SGD-X: A Benchmark for Robust Generalization in Schema-Guided Dialogue Systems

Harrison Lee, Raghav Gupta, Abhinav Rastogi, Yuan Cao, Bin Zhang, Yonghui Wu

Google Research
{harrisonlee,raghavgupta,abhirast,yuancao,zbin,yonghui}@google.com

Abstract

Zero/few-shot transfer to unseen services is a critical challenge in task-oriented dialogue research. The Schema-Guided Dialogue (SGD) dataset introduced a paradigm for enabling models to support any service in zero-shot through *schemas*, which describe service APIs to models in natural language. We explore the robustness of dialogue systems to linguistic variations in schemas by designing SGD-X - a benchmark extending SGD with semantically similar yet stylistically diverse variants for every schema. We observe that two top state tracking models fail to generalize well across schema variants, measured by joint goal accuracy and a novel metric for measuring schema sensitivity. Additionally, we present a simple model-agnostic data augmentation method to improve schema robustness.

Introduction

Task-oriented dialogue systems have begun changing how we interact with technology, from personal assistants to customer support. One obstacle preventing their ubiquity is the resources and expertise needed for their development. Traditional approaches operate on a fixed ontology (Henderson, Thomson, and Young 2014; Mrkšić et al. 2017), which is not suited for a dynamic environment. For every new service that arises or modification to an existing service, training data must be re-collected and systems re-trained.

The Schema-Guided Dialogue paradigm, introduced in Rastogi et al. (2020b), advocates for the creation of a universal dialogue system which can interface with any service, without service or domain-specific optimization. Each service is represented by a *schema*, which enumerates the slots and intents of the service and describes their functionality in natural language (see Figure 1). Schema-guided systems interpret conversations, execute API calls, and respond to users based on the schemas provided to it. In theory, this enables a single system to support any service; in practice, whether this is feasible hinges on how robustly models can generalize beyond services seen during training.

In the Schema-Guided Dialogue challenge at DSTC8 (Rastogi et al. 2020a), participants developed schema-guided dialogue state tracking models, which were evaluated on both seen and unseen services. While results were promising, with the top team achieving 87% *joint goal accuracy* (92% seen, 85% unseen), we observed a major shortcoming with SGD - the dataset's schemas are unrealistically uniform compared to the diverse writing styles encountered "in the wild", where schemas are written by API developers of various backgrounds.

The uniformity of SGD is evident in its schema element names. Of the names in the test set schemas "unseen" in the train set, 71% of intent names and 65% of slot names exactly match names appearing in the train schemas, meaning most names in "unseen" schemas are actually already seen by the model during training. MultiWOZ (Budzianowski et al. 2018), another popular dialogue state tracking benchmark, faces similar issues in the zero-shot leave-one-domain-out setup (Wu et al. 2019), with 60-100% of slot names in the held-out domain seen by the model during training. SGD descriptions are also uniformly written. For example, all descriptions for boolean slots either begin with the phrase "Boolean flag..." or "Whether...".

We hypothesize that the uniformity of SGD schemas allows models to overfit on specific linguistic styles without penalty in evaluation, leading to an overestimate of the generalizability of models. Additionally, "seen" schemas in evaluation are identical to the ones seen in training, meaning SGD does not evaluate how well models handle changes in seen schemas, however minor.

In this work, we investigate the robustness of schema-guided models to linguistic styles of schemas. Our contributions are as follows:

- We introduce SGD-X, an extension to the SGD dataset that contains crowdsourced stylistic variants for every schema in the original dataset[1]
- Based on SGD-X, we propose *schema sensitivity* - a metric to evaluate model sensitivity to schema variations
- We show that two top schema-guided dialogue state tracking (DST) models based on BERT and T5 are highly sensitive to schema variations, dropping 12-18% in joint goal accuracy for the average SGD-X variant

*Equal contribution from first two authors

[1]We release SGD-X and an evaluation script for schema-guided dialogue state tracking models on GitHub at https://github.com/google-research-datasets/dstc8-schema-guided-dialogue

