



**HAL**  
open science

## Are cascade dialogue state tracking models speaking out of turn in spoken dialogues?

Lucas Druart, Léo Jacqmin, Benoît Favre, Lina Maria Rojas-Barahona,  
Valentin Vielzeuf

### ► To cite this version:

Lucas Druart, Léo Jacqmin, Benoît Favre, Lina Maria Rojas-Barahona, Valentin Vielzeuf. Are cascade dialogue state tracking models speaking out of turn in spoken dialogues?. 2023. hal-04267783

**HAL Id: hal-04267783**

**<https://hal.science/hal-04267783>**

Preprint submitted on 2 Nov 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.

# ARE CASCADE DIALOGUE STATE TRACKING MODELS SPEAKING OUT OF TURN IN SPOKEN DIALOGUES?

Lucas Druart<sup>\*1,2</sup> Léo Jacqmin<sup>\*1,3</sup> Benoit Favre<sup>3</sup> Lina Maria Rojas-Barahona<sup>1</sup> Valentin Vielzeuf<sup>1</sup>

<sup>1</sup> Orange Innovation, France <sup>2</sup> LIA - Avignon Université, France <sup>3</sup> LIS - Aix-Marseille Université, France

<sup>1</sup>{lucas1.druart, leo.jacqmin, linamaria.rojasbarahona, valentin.vielzeuf}@orange.com

<sup>2</sup>{first.last}@univ-avignon.fr <sup>3</sup>{first.last}@lislalab.fr \*equal contribution

## ABSTRACT

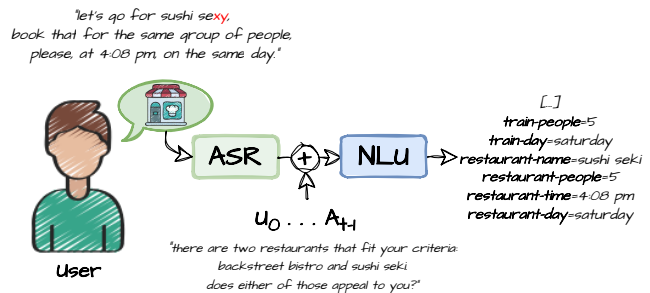
In Task-Oriented Dialogue (TOD) systems, correctly updating the system’s understanding of the user’s needs is key to a smooth interaction. Traditionally TOD systems are composed of several modules that interact with one another. While each of these components is the focus of active research communities, their behavior in interaction can be overlooked. This paper proposes a comprehensive analysis of the errors of state of the art systems in complex settings such as Dialogue State Tracking which highly depends on the dialogue context. Based on spoken MultiWoz, we identify that errors on non-categorical slots’ values are essential to address in order to bridge the gap between spoken and chat-based dialogue systems. We explore potential solutions to improve transcriptions and help dialogue state tracking generative models correct such errors.

**Index Terms**— spoken dialogue systems, context adaptation, cascade systems, error analysis

## 1. INTRODUCTION

Task-Oriented Dialogue (TOD) systems are designed to interactively assist a user with a challenging task such as making a reservation at a restaurant or booking a room in a hotel. A typical approach to implement them is to use a modular architecture [1]. A core component of such systems is Dialogue State Tracking (DST) whose goal is to update a condensed representation of the user needs up to a given turn. Based on this representation, the dialogue system decides which action to take next towards completing the task.

While spoken dialogue was the focus of earlier work [2], recent work on DST has focused on text inputs with no regard for the specificities of spoken language [3]. The underlying assumption is that Automatic Speech Recognition (ASR) and DST can be treated separately and advances from both lines of work effectively integrated. However, this approach fails to account for the differences between written and spoken language [4]. By studying both ASR and DST jointly, this paper identifies remaining challenges to bridge the gap between spoken and written DST.



**Fig. 1.** Spoken DST cascade system at inference for turn  $t$ . Red characters indicate transcription errors. Such examples illustrate the challenge of contextually aware predictions.

