ANALYSIS OF THE IMPACT OF DATASET QUALITY ON TASK-ORIENTED DIALOGUE MANAGEMENT

Miguel Ángel Medina-Ramírez[1],Cayetano Guerra-Artal[1] and Mario Hernández-Tejera[1]

[1]University Institute of Intelligent Systems and Numeric Applications in Engineering, University of Las Palmas de Gran Canarias, Las Palmas de Gran Canarias, Spain

ABSTRACT

Task-oriented dialogue systems have become crucial for users to interact with machines and computers using natural language. One of its key components is the dialogue manager, which guides the conversation towards a good goal for the user by providing the best possible response. Previous works have proposed rule-based systems, reinforcement learning, and supervised learning as solutions for correct dialogue management; in other words, select the best response given input by the user. This work explores the impact of dataset quality on the performance of dialogue managers. We delve into potential errors in popular datasets, such as Multiwoz 2.1 and SGD. For our investigation, we developed a synthetic dialogue generator to regulate the type and magnitude of errors introduced. Our findings suggest that dataset inaccuracies, like mislabeling, might play a significant role in the challenges faced in dialogue management.

KEYWORDS

Dialog Systems, dialogue management, dataset quality, supervised learning

1. INTRODUCTION

Task-oriented dialogue systems (TODS) are a specialised Natural Language Processing (NLP) class designed to enable users to interact with computer systems to accomplish specific tasks. TODS represent a highly active research area due to their potential to improve human-computer interaction and provide users with seamless and efficient task completion. Recent advancements in Artificial Intelligence (AI) and Machine Learning (ML) have fuelled the proliferation of TODS and the exploration of novel architectures and techniques. One of the most widely used approaches due to its simplicity and controllability is the modular pipeline approach [1,2,3], as shown in Figure 1. It consists of four modules:

- **Natural Language Understanding (NLU):** This module transforms the raw user message into user intentions, slots, and domains. However, some recent modular systems [4] omit this module and use the raw user message as the input of the next module.
- **Dialogue State Tracking (DST):** This module iteratively calibrates the dialogue states based on the current input and dialogue history. The dialogue state includes related user intentions and slot-value pairs.

David C. Wyld et al. (Eds): AIBD, MLSC, ACSTY, NATP, CCCIoT, SVC, SOFE, ITCSS -2024
pp. 253-266, 2024. CS & IT - CSCP 2024 DOI: 10.5121/csit.2024.140420

- **Dialogue Policy Learning (DPL):** Based on the calibrated dialogue states from the DST module, this module decides the following action of a dialogue agent.
- **Natural Language Generation (NLG):** This module converts the selected dialogue actions into surface-level natural language, usually the ultimate response form.

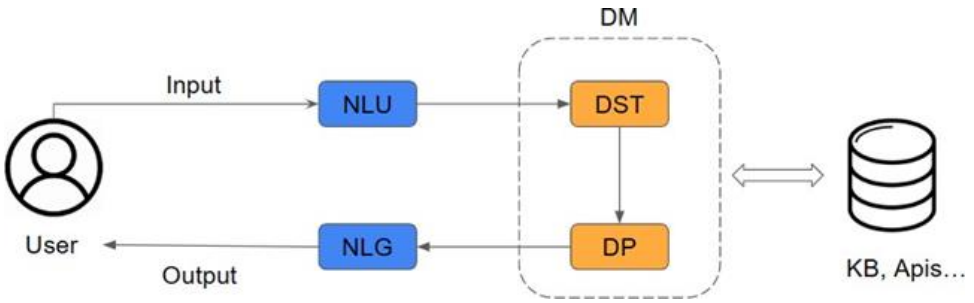


Figure. 1. Structure of a task-oriented dialogue system in the task-completion pipeline.

DST and DPL are the components of Dialogue Managers (DM) in TODS. Rule-based solutions were initially utilized but faced limitations such as domain complexity and task scalability [5]. With advancements in deep learning and the availability of labeled conversational datasets, supervised learning (SL) and reinforcement learning (RL) emerged as viable alternatives for training dialogue policies [2,6]. RL techniques have shown promise through optimizing dialogue policies via user interactions but still face challenges, such as the need for rule-based user simulators and domain-specific reward functions [3,2]. SL approaches, which involve the assignment of classified states to predefined system actions, have proven to be an excellent alternative to RL algorithms, as demonstrated in [7]; Researchers have proposed numerous models based on Transformers, GRU, LSTM, and multilayer perceptron [8,9,7,10]. However, the limited representativeness of available datasets may hinder supervised learning approaches, affecting the generalizability of learned policies and potentially requiring expensive data acquisition efforts.