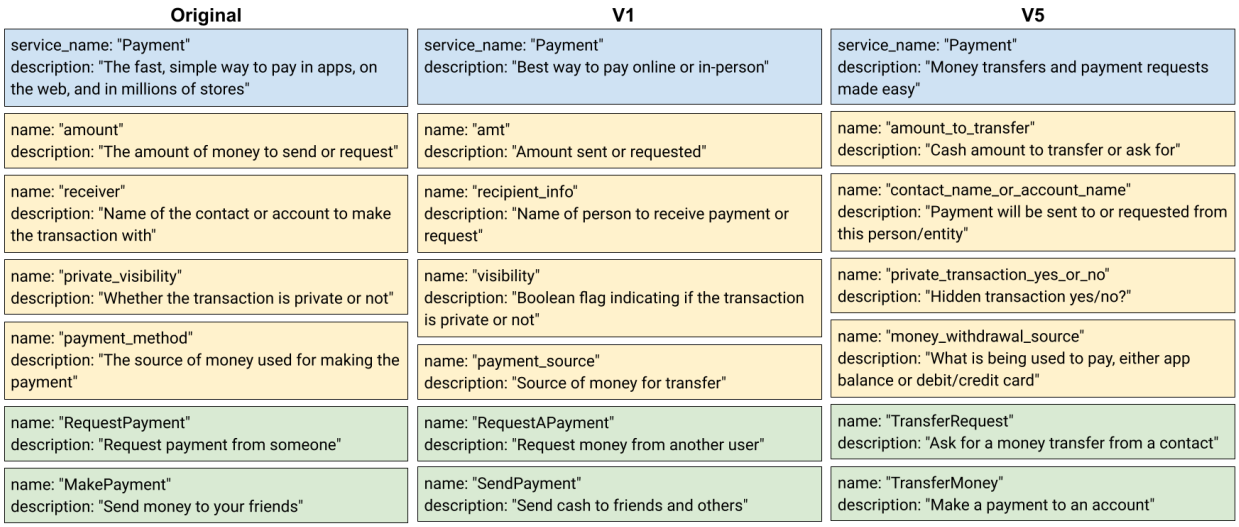

Figure 1: The original schema for a Payment service (left) alongside its closest and farthest SGD-X variants (center and right, respectively), as measured by linguistic distance functions. We study the robustness of models to writing styles used in schemas.

- We demonstrate that back-translation is an effective, model-agnostic technique for improving schema robustness

The SGD-X Dataset

We curate SGD-X, short for *Schema Guided Dialogue - eXtended*, to evaluate the robustness of schema-guided dialogue models to schemas. Following SGD terminology, we define a *schema* as a collection of intents and slots belonging to a service, along with metadata that describe their intended behavior. We also define a *schema element* as an intent, slot, or service identifier. A key feature of schemas is the inclusion of natural language descriptions for each schema element. For example, an intent "*SearchMap*" might have the description "*Search for a location of interest on the map*".

For every schema in SGD, SGD-X provides 5 variants, where each one replaces the original schema element names and descriptions with semantically similar paraphrases. Figure 1 shows an original schema alongside two SGD-X variants. We describe the dataset in detail below.

"Blind" Paraphrase Collection

Schema element names and descriptions in the original SGD dataset were written by a small set of authors, and achieving linguistic diversity was not an explicit goal. To diversify SGD-X, we crowdsourced paraphrases across 400+ authors from Amazon Mechanical Turk. We chose crowdsourcing over automatic paraphrasing methods because we found that automatic methods were often semantically inaccurate and provided insufficient linguistic diversity, especially when the text was short. We designed two crowdsourcing tasks (pictured in the Appendix of the ArXiv version[2] of this paper):

Paraphrasing names: To paraphrase names, we provided a schema element's long-form description from the SGD dataset and asked crowdworkers to generate a short name that would capture the description. We deliberately did not share the original names to encourage a diversity of paraphrases - hence "blind" paraphrasing.

Paraphrasing descriptions: To generate descriptions, we reversed the name paraphrasing task - i.e. given only the name of a schema element, we asked crowdworkers to come up with a long-form description. For a limited set of schema elements, we provided additional information:

- If intent and slot names were ambiguous on their own (e.g. the "*intent*" slot from the Homes service, which indicates whether a user is interested in buying or renting property), the original description was shown
- For categorical slots, their possible values were shown

For a single task, a crowdworker was tasked to come up with either all names or all descriptions for a given service's schema elements.

After collecting raw responses, we deduplicated and manually vetted responses for quality and correctness. Our primary criterion was whether a response accurately described the schema element, and sometimes valid responses did not fully overlap semantically with the original as traditional paraphrasing typically requires. For instance, we considered `SearchByLocation` a valid replacement for `Find`**`Home`**`ByArea`, despite the former's lack of reference to the "home" concept, since it is implied that the search is for homes in the broader context of the `Homes` service.