DST models are robust to some transcription errors [5], however we do not yet have a clear classification of the errors and whether they will impact the final output. This paper provides a comprehensive analysis and categorization of errors which occur between the ASR and DST modules. When comparing spoken cascade DST to text DST, we find that the main performance degradation occurs on non-categorical slots which values come from an open set. Though some enhancements to the system such as specific post-processing of the ASR transcriptions and augmenting the DST training data can mitigate some of this degradation, the gap between spoken and text DST remains. Our analysis highlights potential directions to address these issues.

## 2. RELATED WORK

Early editions of the Dialogue System Technology Challenge (DSTC) focused on spoken dialogue [6, 7]. However, unlike modern DST systems which operate directly on the user utterances [3], the proposed DST systems operated on turn-level Spoken Language Understanding (SLU) outputs: at the time the challenge was to account for the uncertainty of the ASR and SLU [8, 9], in part due to their lower performances.

Following this series of work, DST has been mainly studied in the context of chat corpora [10, 11]. The more re-

cent editions of DSTC have redirected the focus towards spoken dialogue. DSTC10 [12] provided ASR transcriptions on which to build DST adaptation methods for this specific set of transcriptions. The latest edition DSTC11 goes further and provides audio recordings [13]. Our analysis is performed in this novel context and proposes a comprehensive study of the sources of errors.

Both End-to-End (E2E) [14] and cascade approaches [15] have been studied extensively to solve different SLU tasks [16] with independent utterances. Error analysis [17] of E2E and cascade models using a same Wav2Vec2.0 speech encoder underlines that the E2E fine-tuning degrades the model’s transcription capabilities compared to its cascade counterpart, especially on unseen slot-value pairs.

This paper focuses on the behavior of SOTA cascade systems (e.g. native ASR correction, coping with hallucinations, learning with synthetic data augmentation) for contextually aware semantic extraction in which the previous dialogue turns are mandatory to correctly process the current one (e.g. cross-turn references).

### 3. SPOKEN DIALOGUE STATE TRACKING

#### 3.1. Spoken version of MultiWOZ

MultiWoz is a human-human chat-based English TOD dataset commonly used for training and evaluating dialogue systems [18]. A spoken version with vocalized user turns was published in the context of the “*Speech Aware*” track of DSTC11<sup>1</sup> [13]. The user utterances in the dev and test sets (**Dev|Test**) are available as both synthetic and human speech (**TTS|Human**), whereas the training set is only available as synthetic speech. The dataset contains close to 10,000 dialogues with a 80/10/10 train-dev-test split and an average of 13.3 turns per dialogue. Among the pre-defined slots, we can distinguish three groups: categorical slots with a closed set of values, non-categorical slots with an open set of values (~30%) and time slots (~10%).

Additionally, due to a high overlap in the value distributions of the training and evaluation sets [19] which favors memorization, the values for non-categorical slots in the dev and test sets of the spoken version were replaced with new values from an unknown database. Time slots were also offset by a constant amount.

Our analysis draws on a turn-level exact match metric known as Joint-Goal Accuracy (JGA $\uparrow$ ), a Slot Type Accuracy (STA $\uparrow$ ) and a per Slot-type value Precision (SP $\uparrow$ ) to distinguish errors.

#### 3.2. Automatic Speech Recognition

We benchmark three different established ASR approaches (Robust Wav2Vec [20], Conformer-Transducer [21] and Ope-

nAI Whisper [22]) on the Dev-Human data with a common DST model (Table 1) and retain the best one for our experiments.

	WER $\downarrow$	JGA $\uparrow$
<b>Conformer-Transducer</b> (600 M)	10.00	27.10
<b>Robust Wav2Vec</b> (315 M)	16.41	23.90
<b>Whisper</b> (769 M)	<b>8.04</b>	<b>29.30</b>

**Table 1.** WER and JGA of 3 different ASR approaches.

#### 3.3. Dialogue State Tracking

The used DST approach is ranked as the best system of the challenge [23]. It is more robust to ASR noise compared to extractive approaches [5]. It relies on a pre-trained T5 [24] as backbone model which is further fine-tuned on the MultiWOZ dataset.

