OLISIA: a Cascade System for Spoken Dialogue State Tracking
Léo Jacqmin, Lucas Druart, Yannick Estève, Benoît Favre, Lina Maria Rojas-Barahona, Valentin Vielzeuf

To cite this version:
Léo Jacqmin, Lucas Druart, Yannick Estève, Benoît Favre, Lina Maria Rojas-Barahona, et al.. OLISIA: a Cascade System for Spoken Dialogue State Tracking. 2023. hal-04072601v2

HAL Id: hal-04072601
https://hal.science/hal-04072601v2
Preprint submitted on 30 Aug 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Abstract

Though Dialogue State Tracking (DST) is a core component of spoken dialogue systems, recent work on this task mostly deals with chat corpora, disregarding the discrepancies between spoken and written language. In this paper, we propose OLISIA, a cascade system which integrates an Automatic Speech Recognition (ASR) model and a DST model. We introduce several adaptations in the ASR and DST modules to improve integration and robustness to spoken conversations. With these adaptations, our system ranked first in DSTC11 Track 3, a benchmark to evaluate spoken DST. We conduct an in-depth analysis of the results and find that normalizing the ASR outputs and adapting the DST inputs through data augmentation, along with increasing the pre-trained models size all play an important role in reducing the performance discrepancy between written and spoken conversations.

1 Introduction

A majority of recent research on task-oriented dialogue (TOD) systems has focused on chat corpora such as MultiWOZ (Budzianowski et al., 2018). With voice assistants becoming more prominent in our daily lives, there has been a renewed interest in spoken dialogue systems (Faruqui and Hakkani-Tür, 2022). However, state-of-the-art systems trained on chats face robustness issues when dealing with spoken inputs (Kim et al., 2021).

In a TOD system, the role of DST is to predict at each turn and based on the dialogue history the current belief state, i.e., a condensed and updated representation of the user needs. DST plays a central role as the system relies on the belief state to decide which action to take next. The belief state is typically frame-based and represented as a list of <slot, value> pairs.


dstc11/dstc11_20221102a.html

While both cascade and end-to-end approaches have been well studied for Spoken Language Understanding (SLU, Serdyuk et al. (2018)), there has been little recent work on spoken DST. Considering the entire dialogue context, as opposed to only the current turn, requires tricky strategies for end-to-end systems (Tomashenko et al., 2020). In order to leverage state-of-the-art models, a cascade approach with separate ASR and DST components is thus preferred. However, these two components do not benefit from joint optimization and often lack integration.

To address these shortcomings, we propose OLISIA, a cascade system composed of an ASR model and a DST model. OLISIA integrates these two components through adaptations in the ASR outputs and the DST inputs. With this design, our system ranked first in the Speech-Aware Dialog Systems Technology Challenge (DSTC11 Track 3), a benchmark to evaluate spoken DST models.

In this paper, we describe our cascade spoken DST system along with the proposed adaptations. Additionally, we conduct an analysis of these adaptations based on the results from various evaluation
Our contributions can be summarized as follows. In the context of the Speech-Aware Dialog Systems Technology Challenge, we show

- the need for post-processing the ASR output in a pipeline;
- the relevance of different data augmentation techniques for DST;
- the importance of scaling up the foundation model size – both for ASR and DST.

2 Related work

Much of the research in task-oriented dialogue (TOD) systems initially focused on spoken dialogue, for instance leveraging probabilistic modeling to account for the uncertainty associated with noisy utterances (Roy et al., 2000; Williams and Young, 2007; Thomson and Young, 2010). The first editions of the Dialogue State Tracking Challenge (DSTC1 & DSTC2, Williams et al. (2013); Henderson et al. (2014)) introduced the first standardized benchmark for DST, releasing annotated spoken dialogue corpora.

However, the focus of research in TOD gradually shifted to chats, assuming that the upstream ASR model would be able to provide accurate transcriptions. Recently, there has been a renewed interest in spoken dialogue to address the lack of attention on the differences between spoken and textual inputs (Faruqui and Hakkani-Tür, 2022). DSTC10 Track 2 proposed a DST task on spoken conversations which stimulated work on this aspect, though only the n-best ASR hypotheses were provided without audio data.