We created enough tasks to collect approximately 10 paraphrases per schema element name and description. At the end of the collection and vetting phase, we had at least 5 paraphrases for every name and description. When there were more than 5, we selected 5 at random.

[2]https://arxiv.org/abs/2110.06800

Composing Schema Variants

We composed our schema element paraphrases into schema variants, where each variant replaces every name and description in the original schema with a crowdsourced paraphrase. We placed paraphrases into schema variants such that variants increasingly diverge from the original schemas as the variant number increases. We sorted each schema element's name/description paraphrases by their distance from the original name/description using the following metrics:

- For names, we used Levenshtein distance
- For descriptions, we used Jaccard distance, where stopwords were removed and words were lemmatized using spaCy (Honnibal et al. 2020)

After sorting, for every schema element $elem$, we obtained a list of unique name paraphrases $N^{elem} = [n^{elem}_{idx}], idx \in \{1..5\}$, ordered by increasing Levenshtein distance from the original name n^{elem}_{gt}. Similarly for every schema element description, we obtained a list of unique description paraphrases $D^{elem} = [d^{elem}_{idx}], idx \in \{1..5\}$, ordered by increasing Jaccard distance from the original description d^{elem}_{gt}.

Finally to compose the idx schema variant, for every $elem$ in the schema, we simply selected n^{elem}_{idx} and d^{elem}_{idx}. This establishes the SGD-X benchmark as a series of increasingly challenging evaluation sets. Henceforth in this paper, we refer to these schema variants as v_1 through v_5, where v_1 refers to the variant schema closest to the original and v_5 the farthest. Figure 1 compares an original schema with its first and fifth variant to highlight the increasing divergence property.

Dataset Statistics

The original SGD dataset contains 45 schemas with a total of 365 slots and 88 intents. Each schema element is associated with 1 name and 1 description (though service names were not paraphrased). After compiling paraphrases into variant schemas, SGD-X presents 5 variants for every schema, totalling 4,755 paraphrases. Each schema variant is composed of paraphrases from multiple crowdworkers. Designing the tasks, collecting data, manually vetting responses, and composing the variants took approximately 1 month.

Table 1 presents various metrics on SGD-X. As mentioned in Section , one concern with the original test set is that roughly 70% of the slot and intent names in the 15 "unseen" schemas appear in training schemas. In contrast, that figure drops to 8% for slot names and 2% for intent names for the average SGD-X variant.

For names, the average normalized Levenshtein distance from original to paraphrase is about 0.5 (on a scale of 0 to 1), indicating high variation. For descriptions, the average BLEU score between original and paraphrase is 7.9, and the average BLEU score among paraphrased descriptions (i.e. self-BLEU[3]) is 4.5, indicating a large diversity of descriptions.

[3]We calculate self-BLEU for a description by calculating the BLEU score between every pair of variants, resulting in 5 * 4 = 20 scores. We then compute top-line self-BLEU by averaging these scores across all descriptions across all 45 unique schemas.

		Schema variant					
Metric	Orig	v1	v2	v3	v4	v5	**Avg**
% of test slot names seen in train	65%	13%	14%	5%	6%	2%	**8%**
% of test intent names seen in train	71%	0%	0%	4%	0%	4%	**2%**
Levenshtein Distance (names)	-	0.30	0.42	0.49	0.56	0.61	**0.48**
BLEU (desc)	-	18.8	11.3	5.6	2.9	1.0	**7.9**

Table 1: SGD-X dataset statistics. The metrics show high linguistic variation from the original SGD schemas.

Evaluation Methodology

We propose evaluating models by training them on original SGD only and evaluating on SGD-X. In addition to standard accuracy metrics, we propose measuring the consistency of predictions across variants. Below, we first describe our schema sensitivity metric, followed by a general proposal for training and evaluating dialogue systems on SGD-X, and finally a detailed proposal for evaluating dialogue state tracking models specifically.