While SL models are specifically designed to classify within a given range of actions, achieving optimal precision remains a complex endeavor. Our analysis suggests that one of the most influential factors affecting performance doesn't lie so much in the models themselves but in the quality of the datasets. Therefore, thedatasets for evaluating these systems must be rigorously curated, ensuring a fair and balanced comparison. The core objectives of this study are:

1. Our goal is to conduct a detailed analysis of the range of errors commonly encountered in dialogue datasets. To achieve this, we have closely examined the Multiwoz 2.1 dataset, which has been thoroughly analyzed by [11]. Their findings indicate that Multiwoz 2.1 contains various errors that negatively impact its effectiveness.
2. To improve the quality of datasets used in research, we have developed an advanced synthetic dialogue generator. This tool is designed to create datasets that are either devoid of errors or contain a controllable amount of errors. It offers the flexibility to specify the number of dialogues, user intents, entities, and actions. Additionally, it allows for the customization of dialogue events, such as transitions between topics or the inclusion of casual conversation. Crucially, it can finely tune the likelihood and types of errors introduced into the dialogues.

In this work, we first evaluate the current landscape of dialogue management research, identifying gaps and drawing comparisons with our work. We then present the construction and features of a novel synthetic dialogue generator, which allows for a controlled introduction and analysis of errors in dialogue datasets. Detailed examination of these errors helps to understand

their impact on dialogue system performance. Finally, we report on experiments that showcase the utility of our approach, followed by a discussion of the results and implications for future advancements in the field. Our findings validate that employing curated datasets via this generator enhances performance across SL models, irrespective of their architecture. Introducing errors precipitates a notable performance decline, consistently observed across models. Hence, this generator also doubles as a tool for gauging model robustness, proving its utility in evaluations.

2. RELATED WORK

In this section, we summarize the findings from the literature, outlining the focuses, methodologies, and contributions made by various studies. The following table provides a comprehensive overview of related works in the realm of dialogue system dataset analysis and improvement:

Table 1. Summary of Related Work in Dialogue Management for Chatbots

Reference	Focus	Methodology	Contributions
[11]	Quality of dialogue datasets	Evaluation of dataset quality	Identified lack of context and diversity in human conversation representation
[12]	Dialogue state tracking	Analysis and improvement on Multiwoz 2.1	Multiwoz 2.1 Dataset quality evaluation
[13]	Dialogue state tracking	Evaluation and analysis	Taskmaster-1 dataset used for quality assessment
[14]	Agent generalization	Dataset creation	Cleaner, research-oriented dataset designed for generalizing agents
[15]	Dialogue state tracking improvements	Modifications of Multiwoz 2.1	Updated slots and entities for improved tracking
[4]	Dialogue management dependency on NLU	Discussion	Highlighted the dependence of dialogue management on natural language understanding
[16]	Dialogue generation methods	Proposal of methodology	A stack of topics for dialogue generation
[17]	Handling subdialogues	Implementation of dialogue stack	RavenClaw system for precise topic tracking and sub-dialogue management
[18]	Management of non-deterministic dialogues	Use of conversational graphs	Improved dialogue management using a conversation graph
[19]	Task-oriented dialogue framework	Data flow synthesis	Dialogue state as a data flow graph, mapping user inputs to the extendable program

Limited research focuses on studying and analyzing datasets in the field of dialogue management in chatbots. However, recent works such as [11] have examined the quality of datasets used in this field. This study's authors argue that many currently available datasets need more context and adequately reflect the complexity and diversity of human conversations. The authors evaluate

the quality of these datasets using two popular datasets, multiwoz2.1 [12], and Taskmaster-1 [13]. Through a detailed analysis of these datasets, the authors identify various areas in which these datasets lack context, including history independence, solid knowledge base dependence, and ambiguous system responses.

Other datasets, such as SGD [14] and multiwoz2.4 [15], have focused on improving existing datasets to solve different tasks. SGD [14] presents a cleaner and more research-oriented dataset for agent generalization. In contrast, multiwoz2.4 modifies the multiwoz2.1 dataset regarding slots and entities to improve dialogue state tracking performance. Other studies, such as [4], suggest that the dialogue manager depends on NLU. Regarding dialogue generators, studies like [16] suggest creating a dialogue generation by following a stack of topics. Ravenclaw dialogue system [17] implemented this dialogue stack for handling sub-dialogues. However, while a stack structure effectively allows for the handling and conclusion of sub-dialogues, it can also be limiting. Ravenclaw's authors advocate for precise topic tracking to facilitate contextual interpretation of user intents. As human conversations often revisit and interleave topics, there is a need for a more flexible structure for an agent to handle dialogue.

Furthermore, one of the more flexible data structures is a graph. [18] proposes a method for improving the management of non-deterministic dialogues using a conversation graph that represents the possible responses and transitions between dialogue states. Besides, [19] proposes a novel framework for task-oriented dialogue based on data flow synthesis, which involves transforming users' linguistic inputs into executable programs that manipulate data and external services. The authors represent the dialogue state as a data flow graph. Each node is a variable or an external service, and each edge is an operation or a connection. The dialogue manager maps each user input to a program that extends this graph with new nodes and edges.

As we see in [18,19], the graph is the most powerful data structure for dialogue generation. A good representation of a dialogue is a path in the conversational graph, where the nodes represent the current intentions and slots of the dialogue, and the edges represent the possible actions that the model can take based on the current and previous states.