The encoder is inputted the entire dialogue history at turn  $t$  composed of user and agent turn pairs denoted as  $X_t = [U_1, A_1, \dots, U_t, A_t]^2$ . To preserve the dialogue structure, agent and user turns are separated by the special tokens `agent:` and `user:`. The target dialogue state  $DS_t$  is linearized as a comma separated list of `<slot, value>` pairs: `slot1=value1; ...; slotn=valuen`.

For ease of comparison, while keeping a low computational cost, all the experiments are conducted with fine-tuned T5-small models on different versions of the training data with replaced values, with a learning rate of  $5e^{-4}$  and a batch size of 16.

#### 3.4. Enhancements

We propose enhancements to this core “vanilla” model and analyze their impact on the different error sources.

##### 3.4.1. Improving transcriptions

We infer the original MultiWoz data normalisation rules and apply them to the ASR transcriptions for the DST to be able to rely on the same cues. We use the whisper normalizer and format all time mentions to `[hours]:[minutes] [am|pm]`.

Then we propose to correct non-categorical slot values. We first identify the lists of proper nouns in agent and user turns with a BERT based Named-Entity Recognition (NER) model from the NeMo toolkit [25]. We then score each pair of user and agent identified named entities with Character Error Rate (CER) and tune a threshold to replace the user turns’ misspelled proper nouns with the corresponding match from the agent turns.

<sup>2</sup>If the entire dialogue history exceeds the model’s maximum input length, the input is truncated to include only the most recent turns.

<sup>1</sup><https://dstc11.dstc.community/>

### 3.4.2. Improving state tracking

In order to improve the transcription error robustness of the DST model, we augment the training data with examples which present such errors.

Our first approach consists in simulating ASR errors by injecting three types of character-level errors in the ground-truth transcriptions. We randomly insert, delete, or replace a character in the span corresponding to a non-categorical slot value. All three injections can be applied multiple times and combined. New characters are sampled according to a character-level error matrix between Whisper tiny’s predicted transcription and the ground-truth ones over the training set. We refer to the DST model fine-tuned on this data as **Rule-error DST**.

Our second approach (**TTS-ASR DST**) consists in a TTS-ASR pipeline to simulate more realistic speech transcriptions errors. We synthesize user turns with a Tacotron2-based [26] TTS system and transcribe them with an ASR system composed of a CRDNN with CTC/Attention and an RNN LM trained on LibriSpeech.

## 4. ERROR ANALYSIS

We present the main results of an oracle (predictions made on ground-truth transcripts), baseline, and enhanced models in Table 2, along with our challenge performance (obtained with larger models) as a frame of reference.

	Dev		Test	
	TTS	Human	TTS	Human
<b>Text Oracle (JGA/STA)</b>	50.4 / 65.7		47.5 / 61.2	
DSTC11 Baseline	-	-	27.7 / -	23.6 / -
DSTC11 1 <sup>st</sup> place	47.2 / -	43.2 / -	44.0 / -	39.5 / -
<b>W.+Text DST</b>	33.7 / <b>65.0</b>	27.7 / 63.2	31.4 / 61.2	26.5 / 58.3
+ normalization	36.6 / 64.5	32.0 / 62.6	35.0 / 59.9	30.4 / 57.7
+ NER correction	38.1 / 64.6	33.4 / 62.6	35.8 / 59.9	31.6 / 57.6
<b>W.+Rule-error DST</b>	33.8 / 63.7	28.2 / 61.7	36.7 / 70.4	30.0 / <b>67.2</b>
+ normalization	36.3 / 63.1	31.6 / 61.2	40.3 / 69.4	34.5 / 66.6
+ NER correction	37.2 / 62.9	32.6 / 61.0	<b>40.9 / 69.4</b>	<b>35.5 / 66.4</b>
<b>W.+TTS-ASR DST</b>	33.9 / 61.6	29.3 / 61.0	31.0 / 58.5	27.8 / 56.8
+ normalization	38.1 / 63.7	34.4 / 63.2	36.0 / 60.2	32.0 / 58.6
+ NER correction	<b>38.8 / 63.8</b>	<b>34.7 / 63.3</b>	36.4 / 60.2	32.6 / 58.5