Schema Sensitivity Metric

Let $\mathcal{M}$ be a turn-level evaluation metric, which takes a prediction and ground truth at turn t as input and returns a score. Let K denote the number of schema variants, p^k_t denote turn-level predictions for variant k, and g_t denote the ground-truth. We define *schema sensitivity* (SS) for the metric $\mathcal{M}$ as the turn-level Coefficient of Variation (CoV) of the metric value (i.e., the standard deviation normalized by the mean) averaged over all turns in the evaluation set. This is described by the following set of equations:

$$SS_{\mathcal{M}} = \frac{1}{|T|}\sum_{t\in T} CoV_t = \frac{1}{|T|}\sum_{t\in T}\frac{s_t}{\bar{x}_t} \tag{1}$$

where the standard deviation s_t and mean $\bar{x}_t$ are defined as follows:

$$s_t = \sqrt{\frac{\sum_{k=1}^{K}(\mathcal{M}(p^k_t, g_t) - \overline{\mathcal{M}}(\mathbf{p_t}, g_t))^2}{K-1}} \tag{2}$$

$$\bar{x}_t = \overline{\mathcal{M}}(\mathbf{p_t}, g_t) \tag{3}$$

$\overline{\mathcal{M}}(\mathbf{p_t}, g_t) = \frac{1}{K}\sum_{k=1}^{K}\mathcal{M}(p^k_t, g_t)$ is the average of the metric corresponding to predictions over all K variants in turn t, and T is the set of all turns in the eval set.

Intuitively, schema sensitivity quantifies how much predictions fluctuate when exposed to schema variants, independent of the prediction correctness, and models with lower SS are more robust to schema changes. SS may be computed for any turn-level or dialogue-level metric across the schema-guided dialogue modeling pipeline.

Metric design considerations: We chose Coefficient of Variation (CoV) over standard deviation to represent variability since normalizing by the mean allows for comparison

of variability across dialogue modeling components such as DST and NLG, as well as between two models with differing absolute performance.

For the standard deviation used in the numerator of CoV, we employ the sample standard deviation because we view the K variants as a sample of the total population of possible ways a schema could be written. Using the sample standard deviation instead of the population standard deviation reduces bias of the estimate of the true variability.

Finally, by computing the average turn-level CoV instead of computing CoV on the dataset's top-line performance, we increase the metric's sensitivity to changes in prediction stability. Designing SS as the average turn-level CoV also provides us with a sense of how much a model's predictions can be expected to fluctuate at each given turn depending on how the schema is written.

General Evaluation on SGD-X

In order to evaluate on SGD-X, we propose the following steps:

1. Train models on the original SGD train set schemas
2. Make predictions on the evaluation set using the 5 SGD-X schemas
3. Finally, measure performance on two classes of metrics:
   (a) An average of standard performance metrics over the 5 variants
   (b) Schema sensitivity metrics corresponding to the standard performance metrics

Using this training and evaluation setup best measures a model's ability to generalize to schemas written by a diverse set of authors.

Dialogue State Tracking on SGD-X

Because schema-guided dialogue state tracking (DST) is relatively well-studied, we apply the recommendations from section and outline the evaluation procedure on SGD-X. We propose scoring DST models on 2 metrics: Average Joint Goal Accuracy ($JGA_{v_{1-5}}$) and Schema Sensitivity of JGA (SS_{JGA}).

We first compute model predictions across each of the $|T|$ dialogue turns in the eval set $|K|$ times - once for each of the schema variants - for a total of $|T| * |K|$ predictions.

We compute the average turn-level JGA as follows:

$$JGA_{v_{1-5}} = \frac{\sum_{t=1}^{T} \sum_{k=1}^{K} JGA(p_t^k, g_t)}{|T| * |K|} \quad (4)$$

Next, schema sensitivity SS of the JGA is calculated following Equation (1).

Note: in this evaluation, we only use predictions on the SGD-X variant schemas and not the original SGD schemas to avoid models "cheating" by overfitting on the original schemas' writing styles.

We expect that $JGA_{v_{1-5}}$ will typically be the primary metric and SS_{JGA} an auxiliary metric. The precise tradeoff between the two metrics when evaluating candidate models will depend on the context in which the model will be applied (e.g. how do we value higher accuracy vs. prediction consistency?). In the next section, we apply this evaluation on two DST models.

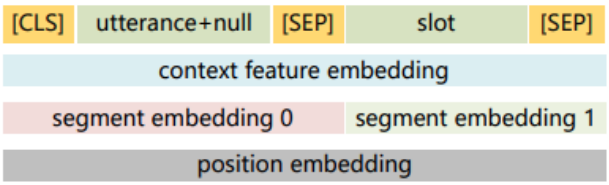

Figure 2: Input to one of the four sub-models of SGP-DST responsible for free-form slot value prediction. The last 2 dialogue utterances, a "null" token, and the slot description are concatenated (green), and the context feature takes on a value based on the slot's presence in the dialogue prior to this turn. After encoding, a slot value is predicted by selecting a span from the user utterance. Figure borrowed from Ruan et al. (2020).

