore, dialogue) generation, such as GPT-3 (Brown et al., 2020) or PaLM (Chowdhery et al., 2022). These powerful transformers could improve the candidates that the WDH system ranks, but also enhance the WDH system itself by providing more accurate sentence embeddings.

Last, the idea of including audio processing pretrained transformers in the pipeline of task-oriented SDSs has shown to be beneficial for dialogue policy learning. The improvement has been significantly higher in the cases where the ASR transcription is not too accurate (Chapter 5), and when using SL as opposed to RL to train the policies. Thus, our findings open a new research direction, which could be particularly important for SDSs in situations where ASRs do not work so well, such as noisy environments, when the users have uncommon accents, or when working with low resource languages.

6.2 | FUTURE WORK

Partly due to having explored many research ideas about dialogue modelling, our proposals have been evaluated and analysed only in one task each. Thus, despite our results supporting the potential of these contributions, future works should ratify our conclusions in more experimental conditions and comparing to a larger number of state-of-the-art models. Next, we detail potential future works to further validate our proposals and discuss some research possibilities related to our studies.

Future works for dialogue generating GANs (see Chapter 2) could compare our top-k softmax approach to circumvent the differentiability problem of text GANs with other alternatives that can be found in the literature, such as the (Straight-Through) Gumbel-softmax or RL rewards at the token level. This way, not only could we conclude that the top-k softmax is a valid approach to train dialogue GANs, but also understand better its upsides and downsides. As for the batch-level discriminator, it would be interesting to compare it to the permutation-invariant discriminator proposed by Lucas et al. (2018). Instead of using a batch of samples to predict whether they are *real* or *fake*, their mini-batches contain a variable proportion of real and fake samples, and they train their discriminator to predict this proportion.

On the other hand, the EMPATHIC VC (in Chapter 3) has also shown great potential according to human evaluations. However, even though end-users found it useful (with scores of 66.7/100 in Spanish, 62.0/100 in French and 54.8/100 in Norwegian), its actual impact on the target population was not measured. Future experiments (maybe with a more advanced system) should analyse the evolution of health and well-being-related metrics of the participants over time. To this end, a system capable of carrying out multiple coaching sessions would be needed, which would probably require a larger engineering effort.

Other potential future works are also related to our contributions to EMPATHIC. The end-to-end coaching chatbot described in Chapter 4 uses scenario embeddings that indicate which kind of dialogue has to be carried out. The EMPATHIC corpus showcases two scenarios (introduction and coaching about nutrition), but other corpora such as MultiWOZ (Budzianowski et al., 2018) or SGD (Rastogi et al., 2020) include 7 and 16 domains, respectively. Using such embeddings in these tasks may lead to higher performance gains. Additionally, the dialogue phase embeddings and, more importantly, the WDH system could also boost the performance of dialogue models in many tasks besides EMPATHIC. A comprehensive study analysing in which tasks these contributions help the most and how the quality of the candidates and the sentence embeddings affect the performance would be really interesting. Furthermore, future works could also determine in which kind of dialogue states the WDH's effect is more relevant. Possibly, more structured dialogue stages could benefit more from our proposal, whereas the performance in more open sections might not improve so much. This could be measured with metrics such as the sentence selection accuracy (see Section 4.7.1), among others.

Regarding our proposed approach for speech-aware dialogue policy learning using audio embeddings, potential future works could take advantage of the latest advances in Speech Synthesis to continue our research with larger—and only text-based—dialogue corpora. The proposed User Audio Sampler could be replaced by a high-quality TTS module (Soltau et al., 2022). This has been successfully attempted in end-to-end Spoken Language Understanding recently (Lugosch et al., 2020). Not only would it allow us to test our approach on more challenging dialogue tasks, but it could also further validate our conclusions if user responses could be simulated taking into account specific background noise, local accents, backchannels, and/or emotions. In those cases, the speech representations should contain information absent in the ASR output and substantially boost the performance of SDSs.

Finally, we would like to remark the difficulty of evaluating dialogue systems, which affects a big part (if not all) of the research presented in this thesis. On the one hand, the information provided by automatic metrics is limited and they do not always correlate with the actual performance of the system. On the other hand, human evaluation is much more reliable, but also costly and/or time-consuming at the same time. This is why we have tried to present several automatic metrics to test each of our proposals: four variety metrics plus a semantic similarity metric in Chapter 2, two word-overlapping metrics plus the next sentence selection accuracy in Chapter 4, and three task completion metrics in Chapter 5. The selection of these metrics has been strongly related to the task and to the proposals that we have evaluated. However, it would be useful to further increase the number of metrics in related future works, or refine them. For example, the next sentence selection accuracy (in Chapter 4) could be complemented with a semantic similarity metric to assess how much each of the models selects sentences with similar meanings to the ground truths. This highlights the importance of research in dialogue evaluation, both human and automatic. More efficient human evaluation procedures could reduce their cost and thus allow a more significant number of evaluations; and better and more robust automatic metrics would reduce the dependence on human evaluations and indirectly lead to better dialogue policy learning, at the same time.

With this, we hope that this thesis represents a step towards bridging the gap between the conversational skills of humans and those of machines.

List of publications

Let us list the publications we have produced as a result of the research described in this thesis, in reverse chronological order, from most recent to oldest.

Publications

Vázquez, A., López Zorrilla, A., Olaso, J. M., & Torres, M. I. (2023). Dialogue management and language generation for a robust conversational virtual coach. *Sensors (Accepted paper)*.

López Zorrilla, A., Vázquez, A., & Torres, M. I. (2023). Batch-level GANs to promote dialogue response variety. *13th International Workshop on Spoken Dialogue Systems Technology (Accepted paper)*.

López Zorrilla, A., Torres, M. I., & Cuayáhuitl, H. (2023). Audio embedding-aware dialogue policy learning. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 31, 525–538. https://doi.org/10.1109/TASLP.2022.3225658

López Zorrilla, A. & Torres, M. I. (2022). A multilingual neural coaching model with enhanced long-term dialogue structure. *ACM Transactions on Interactive Intelligent Systems*, 12(2). https://doi.org/10.1145/3487066

López Zorrilla, A., Torres, M. I., & Cuayáhuitl, H. (2021). Audio embeddings help to learn better dialogue policies. *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 962–968. https://doi.org/10.1109/ASRU51503.2021.9688296

Olaso, J. M., Vázquez, A., Ben Letaifa, L., De Velasco, M., Mtibaa, A., Hmani, M. A., Petrovska-Delacrétaz, D., Chollet, G., Montenegro, C., López Zorrilla, A., et al. (2021). The EMPATHIC virtual coach: a demo. *Proceedings of the 2021 International Conference on Multimodal Interaction*, 848–851. https://doi.org/10.1145/3462244.3481574

Justo, R., Letaifa, L. B., Olaso, J. M., López Zorrilla, A., Develasco, M., Vázquez, A., & Torres, M. I. (2021). A Spanish corpus for talking to the elderly. *Conversational Dialogue Systems for the Next Decade*, 183–192. Springer. https://doi.org/10.1007/978-981-15-8395-7_13

López Zorrilla, A., deVelasco Vázquez, M., & Torres, M. I. (2021). A differentiable Generative Adversarial Network for open domain dialogue. *Increasing Naturalness and Flexibility in Spoken Dialogue Interaction*, 277–289. Springer. https://doi.org/10.1007/978-981-15-9323-9_24

López Zorrilla, A., deVelasco Vázquez, M., & Justo, R. (2020). Euskarazko elkarrizketa sistema automatikoa sare neuronalen bidez [A neural dialogue system in Basque]. *Ekaia. EHUko Zientzia eta Teknologia aldizkaria*, 37. https://doi.org/10.1387/ekaia.20987

Montenegro, C., López Zorrilla, A., Mikel Olaso, J., Santana, R., Justo, R., Lozano, J. A., & Torres, M. I. (2019). A dialogue-act taxonomy for a virtual coach designed to improve the life of elderly. *Multimodal Technologies and Interaction*, 3(3), 52. https://doi.org/10.3390/mti3030052

López Zorrilla, A., De Velasco Vázquez, M., & Justo Blanco, R. (2019). Euskaraz hitz egiten ikasten duten makina autodidaktak [Basque speaking self-learning machines]. *Proceedings of IkerGazte 2019*, 125–132. https://doi.org/10.26876/ikergazte.iii.03.16

López Zorrilla, A., de Velasco Vázquez, M., Irastorza Manso, J., Olaso Fernández, J. M., Justo Blanco, R., & Torres, M. I. (2018). EMPATHIC: Empathic, expressive, advanced virtual coach to improve independent healthy-life-years of the elderly. *Procesamiento de Lenguaje Natural*, 61, 167–170. https://doi.org/10.26342/2018-61-24

Additional publications

Besides our work in dialogue modelling, we have also collaborated with some colleagues in other research projects throughout the duration of the PhD thesis. These collaborations have focused on NLG, emotion analysis from audio and focus detection from speech.

Publications

Vázquez, A., López Zorrilla, A., & Torres, M. I. (2023). How should we represent dialog acts to leverage pretrained natural language generators? *13th International Workshop on Spoken Dialogue Systems Technology (Accepted paper)*.

deVelasco, M., Justo, R., López Zorrilla, A., & Torres, M. I. (2023). Analysis of deep learning-based decision-making in an emotional spontaneous speech task. *Applied Sciences*, 13(2). https://doi.org/10.3390/app13020980

deVelasco, M., Justo, R., López Zorrilla, A., & Torres, M. I. (2022). Automatic analysis of emotions from the voices/speech in Spanish TV debates. *Acta Polytechnica Hungarica*, 19(5). https://doi.org/10.12700/APH.19.5.2022.5.8

deVelasco, M., Justo, R., López Zorrilla, A., & Torres, M. I. (2019). Can spontaneous emotions be detected from speech on TV political debates? *2019 10th IEEE International Conference on Cognitive Infocommunications (CogInfoCom)*, 289–294. https://doi.org/10.1109/CogInfoCom47531.2019.9089948

López Zorrilla, A., deVelasco, M., Cenceschi, S., & Torres, M. I. (2018). Corrective focus detection in Italian speech using neural networks. *Acta Polytechnica Hungarica*, 15(5). https://doi.org/10.12700/aph.15.5.2018.5.7

Appendix A

Wizard of Oz Scenarios

A.1 | General purpose turns

Table A.1.: Set of predefined general purpose turns for both the introductory and nutrition dialogues.