When audio-only data is available, End-to-End Spoken Language Understanding (SLU) systems (Serdyuk et al., 2018) are often preferred because they benefit from joint optimization. Although cascade approaches suffer from error propagation because the textual Natural Language Understanding (NLU) model does not consider the uncertainty of the ASR transcriptions, they remain competitive. In fact several adaptation techniques can boost the cascade’s performance.

Performing hypothesis rescoring with a language model specifically trained on the targeted domain proves to be effective when data is available in high quality and large quantity (Chung et al., 2012). We rather adopt a post-processing approach such as spelling correction which can also help aligning the transcriptions with the targeted domain (Hrinchuk et al., 2020), especially when focusing on domain-specific words.

With the recent advances in Text-to-Speech (TTS) technologies, adapting ASR models by fine-tuning them on synthetic speech of the target domain (Li et al., 2018; Rosenberg et al., 2019; Zheng et al., 2021) is now common. However, with only synthetic speech in the training set, such a fine-tuning might degrade the performances on human speech (Laptev et al., 2020). Simulating ASR hidden representations from text in order to train an end-to-end SLU model reaches higher Named-Entity Recognition (NER) performances than training it on the synthesised speech (Mdhaffar et al., 2022). Therefore, relying on the text NLU model to tolerate and correct some errors of the ASR model seems more adapted in our setting.

Recent approaches have focused on data augmentation techniques to simulate spoken data (Wang et al., 2020; Liu et al., 2021; Tian et al., 2021) in order to make the language understanding components of TOD systems more robust to spoken inputs. Other approaches have sought to leverage the multiple hypotheses provided by the upstream ASR model in the hope that these different hypotheses complement each other to help language understanding (Rojas-Barahona et al., 2016; Li et al., 2020; Ganesan et al., 2021). Similarly, others used a more compact representation of these hypotheses, such as word confusion networks (Henderson et al., 2012; Pal et al., 2020).

3 Speech-Aware Dialog Systems Technology Challenge

The lack of recent work on spoken dialogue can be attributed in part to the lack of available datasets. Track 3 of the Dialog Systems Technology Challenge seeks to promote work on spoken dialogue by releasing a spoken version of MultiWOZ. This Multi-domain (restaurant, hotel, attraction, taxi, train, hospital and police) Wizard-of-Oz dataset is a large-scale human-human task-oriented conversational corpus commonly used for training and evaluating dialogue state tracking (DST), policy optimization and end-to-end dialogue modeling systems. The goal of this track is to characterize

---

3Rebranded as the Dialog Systems Technology Challenge since DSTC6.

4https://dstc11.dstc.community/
the performance of DST models in the presence of ASR errors and speech phenomena such as disfluencies. The organizers released a new version of MultiWOZ 2.1 (Eric et al., 2019) with user utterances voiced by crowdworkers, as illustrated in Figure 1.

Despite being widely used by the research community, MultiWOZ has been shown to exhibit an entity bias and a large overlap in the distribution of slot-values between the training and the evaluation sets which can lead to memorization in generative models (Qian et al., 2021). To encourage generalization, the organizers introduced modifications in the dev and test sets: the values for the slots hotel-name, restaurant-name, train-departure and train-destination were replaced with unseen entities, and time mentions were offset by a constant amount.

User utterances in the dev and test sets are vocalized by crowdworkers. A speech synthesized version of the training data is also provided in the aim of assessing the validity of such data to mitigate the lack of real spoken conversations.

Two verbatim versions of the dev set are provided to the participants, i.e. user utterances are vocalized as is by a TTS system (TTS-v) and human crowdworkers (Human-v). The test set includes the same setup along with a third version containing paraphrased user utterances vocalized by humans to sound more natural (Human-p).5

System submissions are evaluated using Joint Goal Accuracy (JGA) and Slot Error Rate (SER), defined as follows:

\[
JGA = \frac{C}{N_t} = \frac{\text{No. of correct state pred.}}{\text{No. of turns}}
\]

\[
SER = \frac{S + D + I}{N_s} = \frac{\text{No. of slot errors}}{\text{No. of slots in ref.}}
\]

where \(S\), \(D\) and \(I\) respectively denote substitutions, deletions and insertions of \(<\text{slot, value}>\) pairs. Regarding the challenge constraints, any type of model can be used but only MultiWOZ is allowed as training data for the dialogue component.