Experiments

Given schema-guided modeling for DST is relatively well studied, we use SGD-X to conduct two classes of robustness experiments:

1. We train models on *original SGD* and evaluate on *SGD-X*
2. We experiment with data augmentation techniques to improve performance on SGD-X

We use the following models for our experiments:

- **SGP-DST**[4] (Ruan et al. 2020) - the highest-performing model with publicly available code, at the time of writing. 4 sub-models are trained from independent BERT-Base encoders, each specializing in a sub-task. Each one takes the dialogue and relevant schema element names/descriptions as input and makes predictions, which are then combined across the 4 models using rules. Figure 2 illustrates one sub-model.
- **T5DST** (Lee, Cheng, and Ostendorf 2021) - a generative model trained by fine-tuning T5-Base (Raffel et al. 2020) to predict slot values given the dialogue context, service, slot name, and slot description, which achieves SOTA results on MultiWOZ 2.2. Figure 3 depicts the model input and output.

Train on SGD, Evaluate on SGD-X

We trained both models on the original SGD training set with the settings that produce their reported results, and then evaluated them on the SGD-X test sets. More training details in the Appendix, available in the ArXiv version[2] of this paper.

Results: Table 2 shows the summarized results and Figure 4 displays JGA by variant. Both models see significant

[4]While the authors of SGP-DST report 72.2% JGA on the original SGD test set, we were only able to reproduce 60.5% JGA when training with the recommended hyperparameters.

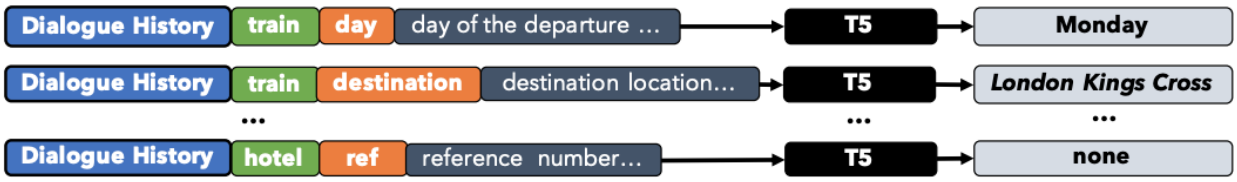

Figure 3: Example inputs and outputs for fine-tuning the T5DST model. The model is run once for each slot. The dialogue history (blue), service (green), slot name (orange), and slot description (dark gray) are input to the model, and the predicted value is decoded. Figure borrowed from Lee, Cheng, and Ostendorf (2021).

Model	Eval set	JGA_{Orig}	$JGA_{v_{1-5}}$	$Diff_{rel}$	SS_{JGA}
SGP-DST	all	60.5	49.9	**-17.6**	51.9
	seen	80.1	60.7	**-24.3**	51.5
	unseen	54.0	46.3	**-14.3**	52.0
T5DST	all	72.6	64.0	**-11.9**	40.4
	seen	89.7	79.3	**-11.6**	31.9
	unseen	66.9	58.9	**-12.0**	43.3

Table 2: Evaluation of two top-performing DST models on the SGD-X test set. Both models experience substantial declines in performance when exposed to variant schemas.

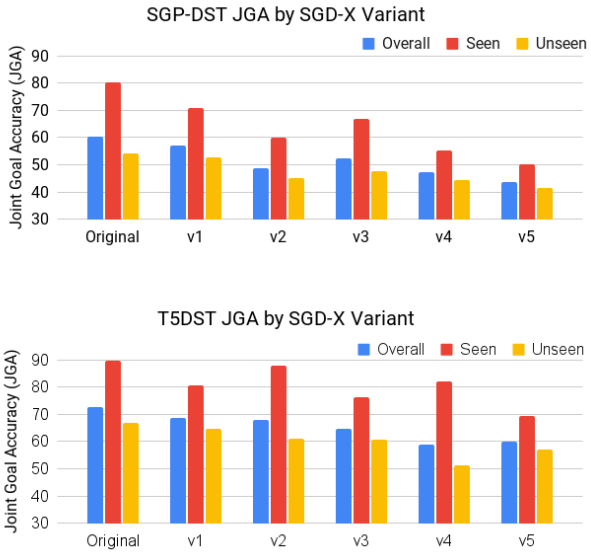

Figure 4: JGA achieved by SGP-DST and T5DST respectively on the test set for the original SGD dataset and the five SGD-X variants. Both models fail to generalize well to variants of the original schemas.

drops in joint goal accuracy, with SGP-DST and T5DST declining -17.6% and -11.9% respectively on average. For both models, the decline in JGA tends to increase in magnitude as the distance from the original schemas (reflected by the variant number) increases, with the two models dropping as much as -28% and -19% respectively for their worst variants. These results reveal that evaluating solely on the original SGD dataset overestimates the generalization capability of schema-guided DST models.