Yes.	That's it.
No.	Not really.
Please.	Why?
Uummm...	Ups...
Yeah	Sure.
Sorry for that.	I see
Really?	I can't believe it.
Tell me more about that.	I'm listening.
Turn up the microphone volume, please.	Lower the microphone volume, please.
Speak louder, please.	Speak lower, please.
Very interesting.	We'll talk about that later.
Fantastic.	Awesome.
Can't believe it.	Well, I'm glad.
How good.	That is something only you know.
What a pity.	I'm sorry.
It does not always rain to everyone's taste.	You know that better than me.
That question is too difficult. I can not answer it.	Think a little, and I'm sure you can come out with of something.
No hurry.	Good, I think we are moving forward.
I do not feel comfortable talking about that.	

A.2 | Introductory scenario

Table A.2.: Set of predefined turns for the system presentation in the introductory dialogue.

Hi, how are you doing?
Good morning
Good afternoon.
Good evening.
What's your name?
Could you please spell your name?
Could you please spell it?
Have I pronounced it correctly?
How are you?
How are you NAME?
I'm Natalie, nice to meet you.
I'm learning to talk to people. Please be patient.
I'm a system that is learning coaching tasks.
I am a virtual agent under development.
I am a virtual agent under development.
I'm part of the EMPATHIC project, in which universities, public centers and companies from several countries, mainly from Europe, collaborate.
Spain, France, United Kingdom, Italy, Norway, Belgium and Israel.
Do you know what coaching is?
What do you think coaching is?
If I have not already mentioned it to you, I am learning coaching, but before getting into this, I would like to get to know you a little more.
What made you come and talk to me?
Which are your main hobbies?
What do you enjoy doing in your free time?
Do you like travelling?
Do you like music?
Do you like eating?
And do you do it often?
I'm specially interested in music and travelling.
Which of these topics do you prefer to chat about?
Which of these topics do you like the most?
Perfect, I also wanted to talk about it!

Table A.3.: Set of predefined turns about travelling in the introductory dialogue.

As you can imagine, I do not travel much. I have to settle for what you tell me.
Since I live on the web, I've been around the world, but I can not see many things, only what you show me on the webcam.
So, do you like to travel?
What has been the place you liked the most?
What struck you the most?
When was that?
What envy are you giving me.
Do you usually travel in company?
I am usually alone. But I've got used to it and I'm well.
Imagine that you are there, in those busy streets. What do you feel? How do you imagine?
Imagine that you are once again surrounded by all that peace. How you feel?
Sorry to be a little gossip, do you have something you like to do whenever you go on vacation?
Come on, tell me some anecdote about it.
Interesting, could you tell me any anecdote?
Do you have a new trip in mind?
Where?

Table A.4.: Set of predefined turns about music in the introductory dialogue.

Since I am always on the Internet, I can listen to all the music I want, it's the good thing about being a virtual system.
And why have you chosen to talk about music?
Why do you prefer to talk about music?
So, you like music, don't you?
Right, why do you like it so much?
Do you like many styles, or you just prefer one?
I'm specially concerned about current commercial music. What do you think?
I also prefer more alternative genres.
Well, you really convinced me. The important thing is something that encourages us.
Do you usually dance when listening to music?
Have you been at any concert or musical event lately?
Did you watch Eurovision?
What do you think about it?
I think ABBA times were better. Do you know that they have met again?
I see you are very interested in music, do you play any instrument?
Have you been playing it for a long time?
In my free time I compose music. It's digital, as you could imagine, haha.

Table A.5.: Set of predefined turns to say goodbye in the introductory dialogue.

It's been a pleasure talking to you, we've had a good time, right?
Well, rest a bit and in a few minutes we will start the nutrition coaching session.
I already told you I'm in the testing phase, so take it easy, please.
Goodbye!
Bye bye!

A.3 | NUTRITION SCENARIO

Table A.6.: Set of predefined turns for the system presentation for the GROW dialogue about nutrition.

How are you?
Hello again.
What's up?
Everything ready for a new session? This time we will talk about your nutrition habits.

Table A.7.: Set of GSQs for the GROW dialogue about nutrition.

You like to eat?
Do you think you eat well?
Can you tell me your usual meal routine?
Tell me how you eat on any given day.
Tell me about your meals on any given day.
Would you like to change that?
How can I help you?
What do you want to achieve?
Could you specify your goal in a few words?
Specifically, what do you hope to obtain from this conversation?
Specifically, what do you hope to obtain from this conversation about your goal?
What would be the best you could extract from this conversation regarding your goal?
How close would that approach your goal?
When do you want to reach your goal?
How much are you willing to get involved to get it?

Table A.8.: Set of MQs for the GROW dialogue about nutrition.

Do you want to change that?
What would you contribute?
Why do you want to achieve the goal you mentioned?
What would be the ideal situation?
What could you gain by achieving it?
How would it benefit you to get it?
How would it affect your environment that you achieve it?
What do you think you're going to feel when you've reached it?
Imagine that you have already achieved it, how do you see yourself?
What resources have you needed to reach that vision?
What did you have to do to get there?
What qualities do others see in you?
What advantages do others believe you have to achieve it?
What benefits would you get if you could change your way of eating according to the goal?

Table A.9.: Set of RQs for the GROW dialogue about nutrition.

What is currently happening in relation to your goal?
How far is your current situation from your objective?
What are you doing now in relation to what you want to achieve?
Has anything that you have done worked for you?
What has not worked for you?
What will you do to avoid what makes it difficult for you to reach the goal?
What things can help you at this time?
How would you like to feel?

Table A.10.: Set of OQs for the GROW dialogue about nutrition.

What obstacles are you encountering?
At what times do these obstacles appear?
What resources do you already have, would you need to overcome them?
What other resources would you need to find?
What skills do you have that could be useful to you?
What have those skills taught you at other times?
What should you improve to get over them and get closer to your goal?

Table A.11.: Set of OGQs for the GROW dialogue about nutrition.

What actions can you take in relation to the stated objective?
What could you do to take another step towards your goal?
What could you do to get a little closer to achieving your goal?
If there were no obstacles, what more options would you come up with to reach your goal?
If an expert in this type of objectives faced this situation, what do you think would do?
Which of the actions you have thought is the one you like the most?
What characteristics does that action have that makes you like it?
What other actions that you have not considered still share those characteristics?
Which one would bring you closer to your goal?
What is the most realistic to implement at the current time?

Table A.12.: Set of PAQs for the GROW dialogue about nutrition.

What will you choose to do?
What are you going to do?
When are you doing it?
What will you do tomorrow? And next week?
How much time are you going to spend on it?
Where will you do it?
How does this plan approach your goal?
What problems could you find to carry out this plan?
How are you going to solve the problems that you might encounter?
Who should you report?

Table A.13.: Set of goodbye turns in the GROW dialogue about nutrition.

It’s been a pleasure talking to you, we’ve had a good time, right?
Goodbye!
Bye bye!

Appendix B

VAAQ Questionnaire

Table B.1.: Pragmatic qualities (in a 5-point likert scale).

I think that communicating with the agent is a decisive support in everyday activity.
I think that communicating with the agent is simple and easy.
I think that communicating with the agent is unmanageable.
I think that communicating with the agent is artificial.
I think that communicating with the agent is useless.
I think that communicating with the agent is qualifying.

Table B.2.: Hedonic qualities identity (in a 5-point likert scale).

I think the agent is friendly.
I think the agent is displeasing.
I think the agent is very human.
I think the agent is threatening.
I think the agent is reassuring.
I think the agent is untrustworthy.

Table B.3.: Hedonic qualities feelings (in a 5-point likert scale).

I think that communicating with the agent is extraordinary.
I think that communicating with the agent is boring.
I think that communicating with the agent is thrilling.
I think that communicating with the agent is trivial.
I think that communicating with the agent is stimulating.
I think that communicating with the agent is disconcerting.

Table B.4.: Attractiveness (in a 5-point likert scale).

I think that communicating with the agent will help to enhance my knowledge.
I think that communicating with the agent can be taken for granted.
I think that communicating with the agent is enjoyable.
I think that communicating with the agent is demotivating.
I think that communicating with the agent is engaging.
I think that communicating with the agent is stressful.

Table B.5.: Intelligibility (in a 5-point likert scale).

The agent express very appropriately feelings while communicating.
The agent’s way of expressing himself is cold and impersonal.
The agent can be easily understood.
It is hard to grasp what the agent says.
Speaking with the agent is natural and effortless.
The agents speaks in an atypical way.

Appendix C

Chatbot Usability and Hedonic Feelings Questionnaires

Table C.1.: Chatbot Usability Questionnaire.

Question code	Question
CUQ-1	The chatbot's personality was realistic and engaging.
CUQ-2	The chatbot seemed too robotic.
CUQ-3	The chatbot was welcoming during initial setup.
CUQ-4	The chatbot seemed very unfriendly.
CUQ-5	The chatbot explained its scope and purpose well.
CUQ-6	The chatbot gave no indication as to its purpose.
CUQ-7	The chatbot was easy to navigate.
CUQ-8	It would be easy to get confused when using the chatbot.
CUQ-9	The chatbot understood me well.
CUQ-10	The chatbot failed to recognise a lot of my inputs.
CUQ-11	Chatbot responses were useful, appropriate and informative.
CUQ-12	Chatbot responses were not relevant.
CUQ-13	The chatbot coped well with any errors or mistakes.
CUQ-14	The chatbot seemed unable to handle any errors.
CUQ-15	The chatbot was very easy to use.
CUQ-16	The chatbot was very complex.

Table C.2.: Hedonic Feelings Questionnaire.