3. SYNTHETIC DIALOGUE GENERATOR

Our primary motivation for creating the dialogue generator was that we needed curated datasets to which we could induce controlled errors to see how the presence or absence of errors affected the performance of the models. Therefore, we needed an algorithm or procedure that would allow us, on the one hand, to generate these synthetic sets and, on the other hand, to parameterize different aspects of this data. We have ruled out using generative models precisely because we want to generate symbolic code that represents intentions, actions, and slots. Instead, we have designed a rule-based system (RBS) because it offers superior controllability for our task than generative models. Furthermore, these procedures allow us to introduce randomization mechanisms that can intentionally change the context or add errors. All these features are mod using configuration files, and this set of texts is called ontology.

3.1. Ontology

We define ontology as the information related to the set of actions, intentions, and slots required to achieve the various objectives of the dialogue satisfactorily. This ontology includes:

- **Topic**: the set of slots belonging to a single domain. The bot must fill a set of slots by asking the user or providing possible values. There are three categories for slots:

  - Mandatory: Essential slots to complete the topic. They are either actively provided by the user or requested by the dialogue management module.
  - Desired: Slots that are actively provided by the user or requested by the dialogue management module, but the task can still be completed if not filled in.
  - Optional: Unnecessary slots to complete the task. They are collected when the user provides but never explicitly requested by the dialogue management module.

- **Domain**: A set of topics the chatbot is programmed to solve and their relationships. For example, in the domain of restaurants, two topics could be finding a restaurant and ordering tickets for a concert.

Domains, topics, and slots are fully customizable. In the case of intention and actions, we create a simple map for many cases without any ambiguities.

3.2. Intentions and Actions

One of the essential aspects of the generator is the intentions and actions that represent what the user can say and the possible responses of the bot. Therefore, our approach was to make them as general as possible to cover many domains.

- **Intentions:** Intentions are predefined actions that represent the motivations behind user queries. They are categories that encompass different types of requests and help the bot generate appropriate responses.

  - INFORM INTENT: The user indicates the task he/she wants to perform (e.g., to book a restaurant). There can be more than one in the input sentence (Example: I want to make a reservation at a restaurant, and I also want to order a taxi to take us there).
  - INFORM: The user can inform the bot about the value of a single slot through the intention. The system will generate multiple.
  - AFFIRM: Positive response to a bot query.
  - NEGATE: Negative response to a bot query.
  - REQUEST: The user asks for the value of a slot (Example: What kind of food did I ask you?).
  - THANK: to show gratitude.
  - GOODBYE: for goodbyes.
  - UNK: for those entries that the NLU cannot classify.
  - CHIT CHAT: for all those entries that deviate from the domains in the dataset.

- **Actions:** Each of the possible responses of the bot to the current state of the dialogue. We can solve many scenarios with the following actions. However, there is a limit to the number of actions.

  - INFORM: to inform or offer a slot to the registered or unregistered user.
  - REQUEST: to request a mandatory slot from the user.
  - CONFIRM: Confirm that the model registered the slot.
  - NOTIFY: to notify the search status if it has succeeded.
  - REQ MORE: to request a mandatory slot from the user.
  - ANSWER CHIT CHAT: reply to the chitchat.

3.3. Rules

According to [18,19], we seek to generate a graph for each data set, where the nodes are the states of the dialogue, composed of intentions, actions, and slots, and the links are the corresponding actions. Each node will have information related to the domain and the corresponding topic. However, implementing this theoretical interpretation of a conversation graph can be challenging in practice due to the many different contexts and events that can change the path of the graph; the user can change their mind during a conversation, which can alter the course of the conversation. For instance, when ordering a pizza, the user may change their order based on their dietary preferences or decide to dine instead of placing a take-out order. We use the "stack of topics" proposed by [17] as the next level of abstraction in a dialogue. We could jump into the context, change slots, or even chit-chat in a conversation. These events are hard to implement using a raw graph; however, we design these events as topics in a stack, so on top, we process one path without knowing the complete graph is a priority. The graph emerges from following the structure of the stack. As a generator, there are randomization mechanisms that can change the context or intentionally add errors. Our generator applies the rules at the top of the pile, adapting them to the node domain and topic. The obligatory slots are the aim of all dialogue-oriented tasks, and we design all rules according to this principle:

- We have a corresponding slot for every INFORM intent, which allows the user to input information for an empty slot or modify the value of a filled slot.
- The INFORM INTENT intent will be the one that starts a dialogue and has no associated slot.
- The corresponding action for INFORM is CONFIRM.
- If any mandatory slots are missing, the action will be REQUEST.
- The NOTIFY action will occur once the dialogue fills all the slots, indicating that an external source has been searched or requested.
- The model will trigger the REQ MORE action once the user fills all the required slots.
- ANSWER CHIT CHAT will occur whenever the intent is CHIT-CHAT.
- At any time, an event can occur that changes the top of the context stack. All information is stored to continue when the dialogue returns to the top of the stack.