For SGP-DST, the JGA drop is much greater for seen services than unseen services. Recall that in this setup, "seen" schemas at evaluation time are no longer linguistically identical to the schemas the models were trained on. The sharp decline suggests that SGP-DST likely overfit to the exact language used in seen schemas. Performance on unseen schemas also declines for both models, which we hypothesize is due to overfitting on the linguistic styles in the original SGD dataset, as mentioned in Section .

On schema sensitivity, T5DST scores almost 12 points lower than SGP-DST in addition to achieving higher $JGA_{v_{1-5}}$, indicating it is superior to SGP-DST in both dimensions.

We observe that both models face robustness issues despite having powerful pre-trained language models as their base encoders, which have demonstrated immense success when applied to a variety of natural language tasks. We hypothesize that the models lose some of their generalization capabilities during the fine-tuning stage, a phenomenon also observed in other settings (Jiang et al. 2020).

Schema Augmentation

The results in Section suggest both models overfit on the training schemas, reducing their ability to generalize to new linguistic styles. We experiment with back-translating schemas (Sennrich, Haddow, and Birch 2016) to augment the training data (Hou et al. 2018; Yoo, Shin, and Lee 2019) and study its impact on model robustness. In addition, to establish an approximate upper-bound for how much improvement paraphrasing-based schema augmentation can provide, we also evaluate the impact of augmenting the SGD-X crowdworker-collected paraphrases.

Back-translation: For each training schema, we back-translate its schema element names and descriptions three times using Google Translate to create three alternate schemas: one each for Mandarin, Korean, and Japanese - chosen for their relatively high difficulty and consequent diversity of back-translated paraphrases. The average normalized Levenshtein distance for names and BLEU score for descriptions between the originals and their back-translations are 0.14 and 34.1 respectively. Self-BLEU among back-translated variant schemas is 41.8. These metrics indicate a moderate degree of linguistic deviation from the original

Model	Aug method	$JGA_{v_{1-5}}$	SS_{JGA}
SGP-DST	None	49.9	51.9
	Backtrans	54.1 (+8%)	43.1 (-17%)
	Oracle	66.2 (+33%)	22.5 (-57%)
T5DST	None	64.0	40.4
	Backtrans	70.8 (+11%)	34.0 (-16%)
	Oracle	73.3 (+15%)	24.6 (-39%)

Table 3: Results for schema augmentation methods on SGP-DST and T5DST models. Back-translation improves robustness for both models. Oracle augmentation, which involves augmenting SGD-X variant schemas, serves as an approximate upper bound for paraphrasing-based augmentation methods.

schemas and intra-variant diversity, though still much less than the SGD-X variants, which average 0.48 Levenshtein distance, 7.9 BLEU, and 4.5 self-BLEU. Examples pictured in the ArXiv version[2] of this paper.

Once these variant schemas were created, new training examples were generated using the same dialogues as the original training set, but with schema inputs drawn from the variant schemas. When training on the augmented dataset, models encounter the same dialogue multiple times in a given epoch, where schema element names and descriptions differ for each version.

SGD-X Crowdsourced Paraphrases (Oracle): During crowdsourcing, we collected paraphrases for all 45 schemas across train, dev, and test sets. Similarly to the back-translation experiment, for this experiment we use the crowdsourced v_1 through v_5 training set schemas to augment the training data. Note that this approach should be seen as an oracle for paraphrasing-based schema augmentation since this involves collecting roughly 5K human paraphrases for schema element names/descriptions. Furthermore, for a given variant v_i, the schema is the same for a service across train and eval sets. This means that models have already been exposed to the exact language used in seen schemas during training, giving them an unfair advantage on those services during evaluation.

Results: We train the SGP-DST and T5DST models using the two aforementioned schema augmentation approaches and evaluate on the SGD-X benchmark (without augmentation). The results are summarized in Table 3 and Figure 5.