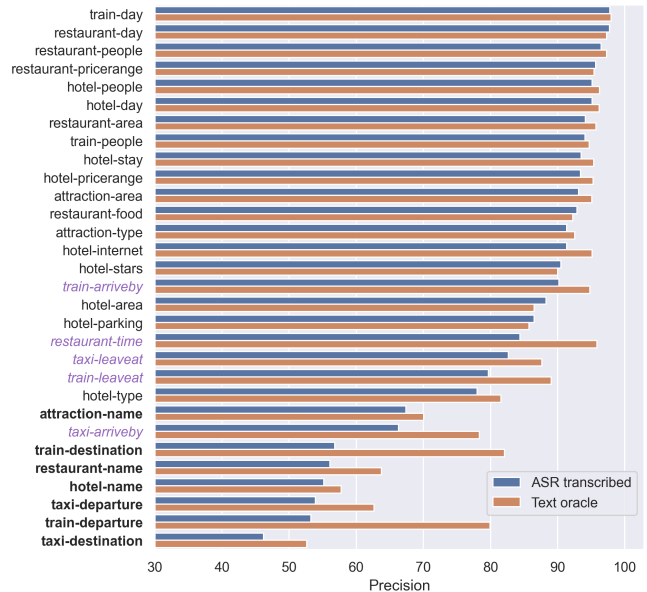
**Table 2.** Main JGA / STA results for spoken DST. Note that Text Oracle shows the upper-bound performance of the model and W. stands for Whisper.

There is a large gap in JGA between the oracle and ASR results, but as we observe a small difference in terms of STA (respectively 65.7% and 63.2%, of which around 40% are omissions), we can already narrow down the issue to value prediction rather than slot identification. We note that both improvements to the transcriptions and to the state tracking contribute to boost performances, especially for human speech, but do not completely recover the difference with the oracle.

We further focus on the **Dev-Human** set since it is the targeted setting and, unless mentioned otherwise, we use **W.+TTS-ASR DST** with normalization and NER correction.

### 4.1. What is different in spoken DST?

In order to explain the gap observed between the performance on the ground-truth transcriptions compared to the predicted transcriptions we present each Slot-type’s value Precision in Fig. 2.



**Fig. 2.** SP comparison between DST predictions on the text oracle and the Whisper transcriptions of the human dev set. Non-categorical slots are shown in bold and time slots in purple.

Non-categorical slots and time slots explain most of this performance gap with locations (*i.e.* values of slots which concern departure and destination of train or taxi trips) being particularly challenging. We observe around an 18% macro-averaged SP decrease for non-categorical slots, a 10% decrease for time slots, and a 1% decrease for categorical slots between DST with ground-truth transcriptions and with ASR transcriptions.

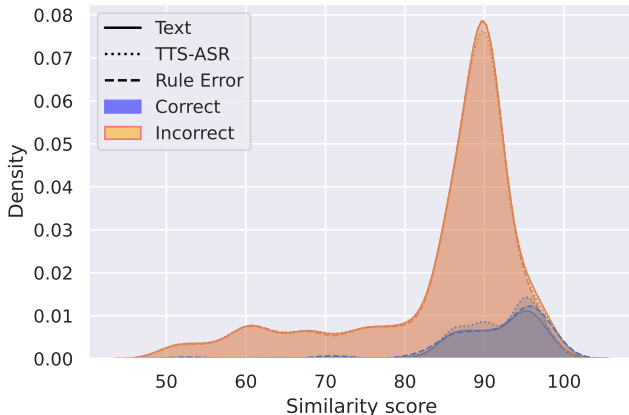
Considering that the main leverage lies in improving predictions for non-categorical slots, the remainder of our analysis focuses on said slots.

### 4.2. Where do these errors come from?

To get a better sense of what kind of errors affects the system overall, we categorize errors for non-categorical slots predictions based on whether the predicted value in the dialogue state matches the reference value, and if the reference value can be found in the dialogue history. A large proportion of

values are incorrect in both the context and the dialogue state (**DS no match, Context no match**) and some values are not correctly identified by the DST module while they can be found in the context (**DS no match, Context match**). We can also note that the generative nature of the DST module enables it to correct some mistranscribed values (**DS match, Context no match**).