3.4. Events

An event is any conversation that disturbs achieving the current objective at the top of the dialogue stack. So, we could highlight three types of events:

- **Chit chat:** any conversation that departs from the defined domains of the dataset. Always come with an intention-action pair: CHIT CHAT and ANSWER CHIT CHAT.
- **Mind-changing:** when we have a slot filled with a specific value, but the user changes his mind by changing its value or leaving it empty.
- **Domain-changing:** when the user wants to complete a task in a specific domain but changes the topic or domain at any given time.

3.5. Errors

Unfortunately, errors are inherent in creating any dataset and may be due to incorrect labeling or poor transcription. When designing a dataset, we need to consider the importance of cleaning our data and checking that all samples are appropriate for the problem we want to solve. In addition, the performance of the models will be directly affected by perturbations in the dataset. This lack of performance is due to the nature of supervised learning models. If we train the algorithms on

low-quality samples, we cannot guarantee they will obtain a good generalization and correct score.

In this section, we study and analyze each of these errors in the data sets applied to TOD, which, according to [11], are very present in many of these sets, mainly in Multiwoz2.1:

- **NLU errors:** If the NLU model does not perform a good classification of the input text, the performance of the dialogue manager will be seriously affected, causing the conversation management to fail.

- **Human labeling errors:** The labeler (a person) has incorrectly labeled these samples. These errors can be a misallocation of tags to intentions, actions, or slots.

- **Limited temporal reference:** Some algorithms, such as TED, are designed to capture temporal dependencies in long conversations. The idea behind this is that the manager needs long-term context information for a dialogue manager to take the right action in a conversation. While this idea may make sense, in reality, datasets are designed intentionally or out of ignorance, with only the previous state in mind, and this is not the case in a real conversation. Humans do not make decisions based solely on the previous state. Thus, the poor temporal generalization of the datasets affects the models used in production, which need to be well-trained to handle such issues. This error is studied in depth by [11].

- **Ambiguities:** We have included this phenomenon as an error because it can cause a substantial performance drop in the models if not considered. It is an inherent ambiguity in human language. When analyzing a dataset, it is possible to find multiple actions for a given dialogue state that do not impact the overall outcome of the conversation. Conversations can take various valid and coherent paths to communicate the intended message effectively. Therefore, trained models using this data can take different actions for the same state that are correct. This one-to-many nature can confound many algorithms designed to obtain the best possible answer. A proposed solution by [20] involves creating atomic actions to expand the action space. This method combines actions with one or more different slots to simplify the problem and improve model performance. We have utilized this method to train dialogue management models for both synthetic and real data.

In this work, we focus only on NLU and mislabeling errors, as they are the most common and abundant in a dataset and can be controlled by probability. Perturbation techniques for the generator consist of choosing a random sample from the dataset, consisting of intentions, slots, and actions, and replacing its actual value with one chosen randomly from all possible ones. Another technique is to replace its actual value with a" UNK" (unknown), pretending that the labeler failed to identify the sample or the NLU model did not classify it well. We can control these error mechanisms by parameters that independently simulate the probability of this happening for actions, intentions, and slots.

4. EXPERIMENTAL SETUP

In this section, we provide an in-depth breakdown of our experimental framework, discussing our choice of datasets, models, and evaluation methodology. The code for our experiments is available in this repository.

4.1. Datasets

Real Datasets: MultiWOZ 2.1[12] is a rich dataset comprising 10,438 human-human dialogues, simulating a Wizard-of-Oz task across seven domains: hotel, restaurant, train, taxi, attraction, hospital, and police. These dialogues are essentially interactions between a user and a wizard (clerk). While the user seeks information, the wizard, backed by a comprehensive knowledge base, offers the requested details or facilitates a booking. These dialogues come annotated with labels highlighting the wizard's actions and the perceived user goal after each user interaction. For our analysis, we segregated MultiWOZ 2.1 into 7,249 training and 1,812 test dialogues, while, unfortunately, 1,377 dialogues were omitted due to incomplete annotations.The SGD [14] dataset encompasses over 20,000 annotated dialogues depicting multi-domain, task-oriented interactions between humans and virtual assistants. These dialogues span 20 domains, from banking and events to travel and weather, encompassing interactions with various services and APIs. Each domain can have multiple APIs with overlapping functionalities but distinct interfaces, mirroring real-world scenarios. This dataset is versatile, being suitable for intent prediction, slot filling, dialogue state tracking, and more. Notably, the SGD dataset contains unseen domains in the evaluation set, aiding in gauging zero-shot or few-shot performance.

Synthetic Datasets: Our syntheticdatasetswere meticulously crafted totestDPL models under different complexity levels: Simple, Medium, and Hard. The variation in complexity arises from the diversity of events, such as chit-chat and mind-changing, as well as from varying quantities of domains and slots. You can access and download these datasets using this link. Please refer to Table 2 for more details on these datasets.

- **Simple:** Contains basic interactions with minimal unexpected events.
- **Medium:** Introduces a moderate level of complexity with occasional unexpected events.
- **Hard:** Mimics real-world scenarios with a high frequency of unexpected events and intricate dialogue structures.

Table 2. Summary of datasets: The datasets vary in terms of the number of dialogues, domains, and slots, providing different levels of complexity for training and testing conversational models. The table also indicates the number of dialogues allocated for training, validation, and testing.