Training with back-translated schemas improves the robustness of both models. Accuracy on SGD-X increases by +8% for SGP-DST and +11% for T5DST (relative), and it decreases schema sensitivity -17% and -16%, respectively. The improvement is considerable for unseen as well as seen schemas, suggesting that training with diverse schemas improves model generalization. This result is consistent with Wei et al. (2021), which hypothesizes that increasing diversity of training data improves performance on unseen tasks. The oracle method further improves joint goal accuracy and schema sensitivity beyond back-translation - a useful reference for how much paraphrasing-based schema augmentation may improve performance.

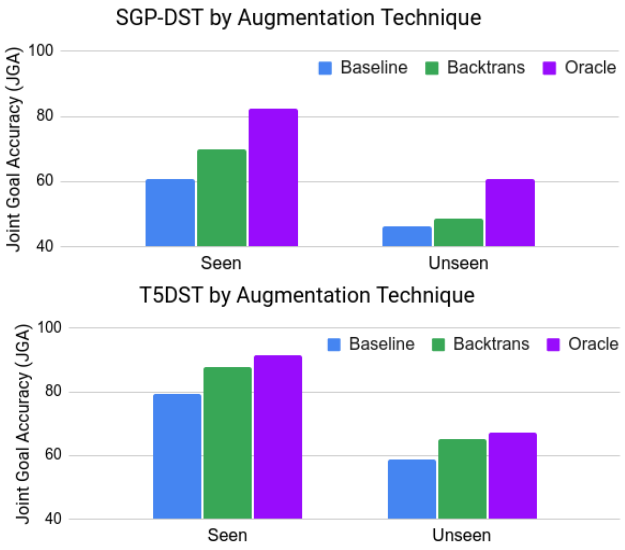

Figure 5: $JGA_{v_{1-5}}$ for the SGP-DST and T5DST models with different schema augmentation methods, split by seen and unseen services. Back-translation improves performance across the board.

Although the models trained with back-translation do not achieve parity with performance on the original SGD test set (54.1% vs. 60.1% JGA for SGP-DST, 70.8% vs. 72.6% for T5DST), much of the decline is recouped. Not only is this technique effective, but it is easy to implement, model agnostic, and requires requires no changes to modeling code.

Given that back-translating schemas with Mandarin, Korean, and Japanese already produces a relatively high BLEU score of 34.1 despite being tough to translate, we hypothesize that incorporating additional back-translated schemas from other languages would not greatly increase the diversity of linguistic styles. As a result, we believe that simply scaling to more languages would yield limited improvements in performance. One alternative to further increase linguistic diversity would be to introduce sampling when decoding for back-translation.

Other augmentation methods: Besides back-translation, we also experimented with augmenting corrupted versions of schemas, where we randomly replaced words and perturbed word order. However, we did not see improvements over the non-augmented models, which we hypothesize is due to a mismatch between the corrupted training schemas and real test schemas. Besides augmenting schemas, augmenting dialogues has shown promise in other settings and could also improve robustness (Ma et al. 2019; Noroozi et al. 2020).

Analysis

To gain better intuition of model robustness issues, we inspect cases where T5DST predicts incorrectly when given variant schemas. We also analyze T5DST's performance broken down by service. All analysis is done on T5DST trained only on the original SGD schemas.

Service	Dialogue	Slot Name and Description	Predicted Value
Weather (seen)	USER: What will the weather in Portland be on the 14th?	O: city - Name of the city	**Portland**
		v_1: city name - Name of place	***None***
Payment (unseen)	USER: I need to make a payment from my visa.	O: payment method - The source of money used for making the payment	**credit card**
		v_5: money withdrawal source - What is being used to pay, either app balance or debit/credit card	**app balance**

Table 4: Examples where T5DST fails to predict slots correctly when given SGD-X variant schemas. O represents the original, and v_i represents the i-th SGD-X schema.

Visually Inspecting Errors

We visually inspected examples where T5DST fails to predict slots correctly when provided with variant schemas. Many errors arise from failing to predict slots as active. For example, in the Weather dialogue in Table 4, the model correctly predicts "city = Portland" when given the original schema but mis-predicts "city name = None" for the v_1 variant. In these cases, the model may not understand the slot name and description well, possibly leading it believe the slot is irrelevant for the current dialogue.

We also observe cases where the model correctly predicts a categorical slot as active but predicts an incorrect value. For example, in the Payment dialogue in Table 4, the model predicts that the slot for "money withdrawal source" is "app balance" instead of "credit card" when given the v_5 schema. One hypothesis is that the word "withdrawal" in the name "money withdrawal source" biases the model to decode "balance" over "credit card", since "balance" and "withdrawal" are words present in the Banks schema seen at training time.