4 Method

In this section, we present our cascade approach with an ASR component which converts the user spoken inputs into text and a DST component which predicts the current dialogue state from the transcript of the previous turns. The overall architecture of the system is shown in Figure 2.

On the ASR side, given that the turns are perfectly segmented, we can easily transcribe the user’s turns with OpenAI Whisper (Radford et al., 2022) transformer model with a forced English decoding. On the DST side, we use a generative DST model based on a pre-trained T5 model (Raffel et al., 2020), which proved to be more robust to spoken inputs than an extractive model in preliminary experiments.

The input to the DST model consists of the entire dialogue history at a given turn, with agent and user utterances separated by delimiter tokens. At each turn, the model outputs the current dialogue state from scratch in the form of \(<\text{slot-name, slot-value}>\) pairs. Formally, let \(U_i\) and \(A_i\) respectively be user and agent utterances at turn \(i\). The input at turn \(T\) is linearized by concatenating the utterances \((U_0, A_1, ..., A_{T-1}, U_T)\) and prepending the delimiters "user:" and "agent:". The output dialogue state is linearized as a semicolon-separated list of strings "slot-name=slot-value".

Our contribution lies in the adaptation of the transcriptions outputted from the ASR for DST and the
adaptation of the DST component to handle speech specificities, which are discussed in the next two sections.

4.1 ASR normalization

The first adaptation we apply to Whisper’s transcriptions is time normalization. Given the data sources on which Whisper was trained, the outputted time formats vary a lot. We use several regular expressions to identify the most salient patterns (e.g. "5 o’8 am", "2 to 3 pm", "midnight", "quarter past 10 am") and map them to the standard "[hour]:[minutes] [am|pm]" format found in Multi-Woz.

The second adaptation we apply is proper noun correction which impacts the values of the slots hotel-name, restaurant-name, taxi-destination, train-destination, taxi-departure and train-departure. Many proper nouns are either misspelled by the user (e.g. the American city Itta Bena is pronounced "I-T-T-A bena") or incorrectly recognized by Whisper (e.g. Itta Bena transcribed as "Itta Benna"). In both cases, we use a Named Entity Recognition (NER) model\(^6\) from Nvidia NeMo (Kuchaiev et al., 2019) to identify lists of proper nouns from both agent and user turns. We then score each pair of user and agent identified named entities with Character Error Rate (CER) and tune a threshold in order to replace the user turns’ misspelled proper nouns with their matching ones from the agent turns.

More formally, given a list of user proper nouns \(l_u\), a list of agent proper nouns \(l_a\) and a threshold \(\delta\), we have:

\[
\forall u \in l_u, a \in l_a \ u = \begin{cases} u \text{ if } \text{CER}(u, a) > \delta \\ a \text{ otherwise} \end{cases}
\]

Where \(u\) and \(a\) refer respectively to user and agent proper nouns.

4.2 DST data augmentation

To improve robustness and reduce the discrepancy between training and testing data, our DST model is fine-tuned on an augmented version of the provided train set. We apply value replacement, paraphrasing and speech simulation, in this order.

In a similar way to how the dev and test sets were modified by the track organizers, our value replacement concerns named entities (town, restaurant and hotel names) and time slots. To replace entity values, we create a new ontology based on data from OpenStreetMap\(^7\). We then sample entities with a uniform distribution over our ontology. Values with time mentions are replaced with a random time.

The value replacement process goes as follows:\(^6\)https://docs.nvidia.com/deeplearning/nemo/user-guide/docs/en/v1.0.0/nlp/token_classification.html

\(^7\)https://wiki.openstreetmap.org/wiki/Overpass_API
1. Successively go through each dialogue state in a dialogue, sample one value from our ontology for each distinct value and replace it in the dialogue state;

2. Track these replacements with a mapping between replaced value and new value;

3. Based on the obtained mapping, perform a string replacement in the dialogue context.

This process ensures dialogue consistency: if a value for a slot is updated during the dialogue, a new value is sampled thanks to the first step, if a value is shared between multiple slots, the same replacement value is used thanks to the second step, and finally, the string replacement in the third step is performed by decreasing lengths in order to avoid replacing sub-strings (e.g. city names can be present in restaurant or hotel names).