	Normal		**Synthetic**		
	MultiWoz2.1	**SGD**	**Simple**	**Medium**	**Hard**
Dialogues	10438	20000	2000	6000	10438
Domains	7	20	2	5	7
Slots	45	45	10	22	45
Train	8438	16000	1200	3600	8438
Val	1000	2000	400	1200	1000
Test	1000	2000	400	1200	1000

4.2. Evaluation Metrics

In dialogue management, precision indicates how many of the predicted responses or actions were relevant, while recall illustrates how many of the actual relevant responses were correctly predicted by the model. The F1 score, being the harmonic mean of precision and recall, provides a balanced measure of a model's performance, especially in situations where there's an uneven class distribution. These metrics, thus, offer a comprehensive view of how well a model performs

in real-world scenarios where both false positives and false negatives have significant implications.

- **F1 Score(F1):** A balanced measure that considers both false positives and false negatives.
- **Precision(P):** Reflects the model's capability to predict only the relevant responses, minimizing false positives.
- **Recall(R):** Highlights the model's strength in capturing all potential correct responses, minimizing false negatives.

4.3. Experimental Infrastructure

All computations were performed using an NVIDIA GeForce RTX 3090, with all models completed within 24 hours across all datasets.

4.4. Models

Our experiments incorporated some of the most referenced models in dialogue management. Their hyperparameter configurations remained consistent with the original specifications:

- **Transformer embedding dialogues (TED)** [8] uses the Star-Space algorithm developed by Facebook [21]. TED's primary goal is to enhance chatbots' performance in dialogue tasks by employing transformer-based encoders to capture temporal relations in the dialogues.
- **Recurrent embedding dialogues (RED)** [8] is the same network as TED but uses an LSTM encoder [22] rather than transformer-based encoders.
- **Planning Enhanced Dialog Policy (PEDP)** [10] improves the performance of chatbots in dialogue tasks by using a planning module to predict intermediate states and individual actions.
- **DiaMultiClass (MC)** [7] is a three-layer MLP.
- **DiaSeq (SEQ)** [7] is a two-layer perceptron to extract features from raw state representations and uses a GRU to predict the following action.
- **DiaMultiDense (MD)** [7] uses a two-layer MLP to extract state features, followed by an ensemble of dense layers, and Gumbel-Softmax [23] functions consecutively.

4.5. Models

The state representation follows the structure from [7]. The representation includes:

- Current slots
- Last user intent: This is derived directly from human annotations, ensuring consistency and accuracy.
- Last system action
- Current dialogue management state

For RED and TED, we use the standard state representation as proposed in [8]. The representation is based on a binary embedding that integrates the above information types. Lastly, we treat the bot response problem as a multi-label prediction task, allowing for combined atomic actions within a single dialogue turn. Each action merges the domain name, action type, and slot name.

4.6. Models

Table 3. Experimental results were obtained using all available datasets. That is in line with the results reported in the literature for the Multiwoz and SGD datasets

	MultiWoz(%)			SGD(%)		
Models	**F1**	**P**	**R**	**F1**	**P**	**R**
MC	39.41	54.60	34.32	73.78	77.77	71.20
MD	35.92	51.93	30.10	78.37	90.33	72.32
SEQ	44.64	51.91	43.66	**86.04**	87.69	**84.65**
RED	**69.52**	65.27	**69.52**	74.44	74.27	77.61
TED	61.98	62.28	67.46	78.33	79.65	80.25
PEDP	66.95	**78.11**	65.02	84.74	**92.07**	81.30

Table 4. Experimental results were obtained using simple, medium, and hard synthetic datasets.

	Simple (%)			Medium (%)			Hard (%)		
Models	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**
MC	85,92	91,44	84,19	86,62	92,68	84,12	85,8	91,74	83,38
SEQ	81,91	89,72	80,19	80,25	90,31	77,66	80,45	90,36	77,87
RED	**100**	**100**	**100**	**100**	**100**	**100**	**99,76**	**99,76**	**99,76**
TED	**100**	**100**	**100**	98,9	98,99	98,95	90,11	94,97	89,55
PEDP	99,98	99,99	99,98	99,55	99,45	99,71	98,67	99,03	98,52

We evaluated different models using real datasets, Multiwoz 2.1 and SGD, and we present the results in Table 3. In the Multiwoz 2.1 dataset, the RED model achieved the highest results in F1 and Recall, with both values at 69.52%. On the other hand, the PEDP model stood out for its precision, which reached a maximum value of 78.11%, suggesting that this model was particularly effective in minimizing false positive responses.