Question code	Question
HFQ-1	I think the communication with the agent was extraordinary.
HFQ-2	I think the communication with the agent was boring.
HFQ-3	I think the communication with the agent was innovative.
HFQ-4	I think the communication with the agent was disappointing.
HFQ-5	I think the communication with the agent was thrilling.
HFQ-6	I think the communication with the agent was trivial.
HFQ-7	I think the communication with the agent was stimulant.
HFQ-8	I think the communication with the agent was depressing.
HFQ-9	I think the communication with the agent was reassuring.
HFQ-10	I think the communication with the agent was stressful.

Appendix D

Set of simplified EMPATHIC dialogue acts

Table D.1.: Abbreviations and descriptions of the simplified EMPATHIC dialogue acts.

Dialogue act	Description
Hello	Hello, salutation.
Ask name	Ask about the user's name and spelling.
Patience request	Patience request.
Self-intro	The system presents itself.
Know coaching?	Ask about the user's knowledge about coaching.
Echo	Repeat something said by the user, to transmit empathy and understanding.
Open Q	An open question about the user.
Yes/no Q	A yes/no question about the user.
Music	A question/statement about music.
Travel	A question/statement about travelling.
Other hobbies	A question/statement about a hobby that is not music nor travelling.
I understand	Explicitly tell the user that their message has been understood.
Clarify	Ask for a clarification.
Neg feedback	Disagree with the user or show a negative/non-positive opinion.
Pos feedback	Show a positive opinion.
Agreement	Agree with the user.
Topic	Open, close or choose a new topic.
Current situation	Questions about the current situation of the user regarding their goal.
GSQ-IS	Goal Setting Question - Ideal Situation. Ask the user which would be the ideal situation in connection with their goal.
GSQ-Obj	Goal Setting Question - Objective. Ask the user to define their goal.
MQ	Motivational Question.
ORQ	Obstacles/Resources Question. Questions to find out which obstacles that hinder the achievement of the goal, and the possible resources to overcome them.
PAQ	Plan Action Question. Question to define a plan that brings the user closer to their goal.
Thanking	Thank the user.
Farewell	Say goodbye.
Other	A system turn that was not classifiable as any of the aforementioned dialogue acts.

Bibliography

Abdulrahman, A., Richards, D., and Bilgin, A. A. (2022). Changing users' health behaviour intentions through an embodied conversational agent delivering explanations based on users' beliefs and goals. *Behaviour & Information Technology*, pages 1–19.

Aceta, C., Fernández, I., and Soroa, A. (2022). Kide4i: A generic semantics-based task-oriented dialogue system for human-machine interaction in industry 5.0. *Applied Sciences*, 12(3):1192.

Ackley, D. H., Hinton, G. E., and Sejnowski, T. J. (1985). A learning algorithm for boltzmann machines. *Cognitive science*, 9(1):147–169.

Adiwardana, D., Luong, M.-T., So, D. R., Hall, J., Fiedel, N., Thoppilan, R., Yang, Z., Kulshreshtha, A., Nemade, G., Lu, Y., et al. (2020). Towards a human-like open-domain chatbot. *Computing Research Repository*, arXiv:2001.09977.

Ajzen, I. et al. (1991). The theory of planned behavior. *Organizational behavior and human decision processes*, 50(2):179–211.

AlAmri, H., Cartillier, V., Das, A., Wang, J., Cherian, A., Essa, I., Batra, D., Marks, T. K., Hori, C., Anderson, P., Lee, S., and Parikh, D. (2019). Audio visual scene-aware dialog. In *CVPR*.

AlAmri, H., Cartillier, V., Lopes, R. G., Das, A., Wang, J., Essa, I., Batra, D., Parikh, D., Cherian, A., Marks, T. K., and Hori, C. (2018). Audio visual scene-aware dialog (AVSD) challenge at DSTC7. *CoRR*, abs/1806.00525.

Alphonse, A., Stewart, K., Brown, J., Perski, O., et al. (2022). Exploring users' experiences with a quick-response chatbot within a popular smoking cessation smartphone app: Semistructured interview study. *JMIR Formative Research*, 6(7):e36869.

Ankerst, M., Breunig, M. M., Kriegel, H.-P., and Sander, J. (1999). Optics: ordering points to identify the clustering structure. *ACM Sigmod record*, 28(2):49–60.

Arora, S., Liang, Y., and Ma, T. (2019). A simple but tough-to-beat baseline for sentence embeddings. In *5th International Conference on Learning Representations, ICLR 2017*.

Baevski, A., Zhou, Y., Mohamed, A., and Auli, M. (2020). wav2vec 2.0: A framework for self-supervised learning of speech representations. In *NeurIPS*.

Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Bani-Issa, W. (2011). Evaluation of the health-related quality of life of emirati people with diabetes: integration of sociodemographic and disease-related variables. *East Mediterr Health J*, 17(11):825–30.

Bao, S., He, H., Xu, J., Lu, H., Wang, F., Wu, H., Zhou, H., Wu, W., Niu, Z.-Y., and Wang, H. (2022). Plato-k: Internal and external knowledge enhanced dialogue generation. *arXiv preprint arXiv:2211.00910*.

Bayer, A. O., Stepanov, E. A., and Riccardi, G. (2017). Towards end-to-end spoken dialogue systems with turn embeddings. In *INTERSPEECH*, pages 2516–2520.

Belfin, R., Shobana, A., Manilal, M., Mathew, A. A., and Babu, B. (2019). A graph based chatbot for cancer patients. In *2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS)*, pages 717–721. IEEE.

Bellman, R. (1966). Dynamic programming. *Science*, 153(3731):34–37.

Bengio, S., Vinyals, O., Jaitly, N., and Shazeer, N. (2015). Scheduled sampling for sequence prediction with recurrent neural networks. In *Advances in Neural Information Processing Systems*, pages 1171–1179.

Bengio, Y., Léonard, N., and Courville, A. (2013). Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*.

Beun, R. J., Fitrianie, S., Griffioen-Both, F., Spruit, S., Horsch, C., Lancee, J., and Brinkman, W.-P. (2017). Talk and tools: the best of both worlds in mobile user interfaces for e-coaching. *Personal and ubiquitous computing*, 21(4):661–674.

Bhathiya, H. S. and Thayasivam, U. (2020). Meta learning for few-shot joint intent detection and slot-filling. In *Proceedings of the 2020 5th International Conference on Machine Learning Technologies*, ICMLT 2020, page 86–92, New York, NY, USA. Association for Computing Machinery.

Bickmore, T. W., Schulman, D., and Sidner, C. (2013). Automated interventions for multiple health behaviors using conversational agents. *Patient education and counseling*, 92(2):142–148.

Bohus, D. and Rudnicky, A. I. (2009). The RavenClaw dialog management framework: Architecture and systems. *Comput. Speech Lang.*, 23(3):332–361.

Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2016). Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*.

Bowman, S. R., Vilnis, L., Vinyals, O., Dai, A. M., Jozefowicz, R., and Bengio, S. (2015). Generating sentences from a continuous space. *arXiv preprint arXiv:1511.06349*.

Brinkschulte, L., Mariacher, N., Schlögl, S., Torres, M. I., Justo, R., Olaso, J. M., Esposito, A., Cordasco, G., Chollet, G., Glackin, C., Pickard, C., Petrovska-Delacretaz, D., Hmani, M. A., Mtibaa, A., Fernandez, A., Kyslitska, D., Fernandez-Ruanova, B., Tenorio-Laranga, J., Aksnes, M., Korsnes, M. S., Reiner, M., Lindner, F., Deroo, O., and Gordeeva, O. (2021). The empathic project: Building an expressive, advanced virtual coach to improve independent healthy-life-years of the elderly.

Brooke, J. (1996). Sus: a "quick and dirty'usability. *Usability evaluation in industry*, page 189.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. (2020). Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Budzianowski, P. and Vulic, I. (2019). Hello, it's gpt-2-how can i help you? towards the use of pretrained language models for task-oriented dialogue systems. *EMNLP-IJCNLP 2019*, page 15.

Budzianowski, P., Wen, T.-H., Tseng, B.-H., Casanueva, I., Ultes, S., Ramadan, O., and Gašić, M. (2018). Multiwoz–a large-scale multi-domain wizard-of-oz dataset for task-oriented dialogue modelling. *arXiv preprint arXiv:1810.00278*.

Bunt, H. (2009). The dit++ taxonomy for functional dialogue markup. In *Proc. of 8th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2009)*, Budapest, Hungary.

Burtsev, M., Seliverstov, A., Airapetyan, R., Arkhipov, M., Baymurzina, D., Bushkov, N., Gureenkova, O., Khakhulin, T., Kuratov, Y., Kuznetsov, D., et al. (2018). Deeppavlov: Open-source library for dialogue systems. In *ACL System Demonstrations*.

Busta, M., Neumann, L., and Matas, J. (2015). Fastext: Efficient unconstrained scene text detector. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1206–1214.

Callejas, Z., Griol, D., Benghazi, K., Noguera, M., Chollet, G., Torres, M. I., and Esposito, A. (2020). Measuring and fostering engagement with mental health e-coaches. In *Companion Publication of the 2020 International Conference on Multimodal Interaction*, ICMI '20 Companion, page 275–279, New York, NY, USA. Association for Computing Machinery.

Cao, Y., Bi, W., Fang, M., Shi, S., and Tao, D. (2022). A model-agnostic data manipulation method for persona-based dialogue generation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7984–8002.

Carlo, A. D., Ghomi, R. H., Renn, B. N., Strong, M. A., and Areán, P. A. (2020). Assessment of real-world use of behavioral health mobile applications by a novel stickiness metric. *JAMA network open*, 3(8):e2011978–e2011978.

Casanueva, I., Budzianowski, P., Su, P., Ultes, S., Rojas-Barahona, L. M., Tseng, B., and Gasic, M. (2018). Feudal reinforcement learning for dialogue management in large domains. In *NAACL-HLT*.

Casas, J., Mugellini, E., and Khaled, O. A. (2018). Food diary coaching chatbot. In *Proceedings of the 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers*, pages 1676–1680.

Chen, C., Qiu, L., Fu, Z., Liu, J., and Yan, R. (2019). Multilingual dialogue generation with shared-private memory. In *CCF International Conference on Natural Language Processing and Chinese Computing*, pages 42–54. Springer.