Based on this new train set with replaced values, we paraphrase the user utterances using SG-GPT (Peng et al., 2020) which allows us to condition the generation of the paraphrase on the previous turn along with the desired dialogue state for the current turn, preventing annotation inconsistencies due to hallucinations or omissions.

Lastly, we obtain the ASR transcripts for the speech simulation by synthesizing the augmented user turns with a Tacotron2-based (Shen et al., 2018) TTS system using SpeechBrain\(^8\) and transcribing them with Whisper.

### 5 Results

We propose an overview of the challenge’s results in section 5.1 and further analysis of the impact of each adaptation on our system’s performance in the following sections.

8\(https://speechbrain.github.io/\)

### 5.1 Overview

The challenge’s leaderboard is shown in Figure 3. All submissions have a gap between their TTS performance and human performance (14.32% average relative JGA decrease and 19.19% average relative SER increase). The gap between the human-verbatim and human-paraphrased is less pronounced (3.36% average relative JGA decrease and 0.08% average relative SER decrease).

Our primary submission \((F-p \text{ in Figure 3})\) consists in an ensemble of 5 instances of our system described in section 4 with transcriptions provided by Whisper-Large\(^9\) and a T5-Large model\(^10\) fine-tuned on the variations of the training data presented in Table 2. Replace-S refers to one version of the train set with value replacement. With Replace-L, a different version of the train set is used at each epoch until convergence,\(^11\) with newly sampled entities for value replacement along with the TTS-ASR and optional paraphrasing pipeline.

We used majority vote on each predicted slot value as ensembling strategy. Our secondary submission \((F-s \text{ in Figure 3})\) consists in the best instance of the models in the ensemble (fine-tuning on \(T_2\)). For all models, we use a learning rate of \(5e^{-4}\) and a batch size of 16. We compare their performance on the dev and test set with the challenge baseline and our system with text oracle\(^12\) in Table 1. Note that we exclude the ASR-TTS data augmentation for the system evaluated on the text oracle (+5 JGA increase).

While there is still room for improvement (10

---

<table>
<thead>
<tr>
<th>Dev</th>
<th>TTS-v</th>
<th>Human-v</th>
<th>Test</th>
<th>TTS-v</th>
<th>Human-v</th>
<th>Human-p</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>JGA↑</td>
<td>SER↓</td>
<td></td>
<td>JGA↑</td>
<td>SER↓</td>
<td>JGA↑</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Baseline</td>
<td>26.3</td>
<td>27.5</td>
<td>22.6</td>
<td>31.6</td>
<td>n.a.</td>
<td></td>
</tr>
<tr>
<td>OLISIA1</td>
<td>47.2</td>
<td>15.7</td>
<td>43.2</td>
<td>17.9</td>
<td></td>
<td></td>
</tr>
<tr>
<td>OLISIA2</td>
<td>44.1</td>
<td>17.3</td>
<td>40.3</td>
<td>19.5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>OLISIA1 (oracle)</td>
<td>57.2</td>
<td>12.5</td>
<td>53.2</td>
<td>13.9</td>
<td></td>
<td></td>
</tr>
<tr>
<td>OLISIA2 (oracle)</td>
<td>55.0</td>
<td>13.6</td>
<td>51.1</td>
<td>15.0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 1: Performance (JGA↑ / SER↓) of our submission compared with the challenge’s baseline and our system with text oracle on both dev and test sets. The baseline results on the test set were not shared.

---

9\(https://github.com/openai/whisper\)

10\(https://huggingface.co/t5-large\)

11Four epochs for T5-Large.

12Text oracle considers the ground truth user turns’ transcriptions. We provide these results in order to give an upper bound of our system’s performances.
JGA points between our system with and without text oracle), both our submissions achieved over 40 JGA on the TTS-verbatim test set. Our performance decrease (4 JGA points) from the TTS-verbatim to the Human-verbatim version is steady across the dev and test sets. We observe a smaller decrease (2 JGA points) from the Human-verbatim to the Human-paraphrased version. Finally, the difference between the dev and test sets (4 JGA points) can be explained by overfitting, a difficulty difference, or both.

<table>
<thead>
<tr>
<th>Replace-S</th>
<th>Replace-L</th>
<th>TTS-ASR</th>
<th>Paraphrase</th>
</tr>
</thead>
<tbody>
<tr>
<td>T1</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>T2</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>T3</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>T4</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>T5</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

Table 2: Composition of the training sets used to fine-tune the models in the ensemble for our primary submission (OLISIA\(_1\)); * denotes the augmented data provided by the track organizers.