If we focus on the case **DS match, Context no match**, one might wonder how far away the best matching span is. We compute the transcribed value similarity by computing the Levenshtein distance between the ground truth  $n$ -gram slot value and all  $n$ -grams from the ASR transcriptions. The final similarity score is the score of the closest  $n$ -gram. Fig. 3 illustrates that only values which are reasonably well transcribed are rightfully corrected by the DST (blue density shifted on the right). However the DST is far from correcting all such slot values (orange density higher than blue one).



**Fig. 3.** Distribution of the similarity scores between ASR mistranscribed values in the dialogue history and the corresponding gold values whether the DST corrected the value.

By further manually reviewing samples of corrections made by the DST model (*i.e.* **DS match, Context no match**), we find that these corrections occur in two broad categories: (i) actual corrections made by the language model (*e.g.* "huntington" corrected as "huntingdon", an English town), and (ii) correction due to normalized spelling in the training data (*e.g.* apostrophes ignored, British English spelling, systematic misspelling of "portugese"). Additionally, we review failures of the DST model (*i.e.* **DS no match, Context match**) and find that most of these errors are due to hallucinations, the main drawback of generative models (*e.g.* "boney to boyd" predicted as "boisey to boise"), or confusions in the context (*e.g.* inversion of departure and destination, user repair).

## 5. TOWARDS POTENTIAL SOLUTIONS

### 5.1. DST correction capacity

Based on Fig. 3, an improvement avenue consists in improving the correction capacity of the DST module on the transcriptions close to the ground-truth values *i.e.* increasing the blue density for similarity scores over 80%. Both the Rule Error and TTS-ASR training regimes for DST slightly improves this correction capacity. Therefore investigating the mechanisms behind such corrections might help us improve spoken dialogue systems performance. A first observation is that only values with high similarity with their target are corrected.

### 5.2. Importance of context

End-to-End approaches might reduce cascading errors thanks to joint-optimization, however such systems are challenging to design since they require careful context propagation given that audio inputs are larger than textual inputs. To highlight this point we assess the importance of context by using the ground-truth transcriptions of the user turns in all but the current turn and comparing that with all user turns transcribed by the ASR. The results in Table 3 show that using the ground-truth transcriptions in the previous context brings a +6 JGA increase.

	All turns as ASR hyp.	Text oracle for turns 0 : t - 1
JGA	34.0	40.1

**Table 3.** Analysis of the importance of the dialogue history.

## 6. CONCLUSION

Since DSTC2 [7], ASR systems have greatly improved, however, this has not led to a corresponding improvement for spoken DST. Through our error analysis, we have underlined that the main leverage point to bridge the gap between text and spoken DST is non-categorical slots' values. For such slots values, we have identified the need for both reducing ASR errors distance to target and increasing DST corrections. While our error injection data augmentation strategy slightly increases this correction capacity, further work in this direction is needed.

The issue of open-set values should also be investigated in a complete dialogue system scenario which includes a realistic database of potential entities of interest to assess whether these errors can be corrected with thanks to the database.

While for computational reasons our study focuses on Whisper-Medium and T5-Small models, we achieved a 3 to 4 points increase in JGA with larger models [23]. The way our conclusions scale with model size remains to be studied in future work.