Chen, S., Wu, Y., Wang, C., Chen, Z., Chen, Z., Liu, S., Wu, J., Qian, Y., Wei, F., Li, J., et al. (2021). Unispeech-sat: Universal speech representation learning with speaker aware pre-training. *arXiv preprint arXiv:2110.05752*.

Chi, E. A., Chiam, C., Chang, T., Lim, S. K., Rastogi, C., Iyabor, A., He, Y., Sowrirajan, H., Narayan, A., Tang, J., et al. (2021). Neural, neural everywhere: Controlled generation meets scaffolded, structured dialogue. *Alexa Prize Proceedings*.

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.

Chowdhery, A., Narang, S., Devlin, J., Bosma, M., Mishra, G., Roberts, A., Barham, P., Chung, H. W., Sutton, C., Gehrmann, S., et al. (2022). Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*.

Conneau, A. and Lample, G. (2019). Cross-lingual language model pretraining. In *Advances in Neural Information Processing Systems*, pages 7059–7069.

Cuayáhuitl, H., Lee, D., Ryu, S., Cho, Y., Choi, S., Indurthi, S., Yu, S., Choi, H., Hwang, I., and Kim, J. (2019). Ensemble-based deep reinforcement learning for chatbots. *Neurocomputing*, 366:118–130.

Cuayáhuitl, H., Yu, S., Williamson, A., and Carse, J. (2017). Scaling up deep reinforcement learning for multi-domain dialogue systems. In *IJCNN*.

Dahlbäck, N., Jönsson, A., and Ahrenberg, L. (1993). Wizard of oz studies—why and how. *Knowledge-based systems*, 6(4):258–266.

de Velasco, M., Justo, R., and Torres, M. I. (2022). Automatic identification of emotional information in spanish tv debates and human–machine interactions. *Applied Sciences*, 12(4):1902.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee.

Denton, E. L., Chintala, S., Fergus, R., et al. (2015). Deep generative image models using a laplacian pyramid of adversarial networks. In *Advances in neural information processing systems*, pages 1486–1494.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186.

Dinan, E., Logacheva, V., Malykh, V., Miller, A., Shuster, K., Urbanek, J., Kiela, D., Szlam, A., Serban, I., Lowe, R., et al. (2020). The second conversational intelligence challenge (convai2). In *The NeurIPS'18 Competition*, pages 187–208. Springer.

Dinan, E., Roller, S., Shuster, K., Fan, A., Auli, M., and Weston, J. (2018). Wizard of wikipedia: Knowledge-powered conversational agents. In *International Conference on Learning Representations*.

Dubosson, F., Schaer, R., Savioz, R., and Schumacher, M. (2017). Going beyond the relapse peak on social network smoking cessation programmes: Chatbot opportunities. *Swiss medical informatics*, 33(00).

Ebrahimi, A., Mager, M., Oncevay, A., Chaudhary, V., Chiruzzo, L., Fan, A., Ortega, J., Ramos, R., Rios, A., Vladimir, I., et al. (2021). Americasnli: Evaluating zero-shot natural language understanding of pretrained multilingual models in truly low-resource languages. *arXiv preprint arXiv:2104.08726*.

Elkan, C. (2003). Using the triangle inequality to accelerate k-means. In *Proceedings of the 20th international conference on Machine Learning (ICML-03)*, pages 147–153.

Eric, M., Goel, R., Paul, S., Kumar, A., Sethi, A., Ku, P., Goyal, A. K., Agarwal, S., Gao, S., and Hakkani-Tur, D. (2019). Multiwoz 2.1: A consolidated multi-domain dialogue dataset with state corrections and state tracking baselines. *arXiv preprint arXiv:1907.01669*.

Esposito, A., Amorese, T., Cuciniello, M., Esposito, A. M., Troncone, A., Torres, M. I., Schlögl, S., and Cordasco, G. (2018). Seniors' acceptance of virtual humanoid agents. In *Italian forum of ambient assisted living*, pages 429–443. Springer.

Esposito, A., Amorese, T., Cuciniello, M., Riviello, M. T., Esposito, A. M., Troncone, A., Torres, M. I., Schlögl, S., and Cordasco, G. (2021). Elder user's attitude toward assistive virtual agents: the role of voice and gender. *Journal of Ambient Intelligence and Humanized Computing*, 12(4):4429–4436.

Ester, M., Kriegel, H.-P., Sander, J., and Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, KDD'96, page 226–231. AAAI Press.

Fadhil, A., Schiavo, G., and Wang, Y. (2019). Coachai: A conversational agent assisted health coaching platform. *arXiv preprint arXiv:1904.11961*.

Feng, F., Yang, Y., Cer, D., Arivazhagan, N., and Wang, W. (2022). Language-agnostic bert sentence embedding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, pages 878–891.

Ficler, J. and Goldberg, Y. (2017). Controlling linguistic style aspects in neural language generation. *Computing Research Repository*, arXiv:1707.02633.

Firdaus, M., Ekbal, A., and Bhattacharyya, P. (2022). Polise: Reinforcing politeness using user sentiment for customer care response generation. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 6165–6175.

Firdaus, M., Jain, U., Ekbal, A., and Bhattacharyya, P. (2021). Seprg: Sentiment aware emotion controlled personalized response generation. In *Proceedings of the 14th International Conference on Natural Language Generation*, pages 353–363.

Fuad, A. and Al-Yahya, M. (2022). Cross-lingual transfer learning for arabic task-oriented dialogue systems using multilingual transformer model mt5. *Mathematics*, 10(5):746.

Ganesan, K., Bamdev, P., Venugopal, A., Tushar, A., et al. (2021). N-best asr transformer: Enhancing slu performance using multiple asr hypotheses. In *59th Annual Meeting of the ACL and the 11th IJCNLP*, pages 93–98. ACL.

Ganhotra, J., Thomas, S., Kuo, H.-K. J., Joshi, S., Saon, G., Tüske, Z., and Kingsbury, B. (2021). Integrating dialog history into end-to-end spoken language understanding systems. *arXiv preprint arXiv:2108.08405*.

Gao, X., Zhang, Y., Lee, S., Galley, M., Brockett, C., Gao, J., and Dolan, B. (2019). Structuring latent spaces for stylized response generation. 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)*, pages 1814–1823.

Gaydhani, A., Finzel, R., Dufresne, S., Gini, M., and Pakhomov, S. (2020). Conversational agent for daily living assessment coaching. In *CEUR Workshop Proceedings*, volume 2760, pages 8–13. CEUR-WS.

Ghandeharioun, A., Shen, J. H., Jaques, N., Ferguson, C., Jones, N., Lapedriza, A., and Picard, R. (2019). Approximating interactive human evaluation with self-play for open-domain dialog systems. *Advances in Neural Information Processing Systems*, 32.

Ghazvininejad, M., Brockett, C., Chang, M.-W., Dolan, B., Gao, J., Yih, W.-t., and Galley, M. (2018). A knowledge-grounded neural conversation model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Ghigi, F., Eskenazi, M., Torres, M. I., and Lee, S. (2014). Incremental dialog processing in a task-oriented dialog. In *Fifteenth Annual Conference of the International Speech Communication Association, Interspeech.*

Gilmartin, E., Saam, C., Vogel, C., Campbell, N., and Wade, V. (2018). Just talking - modelling casual conversation. In *Proceedings of the 19th Annual SIGdial Meeting on Discourse and Dialogue*, pages 51–59, Melbourne, Australia. Association for Computational Linguistics.

Goel, R., Paul, S., and Hakkani-Tür, D. (2019). Hyst: A hybrid approach for flexible and accurate dialogue state tracking. *Proc. Interspeech 2019*, pages 1458–1462.

Golovanov, S., Kurbanov, R., Nikolenko, S., Truskovskyi, K., Tselousov, A., and Wolf, T. (2019). Large-scale transfer learning for natural language generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6053–6058.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680.

Greco, C., Buono, C., Buch-Cardona, P., Cordasco, G., Escalera, S., Esposito, A., Fernandez, A., Kyslitska, D., Kornes, M. S., Palmero, C., et al. (2021). Emotional features of interactions with empathic agents. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2168–2176.

Griol, D. and Callejas, Z. (2019). Discovering dialog rules by means of an evolutionary approach. In *INTERSPEECH*, pages 1473–1477.

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., and Courville, A. C. (2017). Improved training of wasserstein gans. *Advances in neural information processing systems*, 30.

Gunasekara, C., Kim, S., D'Haro, L. F., Rastogi, A., Chen, Y.-N., Eric, M., Hedayatnia, B., Gopalakrishnan, K., Liu, Y., Huang, C.-W., et al. (2020). Overview of the ninth dialog system technology challenge: Dstc9. *arXiv preprint arXiv:2011.06486*.

Haghani, P., Narayanan, A., Bacchiani, M., Chuang, G., Gaur, N., Moreno, P., Prabhavalkar, R., Qu, Z., and Waters, A. (2018). From audio to semantics: Approaches to end-to-end spoken language understanding. In *2018 IEEE Spoken Language Technology Workshop (SLT)*, pages 720–726. IEEE.

Hakkani-Tür, D., Béchet, F., Riccardi, G., and Tur, G. (2006). Beyond asr 1-best: Using word confusion networks in spoken language understanding. *Computer Speech & Language*, 20(4):495–514.

Hakkani-Tur, D. and Riccardi, G. (2003). A general algorithm for word graph matrix decomposition. In *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03).*, volume 1, pages I–I. IEEE.

Ham, D., Lee, J.-G., Jang, Y., and Kim, K.-E. (2020). End-to-end neural pipeline for goal-oriented dialogue systems using GPT-2. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 583–592, Online. Association for Computational Linguistics.

Hassenzahl, M., Burmester, M., and Koller, F. (2003). Attrakdiff: Ein fragebogen zur messung wahrgenommener hedonischer und pragmatischer qualität. In *Mensch & computer 2003*, pages 187–196. Springer.

He, T. and Glass, J. (2020). Negative training for neural dialogue response generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2044–2058.