5.2 ASR cascade adaptations

Given that only a few words in the users’ utterances are really important to the dialogue state, Word Error Rate (WER) is not a good measure for the quality of the transcriptions (Wang et al., 2003). Therefore we compute both JGA and WER at each correction step with the same T5-Large trained on the training set with value replacements. We present the results in Table 3.

<table>
<thead>
<tr>
<th>Whisper raw outputs</th>
<th>TTS-v</th>
<th>Human-v</th>
<th>Human-p</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>37.9 / 4.92</td>
<td>33.8 / 8.4</td>
<td>32.0 / _</td>
</tr>
<tr>
<td>+ Time normalization</td>
<td>40.0 / 4.49</td>
<td>35.6 / 7.89</td>
<td>33.5 / _</td>
</tr>
<tr>
<td>+ Noun correction</td>
<td>40.3 / 4.36</td>
<td>36.1 / 7.71</td>
<td>34.3 / _</td>
</tr>
</tbody>
</table>

Table 3: Impact of the ASR post-processing steps on the test set performances (JGA\(^\uparrow\) / WER\(^\downarrow\)).

Unsurprisingly, the TTS-verbatim version of the test set is much cleaner than the Human-verbatim version because of the higher diversity of the crowd-workers pronunciations compared to the synthetic voices. Hence we observe over 70% relative WER increase on every correction’s step outputs. This noisier version only impacts the JGA by around 10% relative JGA decrease, confirming that only a few words matter to DST.\(^{13}\)

Time normalization improves equally all three versions of the test set (around 5% relative JGA increase).

Proper noun correction does not help much the clean TTS-verbatim (0.75% relative JGA increase), however, it seems to be much more valuable for the noisier Human-verbatim and Human-paraphrased versions (respectively 1.4% and 2.4% relative JGA increase) again illustrating differences between synthetic and natural speech.

5.3 Data augmentation strategies

In order to better understand how each data augmentation technique we used contributes to the overall performance, we conduct an ablation study on different versions of the training data. We incrementally add one augmentation technique at a time to the default train set and fine-tune a T5-Large model on each version. The results are shown in Table 4.

Unsurprisingly, the TTS-verbatim version of the test set is much cleaner than the Human-verbatim version because of the higher diversity of the crowd-workers pronunciations compared to the synthetic voices. Hence we observe over 70% relative WER increase on every correction’s step outputs. This noisier version only impacts the JGA by around 10% relative JGA decrease, confirming that only a few words matter to DST.\(^{13}\)

Note that WER values are missing for the Human-

We observe that introducing new values for the slots hotel-name, restaurant-name, train-departure and train-destination greatly alleviates the issue of memorization with MulitWOZ and enables the model to generalize more, with a 7% to 8% absolute increase in JGA. The speech simulation provides an additional slight improvement which is particularly marked on the Human-paraphrased test set (+1%). This shows that using speech synthesized training data in the absence of real spoken data can help address spoken dialogues at inference. On the other hand, paraphrasing the user utterances leads to an overall worse performance. One possible explanation for this result is the noisy nature of the process, using a generation model can lead to potential inconsistencies both in the flow of the dialogue and in the annotation.

\(^{13}\)Note that WER values are missing for the Human-paraphrased test set as no ground truth transcripts were provided for this version.
5.4 Model size

When dealing with noisy data and robustness issues we often observe that models with more parameters perform better. However, there is a trade-off between the computation resources needed for large models and the performance gains. In this section we attempt to explore this trade-off by exposing the performance gained by each model size increase. For the ASR part, we consider Whisper-Small (244M), Whisper-Medium (769M) and Whisper-Large (1550M). For the DST part, we consider T5-Small (60M), T5-Base (220M), T5-Large (738M) and T5-XL (3B), fine-tuning each model on our best training set ($T_2$). We present the impact of the size of the ASR and DST models respectively in Figure 4 and 5.