Alternatively, in the SGD dataset, the SEQ model stood out, achieving the highest F1 and Recall values, at 86.04% and 84.65%, respectively. This reflects that the SEQ model provided the best performance in terms of balance between precision and recall in this dataset. However, it was the PEDP model that achieved the highest precision, with a value of 92.07%, indicating that this model was extremely effective at generating correct positive predictions. These results vary between the two datasets, underscoring that models can perform differently depending on the characteristics of the dataset they are working with. Overall, it appears that all models performed better with the SGD dataset compared to Multiwoz2.1. In addition to evaluating the models with the real datasets Multiwoz 2.1 and SGD, we also conducted tests with synthetic data. These synthetic datasets were generated with different levels of complexity: simple, medium, and hard.
In the simple synthetic dataset, both the RED and SEQ models achieved perfection in all evaluation metrics, reaching 100% in F1, Precision, and Recall. This indicates that both models were capable of handling this dataset with high precision and completeness. On the other hand, the TED, MD, MC, and PEDP models performed less well, although all achieved a good performance. As the complexity increased with the medium synthetic dataset, the SEQ model maintained its perfect performance. The RED model experienced a slight drop in performance, although it remained high. In contrast, the other models showed a similar performance to what was observed in the simple synthetic dataset. Finally, on the hard synthetic dataset, the SEQ model consistently demonstrated exceptional performance, achieving nearly 100% in all metrics. The rest of the models showed a slight decrease in their performance compared to the less

complex synthetic datasets, indicating that the increasing difficulty of the data poses additional challenges for these models.

Continuing with the robustness tests of the models, we also explored how they behave in the presence of errors in the datasets. To do this, we gradually increased the proportion of errors in the synthetic datasets and observed its impact on the performance of the models, and the results are shown in 2. All models achieved high performance with the dataset without errors. The RED, SEQ and TED achieved perfect performance. MD, MC, and PEDP also demonstrated high performance, although slightly below than others. However, when increasing the errors to 10%, we saw that all models experienced a decrease in their performance. In particular, the TED and RED models were the most affected, with a drop in performance to 80%.

On the other hand, the SEQ model maintained the highest performance. As errors increased to 20% and 40%, the SEQ model showed the highest performance, closely followed by PEDP. RED, MD, MC, and TED continued to experience decreased performance, with TED being the most affected model. When errors reached 40% and 60%, the SEQ model showed notable robustness, maintaining its performance at 80%. On the other hand, the performance of RED and TED fell significantly. Finally, even with very high error levels of 80% and 90%, the SEQ model showed remarkable robustness with a stable performance. In contrast, the other models experienced additional decreases in their performance, showing an almost linear trend.

In conclusion, our findings suggest that the TED, RED, and SEQ models are notably robust when faced with datasets of varying complexity, maintaining high performance even on the most challenging datasets. The MD, MC, and PEDP models also demonstrated respectable performance, but they were more impacted by the increasing complexity of the datasets. Importantly, these experiments also highlight that errors in datasets can significantly impact the performance of models, a factor that is often overlooked when comparing solutions. Our results show that the SEQ model proved to be the most resilient in the face of dataset errors, closely followed by PEDP. While all models experienced a performance drop with the introduction of errors, the SEQ model showed impressive robustness, maintaining consistent performance even at high error levels. In contrast, the RED and TED models were significantly more impacted by the introduction of dataset errors. This study underscores the importance of considering dataset errors in model evaluation and comparison. Therefore, acknowledging the effects of errors in datasets is crucial for developing and deploying more reliable and efficient models.

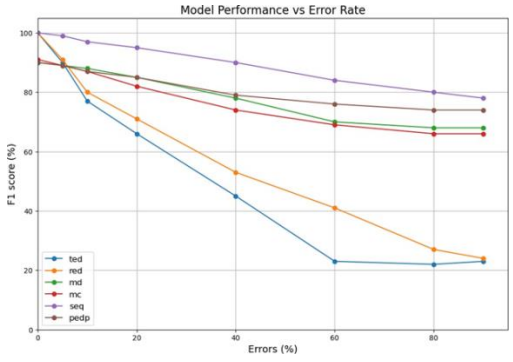


Figure 2. The ability of systems to maintain their performance in the presence of NLU or labeling errors.

5. LIMITATIONS AND FUTURE WORKS

Our study has provided valuable insights into the effects of dataset quality on the performance of TODS. However, several limitations need to be addressed in future research.

First, while synthetic datasets offer a controlled environment to study specific errors, they inevitably lack the richness and unpredictability of real human conversations. A key challenge for future work is to bridge the gap between synthetic and real-world data, perhaps by integrating the two to create more robust and nuanced training materials.

Second, our focus on dataset errors, although crucial, does not encompass all aspects that contribute to the adequate performance of dialogue systems. The interplay between error management, NLU, NLG, model architecture, and the learning algorithm complexity should be examined in greater depth. Further studies could also consider the impact of these factors on dialogue management comprehensively. Third, scalability and complexity pose significant hurdles as we strive to create dialogue systems that manage an ever-growing array of tasks across various domains. There is a need for scalable strategies to generate synthetic datasets representative of this diversity. Creating methodologies for efficiently extending dataset coverage without compromising quality will be an area of ongoing research.