He, Y. and Young, S. (2003). A data-driven spoken language understanding system. In *2003 IEEE Workshop on Automatic Speech Recognition and Understanding (IEEE Cat. No. 03EX721)*, pages 583–588. IEEE.

Hedayatnia, B., Kim, S., Liu, Y., Gopalakrishnan, K., Eric, M., and Hakkani-Tur, D. (2020). Policy-driven neural response generation for knowledge-grounded dialogue systems. *arXiv preprint arXiv:2005.12529*.

Henderson, M., Thomson, B., and Williams, J. D. (2014a). The second dialog state tracking challenge. In *SIGDIAL*.

Henderson, M., Thomson, B., and Williams, J. D. (2014b). The third dialog state tracking challenge. In *SLT*.

Hmani, M., Mtibaa, A., and Petrovska-Delacrétaz, D. (2021). *Joining forces of voice and facial biometrics: a case study in the scope of NIST SRE19*, pages 187–217. Voice Biometrics: Technology, trust and security.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

Holmes, S., Moorhead, A., Bond, R., Zheng, H., Coates, V., and McTear, M. (2019). Usability testing of a healthcare chatbot: Can we use conventional methods to assess conversational user interfaces? In *Proceedings of the 31st European Conference on Cognitive Ergonomics*, pages 207–214.

Holtzman, A., Buys, J., Du, L., Forbes, M., and Choi, Y. (2019). The curious case of neural text degeneration. In *Proceedings of the International Conference on Learning Representations*.

Hori, T., Wang, W., Koji, Y., Hori, C., Harsham, B., and Hershey, J. R. (2019). Adversarial training and decoding strategies for end-to-end neural conversation models. *Computer Speech & Language*, 54:122–139.

Hosseini-Asl, E., McCann, B., Wu, C.-S., Yavuz, S., and Socher, R. (2020). A simple language model for task-oriented dialogue. *Advances in Neural Information Processing Systems*, 33:20179–20191.

Hsu, W.-N., Bolte, B., Tsai, Y.-H. H., Lakhotia, K., Salakhutdinov, R., and Mohamed, A. (2021). Hubert: Self-supervised speech representation learning by masked prediction of hidden units. *arXiv preprint arXiv:2106.07447*.

Hu, T., Long, C., and Xiao, C. (2021). A novel visual representation on text using diverse conditional gan for visual recognition. *IEEE Transactions on Image Processing*, 30:3499–3512.

Huang, C.-Y., Yang, M.-C., and Huang, C.-Y. (2021). An empirical study on factors influencing consumer adoption intention of an ai-powered chatbot for health and weight management. *International Journal of Performability Engineering*, 17(5).

Huang, C.-Y., Yang, M.-C., Huang, C.-Y., Chen, Y.-J., Wu, M.-L., and Chen, K.-W. (2018). A chatbot-supported smart wireless interactive healthcare system for weight control and health promotion. In *2018 IEEE international conference on industrial engineering and engineering management (IEEM)*, pages 1791–1795. IEEE.

Huang, J., Tao, J., Liu, B., Lian, Z., and Niu, M. (2020). Multimodal transformer fusion for continuous emotion recognition. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3507–3511.

Huang, Q., Zhang, Y., Ko, T., Liu, X., Wu, B., Wang, W., and Tang, L. (2022). Personalized dialogue generation with persona-adaptive attention. *arXiv preprint arXiv:2210.15088*.

Inkster, B., Sarda, S., and Subramanian, V. (2018). An empathy-driven, conversational artificial intelligence agent (wysa) for digital mental well-being: real-world data evaluation mixed-methods study. *JMIR mHealth and uHealth*, 6(11):e12106.

Jang, E., Gu, S., and Poole, B. (2016). Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*.

Jiang, S. and de Rijke, M. (2018). Why are sequence-to-sequence models so dull. *EMNLP 2018*, page 81.

Jiang, S., Ren, P., Monz, C., and de Rijke, M. (2019). Improving neural response diversity with frequency-aware cross-entropy loss. In *The World Wide Web Conference*, pages 2879–2885.

Jiang, X., Liang, Y., Chen, W., and Duan, N. (2022). Xlm-k: Improving cross-lingual language model pre-training with multilingual knowledge. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 10840–10848.

Justo, R., Alcaide, J., and Torres, M. (2016). Crowdzientzia: Crowdsourcing for research and development. *Proceedings of IberSpeech*, pages 403–410.

Justo, R., Ben Letaifa, L., Palmero, C., Gonzalez-Fraile, E., Torp Johansen, Annaand Vázquez, A., Cordasco, G., Schlögl, S., Fernández-Ruanova, B., Silva, M., Escalera, S., deVelasco, M., Tenorio-Laranga, J., Esposito, A., Korsnes, M., and Torres, M. I. (2020). Analysis of the interaction between elderly people and a simulated virtual coach. *Journal of Ambient Intelligence and Humanized Computing*, 11(12).

Justo, R., Pérez, A., and Torres, M. I. (2011). Impact of the approaches involved on word-graph derivation from the asr system. In *Iberian Conference on Pattern Recognition and Image Analysis*, pages 668–675. Springer.

Kadariya, D., Venkataramanan, R., Yip, H. Y., Kalra, M., Thirunarayanan, K., and Sheth, A. (2019). kbot: Knowledge-enabled personalized chatbot for asthma self-management. In *2019 IEEE International Conference on Smart Computing (SMARTCOMP)*, pages 138–143. IEEE.

Kannan, A. and Vinyals, O. (2017). Adversarial evaluation of dialogue models. *arXiv preprint arXiv:1701.08198*.

Karras, T., Aila, T., Laine, S., and Lehtinen, J. (2018). Progressive growing of gans for improved quality, stability, and variation. In *International Conference on Learning Representations (ICLR)*.

Keizer, S., Braunschweiler, N., Stoyanchev, S., and Doddipatla, R. (2021). Dialogue strategy adaptation to new action sets using multi-dimensional modelling. In *Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE.

Kettle, L. and Lee, Y.-C. (2021). “welcome to your daily wellness check”: The proposed evaluation of a sms-based conversational agent for managing health and wellbeing. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, volume 65, pages 791–795. SAGE Publications Sage CA: Los Angeles, CA.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kocielnik, R., Xiao, L., Avrahami, D., and Hsieh, G. (2018). Reflection companion: A conversational system for engaging users in reflection on physical activity. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 2(2):1–26.

Komeili, M., Shuster, K., and Weston, J. (2022). Internet-augmented dialogue generation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8460–8478.

Konda, V. R. and Tsitsiklis, J. N. (1999). Actor-citic agorithms. In *NIPS*.

Kramer, M. A. (1991). Nonlinear principal component analysis using autoassociative neural networks. *AIChE journal*, 37(2):233–243.

Kreyssig, F., Casanueva, I., Budzianowski, P., and Gasic, M. (2018). Neural user simulation for corpus-based policy optimisation of spoken dialogue systems. In *Proceedings of the 19th Annual SIGdial Meeting on Discourse and Dialogue*, pages 60–69.

Krippendorff, K. (2011). Computing krippendorff's alpha-reliability. *Annenberg School for Communication Departmental Papers: Philadelphia*.

Krippendorff, K. (2018). *Content analysis: An introduction to its methodology*. Sage publications.

Kulikov, I., Miller, A. H., Cho, K., and Weston, J. (2018). Importance of a search strategy in neural dialogue modelling. *Computing Research Repository*, arXiv:1811.00907.

Kusner, M. J. and Hernández-Lobato, J. M. (2016). Gans for sequences of discrete elements with the gumbel-softmax distribution. *arXiv preprint arXiv:1611.04051*.

Laranjo, L., Dunn, A. G., Tong, H. L., Kocaballi, A. B., Chen, J., Bashir, R., Surian, D., Gallego, B., Magrabi, F., Lau, A. Y. S., and Coiera, E. (2018). Conversational agents in healthcare: a systematic review. *Journal of the American Medical Informatics Association*, 5(9):1248–1258.

Latif, S., Cuayáhuitl, H., Pervez, F., Shamshad, F., Ali, H. S., and Cambria, E. (2021). A survey on deep reinforcement learning for audio-based applications. *CoRR*, abs/2101.00240.

Le, H., Sahoo, D., Chen, N. F., and Hoi, S. C. H. (2019). Multimodal transformer networks for end-to-end video-grounded dialogue systems. In *ACL*.

Le, L., Patterson, A., and White, M. (2018). Supervised autoencoders: Improving generalization performance with unsupervised regularizers. *Advances in Neural Information Processing Systems*, 31:107–117.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *nature*, 521(7553):436–444.

Letaifa, L. B. and Torres, M. I. (2021). Perceptual borderline for balancing multi-class spontaneous emotional data. *IEEE Access*, 9:55939–55954.

Li, B., Torr, P. H., and Lukasiewicz, T. (2022a). Clustering generative adversarial networks for story visualization. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 769–778.

Li, J., Galley, M., Brockett, C., Gao, J., and Dolan, W. B. (2016a). A diversity-promoting objective function for neural conversation models. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 110–119.

Li, J., Galley, M., Brockett, C., Spithourakis, G., Gao, J., and Dolan, B. (2016b). A persona-based neural conversation model. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 994–1003.

Li, J., Monroe, W., Shi, T., Jean, S., Ritter, A., and Jurafsky, D. (2017). Adversarial learning for neural dialogue generation. *arXiv preprint arXiv:1701.06547*.

Li, M., Roller, S., Kulikov, I., Welleck, S., Boureau, Y.-L., Cho, K., and Weston, J. (2020). Don't say that! making inconsistent dialogue unlikely with unlikelihood training. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4715–4728.

Li, M., Weston, J., and Roller, S. (2019). Acute-eval: Improved dialogue evaluation with optimized questions and multi-turn comparisons. *arXiv preprint arXiv:1909.03087*.

Li, Y., Feng, S., Sun, B., and Li, K. (2022b). Diversifying neural dialogue generation via negative distillation. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 407–418.

Li, Z., Kiseleva, J., and de Rijke, M. (2021). Improving response quality with backward reasoning in open-domain dialogue systems. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1940–1944.