## 7. REFERENCES

- [1] Jianfeng Gao, Michel Galley, and Lihong Li, “Neural Approaches to Conversational AI,” in *ACM SIGIR*, 2018.
- [2] Jason D. Williams, Antoine Raux, and Matthew Henderson, “The Dialog State Tracking Challenge Series: A Review,” *Dialogue & Discourse*, 2016.
- [3] Yue Feng, Yang Wang, and Hang Li, “A Sequence-to-Sequence Approach to Dialogue State Tracking,” in *ACL*, 2021.
- [4] Manaal Faruqui and Dilek Hakkani-Tür, “Revisiting the Boundary between ASR and NLU in the Age of Conversational Dialog Systems,” *Computational Linguistics*, 2022.
- [5] Hyundong Cho, Chinnadhurai Sankar, and Christopher Lin et al., “Know thy strengths: Comprehensive dialogue state tracking diagnostics,” in *EMNLP*, 2022.
- [6] Jason Williams, Antoine Raux, and Deepak Ramachandran et al., “The Dialog State Tracking Challenge,” in *SIGDIAL*, 2013.
- [7] Matthew Henderson, Blaise Thomson, and Jason D Williams, “The second dialog state tracking challenge,” in *SIGDIAL*, 2014.
- [8] Matthew Henderson, Milica Gašić, and Blaise Thomson et al., “Discriminative spoken language understanding using word confusion networks,” in *SLT*, 2012.
- [9] Gokhan Tur, Anoop Deoras, and Dilek Hakkani-Tür, “Semantic parsing using word confusion networks with conditional random fields,” in *Interspeech*, 2013.
- [10] Chien-Sheng Wu, Andrea Madotto, and Ehsan Hosseini-Asl et al., “Transferable Multi-Domain State Generator for Task-Oriented Dialogue Systems,” in *ACL*, 2019.
- [11] Michael Heck, Carel van Niekerk, and Nurul Lubis et al., “TripPy: A Triple Copy Strategy for Value Independent Neural Dialog State Tracking,” in *SIGDIAL*, 2020.
- [12] Seokhwan Kim, Yang Liu, and Di Jin et al., “‘‘how robust r u?’’: Evaluating task-oriented dialogue systems on spoken conversations,” *ASRU*, 2021.
- [13] Hagen Soltau, Izhak Shafran, and Mingqiu Wang et al., “Speech aware dialog system technology challenge (dstc11),” *SIGDIAL*, 2023.
- [14] Dmitriy Serdyuk, Yongqiang Wang, and Christian Fuegen et al., “Towards end-to-end spoken language understanding,” in *ICASSP*, 2018.
- [15] Ngoc Thang Vu, Pankaj Gupta, Heike Adel, and Hinrich Schütze, “Bi-directional recurrent neural network with ranking loss for spoken language understanding,” in *ICASSP*, 2016.
- [16] Libo Qin, Tianbao Xie, and Wanxiang Che et al., “A survey on spoken language understanding: Recent advances and new frontiers,” in *IJCAI*, 2021.
- [17] Salima Mdhaffar, Valentin Pelloin, and Antoine Caubrière et al., “Impact Analysis of the Use of Speech and Language Models Pretrained by Self-Supervision for Spoken Language Understanding,” in *LREC*, 2022.
- [18] Paweł Budzianowski, Tsung-Hsien Wen, and Bo-Hsiang Tseng et al., “Multiwoz - a large-scale multi-domain wizard-of-oz dataset for task-oriented dialogue modelling,” in *EMNLP*, 2018.
- [19] Kun Qian, Ahmad Beirami, and Zhouhan Lin et al., “Annotation Inconsistency and Entity Bias in MultiWOZ,” in *SIGDIAL*. 2021, *ACL*.
- [20] Wei-Ning Hsu, Anuroop Sriram, and Alexei Baevski et al., “Robust wav2vec 2.0: Analyzing domain shift in self-supervised pre-training,” in *Interspeech*, 2021.
- [21] Anmol Gulati, James Qin, and Chung-Cheng Chiu et al., “Conformer: Convolution-augmented Transformer for Speech Recognition,” in *Interspeech*, 2020.
- [22] Alec Radford, Jong Wook Kim, and Tao Xu et al., “Robust speech recognition via large-scale weak supervision,” *ArXiv*, 2022.
- [23] Léo Jacqmin, Lucas Druart, and Yannick Estève et al., “OLISIA: a Cascade System for Spoken Dialogue State Tracking,” in *SIGDIAL*, 2023.
- [24] Colin Raffel, Noam Shazeer, and Adam Roberts et al., “Exploring the limits of transfer learning with a unified text-to-text transformer,” *Machine Learning Research*, 2020.
- [25] Oleksii Kuchaiev, Jason Li, and Huyen Nguyen et al., “Nemo: a toolkit for building ai applications using neural modules,” 2019.
- [26] Jonathan Shen, Ruoming Pang, and Ron J. Weiss et al., “Natural tts synthesis by conditioning wavenet on mel spectrogram predictions,” in *ICASSP*, 2018.