Building upon our current research, the following avenues are proposed for future work:

- **Developing Hybrid Datasets:** Future research could focus on creating hybrid datasets that combine real conversation elements with synthetically generated errors. This approach could provide a middle ground that maintains the complexity of real dialogues while allowing controlled error analysis.
- **Improving NLU and NLG:** Exploring the boundaries of NLU and NLG within the context of dataset errors could yield significant improvements in dialogue system performance. This includes the enhancement of entity recognition, context understanding, and the generation of more coherent and contextually relevant responses.
- **Cross-domain and Multi-domain Studies:** Investigatingthetransferability of models trained on synthetic datasets to cross-domain and multi-domain scenarios would be valuable. This involves developing models that generalize well across different domains and adapt to new ones with minimal additional training.
- **Exploring Alternative Learning Paradigms:** Alternativestosupervised and reinforcement learning, such as semi-supervised, unsupervised, and transfer learning, should be explored for their potential to reduce dependency on large annotated datasets.
- **Integration with Large Language Models (LLMs):** As Large Language Models continue to advance, their integration into task-oriented dialogue systems to enhance natural language understanding and generation becomes feasible. Future work could investigate how pre-trained LLMs can be fine-tuned using transfer learning techniques to better capture the nuances of specific domains or tasks without requiring extensive domain-specific, labeled training data.

6. CONCLUSIONS

This work emphasizes the significance of high-quality, curated datasets for accurate model evaluation in dialogue management. We have introduced a taxonomy that categorizes the primary errors found in these datasets, highlighting the necessity for their meticulous handling. Moreover, our synthetic dataset generator has been crafted as a tool for researchers and developers to assess their dialogue management models. Using this tool, they can explore model behavior in the presence of various errors, offering deeper insights into their system's robustness and performance.

ACKNOWLEDGMENTS

ACIISI-Gobierno de Canarias and European FEDER Funds Grant EIS 2021 04 partially supported this research.

REFERENCES

[1] H. Brabra, M. Baez, B. Benatallah, W. Gaaloul, S. Bouguelia, and S. Zamanirad, "Dialogue management in conversational systems: A review of approaches, challenges, and opportuni- ties," IEEE Transactions on Cognitive and Developmental Systems, vol. 14, no. 3, pp. 783–798, 2022.

[2] J. Ni, T. Young, V. Pandelea, F. Xue, and E. Cambria, "Recent advances in deep learning based dialogue systems: a systematic survey," Artificial Intelligence Review 2022, pp. 1–101, 8 2022. [Online]. Available: https://link.springer.com/article/10.1007/s10462-022-10248-8

[3] Z. Zhang, R. Takanobu, Q. Zhu, M. Huang, and X. Zhu, "Recent advances and challenges in task-oriented dialog systems," Science China Technological Sciences, vol. 63, no. 10, pp. 2011–2027, 2020.

[4] A.-Y. Kim, H.-J. Song, S.-B. Park, and R. Zunino, "A two-step neural dialog state tracker for task-oriented dialog processing," Intell. Neuroscience, vol. 2018, jan 2018. [Online]. Available: https://doi.org/10.1155/2018/5798684

[5] J. Weizenbaum, "Eliza—a computer program for the study of natural language communication between man and machine," Commun. ACM, vol. 9, no. 1, p. 36–45, jan 1966. [Online]. Available: https://doi.org/10.1145/365153.365168

[6] Z. Zhang, X. Li, J. Gao, and E. Chen, "Budgeted policy learning for task-oriented dialogue systems," in Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Florence, Italy: Association for Computational Linguistics, Jul. 2019, pp. 3742–3751. [Online]. Available: https://aclanthology.org/P19-1364

[7] Z. Li, J. Kiseleva, and M. de Rijke, "Rethinking supervised learning and reinforcement learning in task-oriented dialogue systems," in Findings of the Association for Computational Linguistics: EMNLP 2020. Online: Association for Computational Linguistics, Nov. 2020, pp. 3537–3546. [Online]. Available: https://aclanthology.org/2020.findings-emnlp.316

[8] V. Vlasov, J. E. M. Mosig, and A. Nichol, "Dialogue transformers," 2019. [Online]. Available: https://arxiv.org/abs/1910.00486

[9] V. Vlasov, A. Drissner-Schmid, and A. Nichol, "Few-shot generalization across dialogue tasks," 2018. [Online]. Available: https://arxiv.org/abs/1811.11707

[10] S. Zhang, J. Zhao, P. Wang, Y. Li, Y. Huang, and J. Feng, "think before you speak": Improving multi-action dialog policy by planning single-action dialogs," 2022. [Online]. Available: https://arxiv.org/abs/2204.11481

[11] J. E. M. Mosig, V. Vlasov, and A. Nichol, "Where is the context? – a critique of recent dialogue datasets," 2020. [Online]. Available: https://arxiv.org/abs/2004.10473

[12] M. Eric, R. Goel, S. Paul, A. Sethi, S. Agarwal, S. Gao, A. Kumar, A. Goyal, P. Ku, andD. Hakkani-Tur, "MultiWOZ 2.1: A consolidated multi-domain dialogue dataset with state corrections and state tracking baselines," in Proceedings of the Twelfth Language Resources and Evaluation Conference. Marseille, France: European Language Resources Association, May 2020, pp. 422–428. [Online]. Available: https://aclanthology.org/2020.lrec-1.53

[13] B. Byrne, K. Krishnamoorthi, C. Sankar, A. Neelakantan, B. Goodrich, D. Duckworth,S. Yavuz, A.Dubey, K.-Y. Kim, and A. Cedilnik, "Taskmaster-1: Toward a realistic and diverse dialog dataset," in Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 4516–4525. [Online]. Available: https://aclanthology.org/D19-1459

[14] A. Rastogi, X. Zang, S. Sunkara, R. Gupta, and P. Khaitan, "Towards scalable multi-domain conversational agents: The schema-guided dialogue dataset," in Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, no. 05, 2020, pp. 8689–8696.