Li, Z., Wu, J., Miao, J., and Yu, X. (2022c). Improve the response diversity of multi-turn dialogue system by combining knowledge. *IAENG International Journal of Computer Science*, 49(3).

Lin, K., Li, D., He, X., Zhang, Z., and Sun, M.-T. (2017). Adversarial ranking for language generation. *Advances in neural information processing systems*, 30.

Lin, Q., Kundu, S., and Ng, H. T. (2020a). A co-attentive cross-lingual neural model for dialogue breakdown detection. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4201–4210.

Lin, Z., Liu, Z., Winata, G. I., Cahyawijaya, S., Madotto, A., Bang, Y., Ishii, E., and Fung, P. (2020b). Xpersona: Evaluating multilingual personalized chatbot. *Computing Research Repository*, arXiv:2003.07568.

Lison, P. and Tiedemann, J. (2016). Opensubtitles2016: Extracting large parallel corpora from movie and tv subtitles. *European Language Resources Association.*

Lison, P., Tiedemann, J., Kouylekov, M., et al. (2019). Open subtitles 2018: Statistical rescoring of sentence alignments in large, noisy parallel corpora. In *LREC 2018, Eleventh International Conference on Language Resources and Evaluation.* European Language Resources Association (ELRA).

Litman, D., Strik, H., and Lim, G. S. (2018). Speech technologies and the assessment of second language speaking: Approaches, challenges, and opportunities. *Language Assessment Quarterly*, 15(3):294–309.

Liu, B. and Lane, I. (2018). End-to-end learning of task-oriented dialogs. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Student Research Workshop*, pages 67–73.

Liu, C.-W., Lowe, R., Serban, I., Noseworthy, M., Charlin, L., and Pineau, J. (2016). How not to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. In *EMNLP*.

Liu, Y., Han, K., Tan, Z., and Lei, Y. (2017). Using context information for dialog act classification in dnn framework. In *Proceedings of the 2017 conference on empirical methods in natural language processing*, pages 2170–2178.

López Zorrilla, A., De Velasco Vázquez, M., and Justo Blanco, R. (2019). Euskaraz hitz egiten ikasten duten makina autodidaktak [basque speaking self-learning machines]. In *Proceedings of IkerGazte 2019.* Udako Euskal Unibertsitatea.

López Zorrilla, A., deVelasco Vázquez, M., and Justo, R. (2020). Euskarazko elkarrizketa sistema automatikoa sare neuronalen bidez [a neural dialogue system in basque]. *Ekaia. EHUko Zientzia eta Teknologia aldizkaria.*

López Zorrilla, A., deVelasco Vázquez, M., and Torres, M. I. (2021a). A differentiable generative adversarial network for open domain dialogue. In Marchi, E., Siniscalchi, S. M., Cumani, S., Salerno, V. M., and Li, H., editors, *Increasing Naturalness and Flexibility in Spoken Dialogue Interaction: 10th International Workshop on Spoken Dialogue Systems*, pages 277–289, Singapore. Springer Singapore.

López Zorrilla, A., Dugan, N., Torres, M. I., Glackin, C., Chollet, G., and Cannings, N. (2016). Some ASR experiments using deep neural networks on Spanish databases. *Advances in Speech and Language Technologies for Iberian Languages. IberSPEECH.*

López Zorrilla, A. and Torres, M. I. (2022). A multilingual neural coaching model with enhanced long-term dialogue structure. *ACM Transactions on Interactive Intelligent Systems*, 12(2).

López Zorrilla, A., Torres, M. I., and Cuayáhuitl, H. (2021b). Audio embeddings help to learn better dialogue policies. In *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 962–968. IEEE.

López Zorrilla, A., Torres, M. I., and Cuayáhuitl, H. (2022). Audio embedding-aware dialogue policy learning. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*.

López Zorrilla, A., Vázquez, A., and Torres, M. I. (2023). Batch-level GANs to promote dialogue response variety. In *13th International Workshop on Spoken Dialogue Systems Technology (Under revision).*

López Zorrilla, A., Velasco Vázquez, M. d., Irastorza, J., Olaso Fernández, J. M., Justo Blanco, R., and Torres Barañano, M. I. (2018). Empathic: Empathic, expressive, advanced virtual coach to improve independent healthy-life-years of the elderly. *Sociedad Española para el Procesamiento del Lenguaje Natural.*

Loshchilov, I. and Hutter, F. (2017). Decoupled weight decay regularization. *Computing Research Repository*, arXiv:1711.05101.

Lu, J., Kannan, A., Yang, J., Parikh, D., and Batra, D. (2017). Best of both worlds: Transferring knowledge from discriminative learning to a generative visual dialog model. In *Advances in Neural Information Processing Systems*, pages 314–324.

Lucas, T., Tallec, C., Ollivier, Y., and Verbeek, J. (2018). Mixed batches and symmetric discriminators for gan training. In *International Conference on Machine Learning*, pages 2844–2853. PMLR.

Lugosch, L., Meyer, B. H., Nowrouzezahrai, D., and Ravanelli, M. (2020). Using speech synthesis to train end-to-end spoken language understanding models. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8499–8503. IEEE.

Luo, T.-C. and Chien, J.-T. (2021). Variational dialogue generation with normalizing flows. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7778–7782. IEEE.

Luo, X., Qin, M. S., Fang, Z., and Qu, Z. (2021). Artificial intelligence coaches for sales agents: Caveats and solutions. *Journal of Marketing*, 85(2):14–32.

Luong, M.-T., Pham, H., and Manning, C. D. (2015). Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025.*

MacQueen, J. et al. (1967). Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, pages 281–297. Oakland, CA, USA.

Maddison, C. J., Mnih, A., and Teh, Y. W. (2016). The concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712.*

Maher, C. A., Davis, C. R., Curtis, R. G., Short, C. E., and Murphy, K. J. (2020). A physical activity and diet program delivered by artificially intelligent virtual health coach: Proof-of-concept study. *JMIR mHealth and uHealth*, 8(7):e17558.

Mehri, S. and Eskenazi, M. (2020). Unsupervised evaluation of interactive dialog with dialogpt. In *Proceedings of the 21th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 225–235.

Mishra, K., Firdaus, M., and Ekbal, A. (2022). Please be polite: Towards building a politeness adaptive dialogue system for goal-oriented conversations. *Neurocomputing*, 494:242–254.

Mohan, S., Venkatakrishnan, A., and Hartzler, A. L. (2020). Designing an ai health coach and studying its utility in promoting regular aerobic exercise. *ACM Trans. Interact. Intell. Syst.*, 10(2).

Montenegro, C., López Zorrilla, A., Mikel Olaso, J., Santana, R., Justo, R., Lozano, J. A., and Torres, M. I. (2019a). A dialogue-act taxonomy for a virtual coach designed to improve the life of elderly. *Multimodal Technologies and Interaction*, 3(3):52.

Montenegro, C., Santana, R., and Lozano, J. A. (2019b). Data generation approaches for topic classification in multilingual spoken dialog systems. In *Proceedings of the 12th ACM International Conference on PErvasive Technologies Related to Assistive Environments*, pages 211–217.

Montenegro, C., Santana, R., and Lozano, J. A. (2021). Analysis of the sensitivity of the end-of-turn detection task to errors generated by the automatic speech recognition process. *Eng. Appl. Artif. Intell.*, 100:104189.

Mosig, J. E., Mehri, S., and Kober, T. (2020). Star: A schema-guided dialog dataset for transfer learning. *arXiv preprint arXiv:2010.11853*.

Müller, M., Choudhary, S., Chung, C., Mouchtaris, A., and Kunzmann, S. (2021). In pursuit of babel-multilingual end-to-end spoken language understanding. In *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 1042–1049. IEEE.

Nabeel, M., Riaz, A., and Wang, Z. (2019). Cas-gans: An approach of dialogue policy learning based on gan and rl techniques. *International Journal of Advanced Computer Science and Applications*, 10(7).

Nasri, M., Hmani, M. A., Mtibaa, A., Petrovska-Delacretaz, D., Slima, M. B., and Hamida, A. B. (2020). Face emotion recognition from static image based on convolution neural networks. In *2020 5th International Conference on Advanced Technologies for Signal and Image Processing (ATSIP)*, pages 1–6. IEEE.

Nedelchev, R., Lehmann, J., and Usbeck, R. (2020). Language model transformers as evaluators for open-domain dialogues. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6797–6808.

Nguyen, D. T., Sharma, S., Schulz, H., and Asri, L. E. (2018). From film to video: Multi-turn question answering with multi-modal context. *CoRR*, abs/1812.07023.

Ni, J., Abrego, G. H., Constant, N., Ma, J., Hall, K., Cer, D., and Yang, Y. (2022). Sentence-t5: Scalable sentence encoders from pre-trained text-to-text models. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1864–1874.

Niu, T. and Bansal, M. (2018). Polite dialogue generation without parallel data. *Transactions of the Association for Computational Linguistics*, 6:373–389.

Nowak, M. A. (2000). Evolutionary biology of language. *Philosophical Transactions of the Royal Society of London. Series B: Biological Sciences*, 355(1403):1615–1622.

Olafsson, S., O'Leary, T. K., and Bickmore, T. W. (2020). Motivating health behavior change with humorous virtual agents. In *Proceedings of the 20th ACM International Conference on Intelligent Virtual Agents*, IVA '20, New York, NY, USA. Association for Computing Machinery.

Olaso, J. M., Milhorat, P., Himmelsbach, J., Boudy, J., Chollet, G., Schlögl, S., and Torres, M. I. (2016). *A Multi-lingual Evaluation of the vAssist Spoken Dialog System. Comparing Disco and RavenClaw*, pages 221–232. Springer Singapore.

Olaso, J. M. and Torres, M. I. (2017). User experience evaluation of a conversational bus information system in spanish. In *8th IEEE International Conference on Cognitive InfoCommunications*, Debrecen, Hungary, September. IEEE.