Interestingly JGA does not increase much when moving from T5-Small to T5-Base whereas it increases by almost 6 points when using T5-Large instead of T5-Base. The performance then drops back 6 points when using T5-XL, although this larger model seems to be more robust to the paraphrased test set. This suggests that XL models tend to overfit, while Large models provide a good compromise between the number of parameters and generalization. For Whisper this trend is different: using Whisper-Medium instead of Whisper-Small increases JGA of at least 2 points on the Human-verbatim dev set while using Whisper-Large instead of Whisper-Medium only increases JGA of 1 point or less. It is noteworthy that the lower parameter ratio between the Large and Medium models might explain this lower JGA increase.

While our whisper models were not fine-tuned on any data, we can already observe that the decrease of WER obtained by using Whisper-Large instead of Whisper-Medium on the TTS-verbatim version is not found on the Human-verbatim version.

5.5 Ensemble

We compare two different ensembling strategies with a single model fine-tuned on $T_2$ in Table 5 (all models are based on T5-Large). Both ensembling strategies consisted of a majority vote for each slot-value. In one case, we used five models fine-tuned on the same train set with different random seeds and in the other, we fine-tuned five models on five different variations of the train set (c.f. Table 2).

<table>
<thead>
<tr>
<th></th>
<th>TTS-v</th>
<th>Human-v</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 model (no ensemble)</td>
<td>44.1</td>
<td>40.2</td>
</tr>
<tr>
<td>5 models - same train set</td>
<td>44.4</td>
<td>40.4</td>
</tr>
<tr>
<td>5 models - different train sets</td>
<td>47.8</td>
<td>43.5</td>
</tr>
<tr>
<td>9 models - different train sets</td>
<td>48.5</td>
<td>43.9</td>
</tr>
</tbody>
</table>

Table 5: Comparison of different ensembling strategies on the dev set in terms of JGA.

We find that the value of the ensemble from the
same train set is limited, only providing a slight increase in JGA compared to the single model. The advantage of ensembling appears with five models fine-tuned on different train sets, providing a 3% absolute increase in JGA. This suggests that the ensemble benefits from having contrasting views of the same instance at evaluation, and by extension that T5 models fail to learn invariant representations of proper nouns (or that our method does not enable that). It is also noteworthy that performance does not increase that much beyond 5 models, with marginally better results at 9 models, likely showing a performance ceiling.

6 Limitations

Overall, the dataset released in this challenge is a good step towards bridging the gap between written and spoken dialogue systems. However, as the user utterances were read aloud by humans, this spoken data lacks in spontaneity associated with actual speech. It would be interesting to see if our findings hold on spontaneous spoken dialogue.

One limitation of our system lies in the use of Transformer-based models for both ASR and DST. While these models provide attractive performances they come with their own limitation of quadratic memory and limited input size. Dialogues in MultiWOZ are relatively short and this was not a concern in this work, but this could become problematic when dealing with longer, more realistic conversations. One alternative would be to reduce the input context to the most recent turns and include the linearized previous dialogue state in the DST input.

Another concern is the use of large pre-trained models to achieve competitive results. As pointed out before (Strubell et al., 2019), training these large models on abundant data requires a substantial electrical consumption. In light of the human impact on the environment, we should promote efficiency as the main performance factor rather than metric scores.

7 Conclusion and future work

This work introduced OLISIA, a cascade system for spoken DST that integrates an ASR and a DST model through several adaptations. We used ASR normalization and DST data augmentation to adjust each component to its counterpart. We have shown the importance of these adaptations in improving robustness to spoken inputs and our system achieved first place in the Speech-Aware Dialog Systems Technology Challenge.

While having user turns as speech and agent turns as text is a natural setup for a spoken dialogue system, this mix of modalities makes it more challenging to develop end-to-end systems. Recent progress on speech and text multimodal models could prove to be useful in addressing this problem (Ao et al., 2022). Another possibility would be to exploit agent turns using an audio only model, either as synthesized speech or as intermediary ASR representations (Mdahaffar et al., 2022).

This work focused on a cascade approach but a thorough comparison of end-to-end and cascade approaches could also be helpful in further research on spoken dialogue systems.

Acknowledgements

We thank Marco Dinareli for the valuable discussions and fruitful suggestions regarding end-to-end spoken DST.

This work was partially funded by the ANR CIFRE contracts N°2022/0286 and N°2022/0011 and by Orange Innovation.

This work was performed using HPC resources from GENCI-IDRIS (Grant 2022-AD011011407R2).

References