[15] F. Ye, J. Manotumruksa, and E. Yilmaz, "MultiWOZ 2.4: A multi-domain task-oriented dialogue dataset with essential annotation corrections to improve state tracking evaluation," in Proceedings of the 23rd Annual Meeting of the Special Interest Group on Discourse and Dialogue. Edinburgh,

UK: Association for Computational Linguistics, Sep. 2022, pp. 351–360. [Online]. Available: https://aclanthology.org/2022.sigdial-1.34

[16] B. J. Grosz and C. L. Sidner, "Attention, intentions, and the structure of discourse," Computational Linguistics, vol. 12, no. 3, pp. 175–204, 1986. [Online]. Available: https://aclanthology.org/J86-3001

[17] D. Bohus and A. I. Rudnicky, "The ravenclaw dialog management framework: Architecture and systems," Comput. Speech Lang., vol. 23, no. 3, p. 332–361, jul 2009.

[18] M. Gritta, G. Lampouras, and I. Iacobacci, "Conversation Graph: Data Augmentation, Training, and Evaluation for Non-Deterministic Dialogue Management," Transactions of the Association for Computational Linguistics, vol. 9, pp. 36–52, 02 2021. [Online]. Available: https://doi.org/10.1162/tacl a 00352

[19] J. Andreas, J. Bufe, D. Burkett, C. Chen, J. Clausman, J. Crawford, K. Crim, J. DeLoach,L. Dorner, J. Eisner, H. Fang, A. Guo, D. Hall, K. Hayes, K. Hill, D. Ho, W. Iwaszuk, S. Jha,D. Klein, J. Krishnamurthy, T. Lanman, P. Liang, C. H. Lin, I. Lintsbakh, A. McGovern,A.Nisnevich, A.Pauls, D.Petters, B.Read, D.Roth, S.Roy, J.Rusak, B.Short,D.Slomin, B. Snyder, S. Striplin, Y. Su, Z. Tellman, S. Thomson, A. Vorobev, I. Witoszko,J. Wolfe, A. Wray, Y. Zhang, and A. Zotov, "Task-oriented dialogue as dataflow synthesis," Transactions of the Association for Computational Linguistics, vol. 8, pp. 556–571, 2020. [Online]. Available: https://aclanthology.org/2020.tacl-1.36

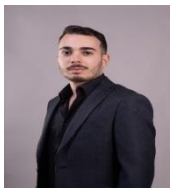
[20] S. Lee, Q. Zhu, R. Takanobu, X. Li, Y. Zhang, Z. Zhang, J. Li, B. Peng, X. Li, M. Huang, and J. Gao, "Convlab: Multi-domain end-to-end dialog system platform," 2019. [Online]. Available: https://arxiv.org/abs/1904.08637

[21] L. Wu, A. Fisch, S. Chopra, K. Adams, A. Bordes, and J. Weston, "Starspace: Embed all the things!" Proceedings of the AAAI Conference on Artificial Intelligence, vol. 32, no. 1, Apr. 2018. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/11996

[22] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural Comput., vol. 9, no. 8, p. 1735–1780, nov 1997. [Online]. Available: https://doi.org/10.1162/neco.1997.9.8.1735

[23] E. Jang, S. Gu, and B. Poole, "Categorical reparameterization with gumbel-softmax," 2016. [Online]. Available: https://arxiv.org/abs/1611.01144

AUTHORS



Miguel Angel Medina Ramirez received his degree in Computer Science from the University of Las Palmas de Gran Canaria. He pursued a Master's in Deep Learning from the University Institute of Intelligent Systems and Numeric Applications in Engineering (SIANI). He is a PhD student at the University of Las Palmas de Gran Canaria. His research focuses on dialogue systems, transformers, and NLP. Apart from his academic endeavors, Miguel Ángel is a software engineer with experience in application development and data science.

Cayetano Guerra-Arta brings 20 years of rich experience in Artificial Intelligence, with a deep focus on machine learning, neural networks, and natural language processing. He has held several positions in auditing, advising, and developing intelligent applications for various businesses and organizations.

Mario Hernandez-Tejera is a Computer Science and Artificial Intelligence Professor at the University of Las Palmas de Gran Canaria. He has over 40 years of research experience in Artificial Intelligence, with his main areas of interest being machine learning, neural networks, computer vision, natural language processing, and intelligent systems engineering. He has published over 100 papers in journals and more than 150 presentations at congresses, conferences, and symposia. He has supervised 17 doctoral theses and has been an invited speaker at different conferences and congresses. He is a member of various professional organizations.