Olaso, J. M., Vázquez, A., Ben Letaifa, L., de Velasco, M., Mtibaa, A., Hmani, M. A., Petrovska-Delacrétaz, D., Chollet, G., Montenegro, C., López Zorrilla, A., Justo, R., Santana, R., Tenorio-Laranga, J., González-Fraile, E., Fernández-Ruanova, B. n., Cordasco, G., Esposito, A., Gjellesvik, K. B., Johansen, A. T., Kornes, M. S., Pickard, C., Glackin, C., Cahalane, G., Buch, P., Palmero, C., Escalera, S., Gordeeva, O., Deroo, O., Fernández, A., Kyslitska, D., Lozano, J. A., Torres, M. I., and Schlögl, S. (2021). The EMPATHIC virtual coach: a demo. In *Proceedings of the 2021 International Conference on Multimodal Interaction*, pages 848–851.

Ortega, D., Li, C.-Y., Vallejo, G., Denisov, P., and Vu, N. T. (2019). Context-aware neural-based dialog act classification on automatically generated transcriptions. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7265–7269. IEEE.

Ortiz Suárez, P. J., Romary, L., and Sagot, B. (2020). A monolingual approach to contextualized word embeddings for mid-resource languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1703–1714, Online. Association for Computational Linguistics.

Palmero, C., Selva, J., Bagheri, M., and Escalera, S. (2018). Recurrent cnn for 3d gaze estimation using appearance and shape cues. arxiv 2018. *arXiv preprint arXiv:1805.03064*.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Peng, B., Li, C., Li, J., Shayandeh, S., Liden, L., and Gao, J. (2020). Soloist: Few-shot task-oriented dialog with a single pre-trained auto-regressive model. *arXiv preprint arXiv:2005.05298*.

Pepino, L., Riera, P., and Ferrer, L. (2021). Emotion recognition from speech using wav2vec 2.0 embeddings. *arXiv preprint arXiv:2104.03502*.

Pittermann, J. and Pittermann, A. (2006). Integrating emotion recognition into an adaptive spoken language dialogue system. In *2006 2nd IET International Conference on Intelligent Environments-IE 06*, volume 1, pages 197–202. IET.

Radford, A., Narasimhan, K., Salimans, T., and Sutskever, I. (2018). Improving language understanding by generative pre-training.

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

Ralston, K., Chen, Y., Isah, H., and Zulkernine, F. (2019). A voice interactive multilingual student support system using ibm watson. In *2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA)*, pages 1924–1929.

Rastogi, A., Zang, X., Sunkara, S., Gupta, R., and Khaitan, P. (2020). Towards scalable multi-domain conversational agents: The schema-guided dialogue dataset. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8689–8696.

Raux, A., Langner, B., Bohus, D., Black, A. W., and Eskenazi, M. (2005). Let's go public! taking a spoken dialog system to the real world. In *in Proc. of Interspeech 2005*. Citeseer.

Reimers, N. and Gurevych, I. (2020). Making monolingual sentence embeddings multilingual using knowledge distillation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Richards, D. and Caldwell, P. (2017). Improving health outcomes sooner rather than later via an interactive website and virtual specialist. *IEEE journal of biomedical and health informatics*, 22(5):1699–1706.

Riek, L. D. (2012). Wizard of oz studies in hri: a systematic review and new reporting guidelines. *Journal of Human-Robot Interaction*, 1(1):119–136.

Rodríguez-Cantelar, M., D'Haro, L. F., and Matía, F. (2020). Automatic evaluation of non-task oriented dialog systems by using sentence embeddings projections and their dynamics. In *Conversational Dialogue Systems for the Next Decade*, pages 71–84. Springer.

Roller, S., Dinan, E., Goyal, N., Ju, D., Williamson, M., Liu, Y., Xu, J., Ott, M., Shuster, K., Smith, E. M., et al. (2020). Recipes for building an open-domain chatbot. *Computing Research Repository*, arXiv:2004.13637.

Saha, T., Chopra, S., Saha, S., Bhattacharyya, P., and Kumar, P. (2021). A large-scale dataset for motivational dialogue system: An application of natural language generation to mental health. In *2021 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE.

Saleh, A., Jaques, N., Ghandeharioun, A., Shen, J. H., and Picard, R. (2019). Hierarchical reinforcement learning for open-domain dialog. *Computing Research Repository*, arXiv:1909.07547.

Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., and Chen, X. (2016). Improved techniques for training gans. In *Advances in Neural Information Processing Systems*, pages 2234–2242.

Sano, M., Ouchi, H., and Tsuboi, Y. (2018). Addressee and response selection for multilingual conversation. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3631–3644.

Sauer, A., Chitta, K., Müller, J., and Geiger, A. (2021). Projected gans converge faster. *Advances in Neural Information Processing Systems*, 34:17480–17492.

Saxena, A., Prasad, M., Gupta, A., Bharill, N., Patel, O. P., Tiwari, A., Er, M. J., Ding, W., and Lin, C.-T. (2017). A review of clustering techniques and developments. *Neurocomputing*, 267:664–681.

Sayas, S. (2018a). Dialogues on leisure and free time. In *Technical Report DP3. Sayasalud and Empathic project.*

Sayas, S. (2018b). Dialogues on nutrition. In *Technical Report DP1. Sayasalud and Empathic project.*

Sayas, S. (2018c). Dialogues on physical exercise. In *Technical Report DP2. Sayasalud and Empathic project*.

Schlögl, S., Doherty, G., Karamanis, N., and Luz, S. (2010). Webwoz: a wizard of oz prototyping framework. In *Proceedings of the 2nd ACM SIGCHI symposium on Engineering interactive computing systems*, pages 109–114.

Schmidt, M., Stier, D., Werner, S., and Minker, W. (2019). Exploration and assessment of proactive use cases for an in-car voice assistant. *Studientexte zur Sprachkommunikation: Elektronische Sprachsignalverarbeitung 2019*, pages 148–155.

Schröder, H., Fitó, M., Estruch, R., Martínez-González, M. A., Corella, D., Salas-Salvadó, J., Lamuela-Raventós, R., Ros, E., Salaverría, I., Fiol, M., Lapetra, J., Vinyoles, E., Gómez-Gracia, E., Lahoz, C., Serra-Majem, L., Pintó, X., Ruiz-Gutierrez, V., and Covas, M. (2011). A Short Screener Is Valid for Assessing Mediterranean Diet Adherence among Older Spanish Men and Women. *The Journal of Nutrition*, 141(6):1140–1145.

Sennrich, R., Haddow, B., and Birch, A. (2016). Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725.

Seo, S., Kwak, D., and Lee, B. (2021). Integration of pre-trained networks with continuous token interface for end-to-end spoken language understanding. *arXiv preprint arXiv:2104.07253*.

Serban, I. V., Sordoni, A., Bengio, Y., Courville, A. C., and Pineau, J. (2016). Building end-to-end dialogue systems using generative hierarchical neural network models. In *AAAI*, volume 16, pages 3776–3784.

Serras, M. (2021). *Contributions to attributed probabilistic finite state bi-automata for dialogue management*. PhD thesis, University of the Basque Country UPV/EHU.

Serras, M., Torres, M. I., and del Pozo, A. (2019a). Goal-conditioned user modeling for dialogue systems using stochastic bi-automata. In *ICPRAM*.

Serras, M., Torres, M. I., and Del Pozo, A. (2019b). User-aware dialogue management policies over attributed bi-automata. *Pattern Analysis and Applications*, 22(4):1319–1330.

Sheikh, J. I. and Yesavage, J. A. (1986). Geriatric Depression Scale (GDS): recent evidence and development of a shorter version. *Clinical Gerontologist: The Journal of Aging and Mental Health*.

Sherman, L. E., Michikyan, M., and Greenfield, P. (2013). The effects of text, audio, video, and in-person communication on bonding between friends. *Cyberpsychology: Journal of Psychosocial Research on Cyberspace*, 7(2).

Shetty, R., Rohrbach, M., Hendricks, L. A., Fritz, M., and Schiele, B. (2017). Speaking the same language: Matching machine to human captions by adversarial training. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

Shi, W. and Yu, Z. (2018). Sentiment adaptive end-to-end dialog systems. In *ACL*.

Siglen, E., Vetti, H. H., Lunde, A. B. F., Hatlebrekke, T. A., Strømsvik, N., Hamang, A., Hovland, S. T., Rettberg, J. W., Steen, V. M., and Bjorvatn, C. (2022). Ask rosa–the making of a digital genetic conversation tool, a chatbot, about hereditary breast and ovarian cancer. *Patient Education and Counseling*, 105(6):1488–1494.

Sinha, C., Cheng, A. L., and Kadaba, M. (2022). Adherence and engagement with a cognitive behavioral therapy based conversational agent (wysa) in adults with chronic pain: Survival analysis. *JMIR Formative Research*.

Soltau, H., Shafran, I., Wang, M., Rastogi, A., Zhao, J., Jia, Y., Han, W., Cao, Y., and Miranda, A. (2022). Speech aware dialog system technology challenge (dstc11). *arXiv preprint arXiv:2212.08704*.

Sordoni, A., Galley, M., Auli, M., Brockett, C., Ji, Y., Mitchell, M., Nie, J.-Y., Gao, J., and Dolan, B. (2015). A neural network approach to context-sensitive generation of conversationaesponses. *arXiv preprint arXiv:1506.06714*.

Steinbach, M., Ertöz, L., and Kumar, V. (2004). The challenges of clustering high dimensional data. In *New directions in statistical physics*, pages 273–309. Springer.

Su, H., Shen, X., Hu, P., Li, W., and Chen, Y. (2018). Dialogue generation with gan. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Sun, Y., Hu, Y., Xing, L., Yu, J., and Xie, Y. (2020). History-adaption knowledge incorporation mechanism for multi-turn dialogue system. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8944–8951.

Sutskever, I., Vinyals, O., and Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

Swarup, P., Maas, R., Garimella, S., Mallidi, S. H., and Hoffmeister, B. (2019). Improving asr confidence scores for alexa using acoustic and hypothesis embeddings. In *Interpeech*.

Tainta, M., Olaso, J. M., Torres, M. I., Ecay-Torres, M., Balluerka, N., Ros, N., Izquierdo, M., Saéz de Asteasu, M., Etxebarria, U., Gayoso, L., Mateo, M., Ibarrondo, O., Alberdi, E., Capetillo-Zárate, E., Bravo, J. A., and Martínez-Lage, P. (2022). The cita go-on trial: A person-centered, digital, intergenerational, and cost-effective dementia prevention multi-modal intervention model to guide strategic policies facing the demographic challenges of progressive aging. In *Iberspeech*.

Takanobu, R., Zhu, H., and Huang, M. (2019). Guided dialog policy learning: Reward estimation for multi-domain task-oriented dialog. In *EMNLP-IJCNLP*.

Tenorio-Laranga, J., Fernández-Ruanova, B., Torres, M. I., Justo, R., Alday, A., and Hernaiz, J. L. (2019). Designing a virtual coach: Involvement of end-users from early design to prototype. *Journal of Ambient Intelligence and Humanized Computing*, 19(4):207.

Tomashenko, N., Raymond, C., Caubrière, A., De Mori, R., and Estève, Y. (2020). Dialogue history integration into end-to-end signal-to-concept spoken language understanding systems. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8509–8513. IEEE.

Torres, M. I. (2013). Stochastic bi-languages to model dialogs. In *Proceedings of the 11th International Conference on Finite State Methods and Natural Language Processing*, pages 9–17, St Andrews, Scotland. Association for Computational Linguistics.

Torres, M. I., Olaso, J. M., Montenegro, C., Santana, R., Vázquez, A., Justo, R., Lozano, J. A., Schlögl, S., Chollet, G., Dugan, N., Irvine, M., Glackin, N., Pickard, C., Esposito, A., Cordasco, G., Troncone, A., Petrovska-Delacretaz, D., Mtibaa, A., Hmani, M. A., Korsnes, M. S., Martinussen, L. J., Escalera, S., Cantariño, C. P., Deroo, O., Gordeeva, O., Tenorio-Laranga, J., Gonzalez-Fraile, E., Fernandez-Ruanova, B., and Gonzalez-Pinto, A. (2019). The empathic project: Mid-term achievements. In *Proceedings of the 12th ACM International Conference on Pervasive Technologies Related to Assistive Environments*, PETRA '19, page 629–638, New York, NY, USA. Association for Computing Machinery.

Tuan, Y.-L. and Lee, H.-Y. (2019). Improving conditional sequence generative adversarial networks by stepwise evaluation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Vinyals, O. and Le, Q. (2015). A neural conversational model. *arXiv preprint arXiv:1506.05869*.

Vázquez, A., López Zorrilla, A., Olaso, J. M., and Torres, M. I. (2023). Dialogue management and language generation for a robust conversational virtual coach. *Applied Sciences (Under revision)*.

Vázquez, A. (2019). EMPATHIC-NLG: un generador de lenguaje natural adaptado al coaching (EMPATHIC-NLG: a natural language generator adapted to coaching). Master's thesis, University of the Basque Country UPV/EHU.

Wang, J., Liu, J., Bi, W., Liu, X., He, K., Xu, R., and Yang, M. (2020). Dual dynamic memory network for end-to-end multi-turn task-oriented dialog systems. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4100–4110.

Wang, W., Zhang, Z., Guo, J., Dai, Y., Chen, B., and Luo, W. (2022). Task-oriented dialogue system as natural language generation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2698–2703.

Welleck, S., Kulikov, I., Roller, S., Dinan, E., Cho, K., and Weston, J. (2020). Neural text generation with unlikelihood training. In *International Conference on Learning Representations*.

Weng, Y., Miryala, S. S., Khatri, C., Wang, R., Zheng, H., Molino, P., Namazifar, M., Papangelis, A., Williams, H., Bell, F., et al. (2020). Joint contextual modeling for asr correction and language understanding. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6349–6353. IEEE.

Whitmore, J. (1992). Growing human potential and purpose: The principles and practice of coaching and leadership.

Williams, J., Raux, A., Ramachandran, D., and Black, A. (2013). The dialog state tracking challenge. In *SIGDIAL*.

Williams, J. D., Atui, K. A., and Zweig, G. (2017). Hybrid code networks: practical and efficient end-to-end dialog control with supervised and reinforcement learning. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 665–677.

Williams, J. D. and Young, S. (2007). Partially observable markov decision processes for spoken dialog systems. *Computer Speech & Language*, 21(2):393–422.

Williams, J. D. and Zweig, G. (2016). End-to-end lstm-based dialog control optimized with supervised and reinforcement learning. *CoRR*, abs/1606.01269.

Williams, R. J. (1992). Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4).

Wolf, T., Sanh, V., Chaumond, J., and Delangue, C. (2019). Transfertransfo: A transfer learning approach for neural network based conversational agents. *Computing Research Repository*, arXiv:1901.08149.

World Health Organization, . et al. (1996). WHOQOL-BREF: introduction, administration, scoring and generic version of the assessment: field trial version, December 1996. Technical report, World Health Organization.

Wu, L., Xia, Y., Zhao, L., Tian, F., Qin, T., Lai, J., and Liu, T.-Y. (2017). Adversarial neural machine translation. *arXiv preprint arXiv:1704.06933*.

Wu, S., Li, Y., Zhang, D., Zhou, Y., and Wu, Z. (2020). Diverse and informative dialogue generation with context-specific commonsense knowledge awareness. In *ACL*.

Wu, S., Li, Y., Zhang, D., Zhou, Y., and Wu, Z. (2021a). Topicka: Generating commonsense knowledge-aware dialogue responses towards the recommended topic fact. In *IJCAI*, pages 3766–3772.

Wu, T.-W., Su, R., and Juang, B.-H. (2021b). A context-aware hierarchical bert fusion network for multi-turn dialog act detection. *arXiv preprint arXiv:2109.01267*.

Xing, C., Wu, W., Wu, Y., Liu, J., Huang, Y., Zhou, M., and Ma, W.-Y. (2016). Topic augmented neural response generation with a joint attention mechanism. *arXiv preprint arXiv:1606.08340*, 2(2).

Xing, C., Wu, W., Wu, Y., Liu, J., Huang, Y., Zhou, M., and Ma, W.-Y. (2017). Topic aware neural response generation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 31.

Xu, A., Liu, Z., Guo, Y., Sinha, V., and Akkiraju, R. (2017). A new chatbot for customer service on social media. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, CHI '17, page 3506–3510, New York, NY, USA. Association for Computing Machinery.

Xu, J., Ren, X., Lin, J., and Sun, X. (2018). Diversity-promoting gan: A cross-entropy based generative adversarial network for diversified text generation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3940–3949.

Xue, L., Constant, N., Roberts, A., Kale, M., Al-Rfou, R., Siddhant, A., Barua, A., and Raffel, C. (2020). mt5: A massively multilingual pre-trained text-to-text transformer. *arXiv preprint arXiv:2010.11934*.

Yan, X. and Nakashole, N. (2021). A grounded well-being conversational agent with multiple interaction modes: Preliminary results. In *Proceedings of the 1st Workshop on NLP for Positive Impact*, pages 143–151.

Yang, S.-w., Chi, P.-H., Chuang, Y.-S., Lai, C.-I. J., Lakhotia, K., Lin, Y. Y., Liu, A. T., Shi, J., Chang, X., Lin, G.-T., et al. (2021). Superb: Speech processing universal performance benchmark. *arXiv preprint arXiv:2105.01051*.

Yang, Y., Cer, D., Ahmad, A., Guo, M., Law, J., Constant, N., Abrego, G. H., Yuan, S., Tar, C., Sung, Y.-H., et al. (2019). Multilingual universal sentence encoder for semantic retrieval. *arXiv preprint arXiv:1907.04307*.

Young, S. J. (2000). Probabilistic methods in spoken–dialogue systems. *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 358(1769):1389–1402.

Young, T., Pandelea, V., Poria, S., and Cambria, E. (2020). Dialogue systems with audio context. *Neurocomputing*, 388.

Yu, J., Li, X., Koh, J. Y., Zhang, H., Pang, R., Qin, J., Ku, A., Xu, Y., Baldridge, J., and Wu, Y. (2021). Vector-quantized image modeling with improved vqgan. *arXiv preprint arXiv:2110.04627*.

Yu, L., Zhang, W., Wang, J., and Yu, Y. (2017). Seqgan: Sequence generative adversarial nets with policy gradient. In *AAAI*, pages 2852–2858.

Zhang, J., Oh, Y. J., Lange, P., Yu, Z., and Fukuoka, Y. (2020a). Artificial intelligence chatbot behavior change model for designing artificial intelligence chatbots to promote physical activity and a healthy diet. *Journal of medical Internet research*, 22(9):e22845.

Zhang, S., Dinan, E., Urbanek, J., Szlam, A., Kiela, D., and Weston, J. (2018). Personalizing dialogue agents: I have a dog, do you have pets too? In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2204–2213.

Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q., and Artzi, Y. (2019). Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Zhang, T., Ramakrishnan, R., and Livny, M. (1996). Birch: An efficient data clustering method for very large databases. In *Proceedings of the 1996 ACM SIGMOD International Conference on Management of Data*, SIGMOD '96, page 103–114, New York, NY, USA. Association for Computing Machinery.

Zhang, Y., Gan, Z., and Carin, L. (2016). Generating text via adversarial training. In *NIPS workshop on Adversarial Training*, volume 21, pages 21–32.

Zhang, Y., Sun, S., Galley, M., Chen, Y.-C., Brockett, C., Gao, X., Gao, J., Liu, J., and Dolan, W. B. (2020b). Dialogpt: Large-scale generative pre-training for conversational response generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 270–278.

Zhao, T., Zhao, R., and Eskenazi, M. (2017). Learning discourse-level diversity for neural dialog models using conditional variational autoencoders. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 654–664.

Zheng, Y., Zhang, R., Huang, M., and Mao, X. (2020). A pre-training based personalized dialogue generation model with persona-sparse data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 9693–9700.

Zhou, H., Huang, M., Zhang, T., Zhu, X., and Liu, B. (2018). Emotional chatting machine: Emotional conversation generation with internal and external memory. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.