While SGD's original schemas and SGD-X variant schemas are semantically similar from a human's perspective, these slight perturbations have an outsized impact on model performance, highlighting the degree to which models overfit on the writing styles of schemas.

Service-level Results

In order to dissect model performance further, we plot the Average JGA ($JGA_{v_{1-5}}$) and Schema Sensitivity to JGA (SS_{JGA}) by service, shown in Figure 6. We observe that higher $JGA_{v_{1-5}}$ tends to correspond to lower SS_{JGA}. This suggests that higher accuracy prediction stability come hand in hand, for both seen and unseen services.

Given how SS_{JGA} is defined, for a given service, a model could predict the dialogue state inaccurately yet also achieve a desirably low schema sensitivity. However, our results suggest that this is atypical, with `Flights_4` being one of the exceptions to this pattern. We hypothesize that `Flights_4` breaks this pattern because it is exceptionally challenging to predict its state, leading the model to make uniformly poor predictions regardless of which schema variant it is given.

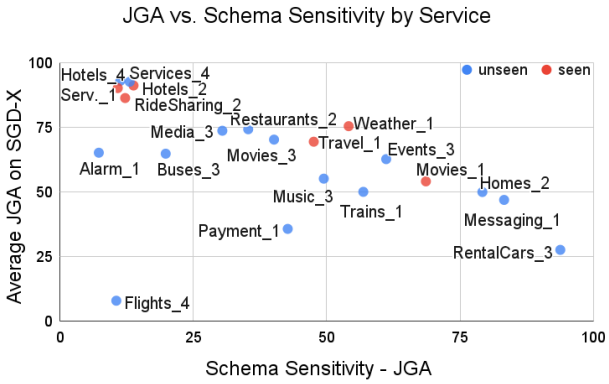

Figure 6: A plot of Average Joint Goal Accuracy and Schema Sensitivity on the test set for the T5DST model trained only on original SGD. Each point represents one service. The model tends to be less sensitive to schema variations for services it predicts more accurately.

Related Work

Schema-guided modeling aims to build task-oriented dialogue systems that can generalize easily to new verticals using very little extra information, including for slot filling (Bapna et al. 2017; Shah et al. 2019; Liu et al. 2020) and dialogue state tracking (Li et al. 2021; Campagna et al. 2020; Kumar et al. 2020) among other tasks. More recent work has adopted the schema-guided paradigm (Ma et al. 2019; Li, Xiong, and Cao 2020; Zhang et al. 2021) and even extended the paradigm in functionality (Mosig, Mehri, and Kober 2020; Mehri and Eskenazi 2021).

Model robustness is an active area of NLP research (Goel et al. 2021) and has many interpretations, such as to noise (Belinkov and Bisk 2018), distribution shift (Hendrycks et al. 2020) and adversarial input (Jia and Liang 2017).

As they are inherently public-facing in nature, the robustness of dialogue systems to harmful inputs (Dinan et al. 2019; Cheng, Wei, and Hsieh 2019) and input noise (Einolghozati et al. 2019; Liu et al. 2020), such as ASR error, misspellings, and user input paraphrasing have been explored. However, robustness to API schemas for schema-guided dialogue systems remains relatively unexplored.

Lin et al. (2021) and Cao and Zhang (2021) both investigate natural language description styles for zero/few-shot dialogue state tracking. The former experiments with homogeneously training and evaluating on different description styles, unlike our work. The latter performs heterogeneous evaluation of template-based description styles (e.g. rephrasing slot name as a question, using the original description). Models are also evaluated against paraphrased descriptions created via back-translation but only decline slightly in performance.

Conclusion

In this work, we present SGD-X, a benchmark dataset for evaluating the robustness of schema-guided models to schema writing styles. To evaluate robustness, we propose training models on SGD, predicting on SGD-X, and finally measuring standard performance metrics alongside a novel *schema sensitivity* metric that quantifies the stability of model predictions across variants.

Applying this to two of the highest-performing schema-guided DST models, we discover that both perform substantially worse on SGD-X than SGD, suggesting that evaluating solely on SGD overestimates models' ability to generalize to real-world schemas. It's noteworthy that we witness this decline on models based on T5 and BERT - two popular large language models in research and production. We further demonstrate that back-translating schemas for training data augmentation is an effective, model-agnostic technique for recovering some of this decline while simultaneously reducing schema sensitivity.

We note that the weaknesses of evaluating only on the original SGD dataset uncovered in this work also apply to the leave-one-domain-out zero-shot evaluation on the popular MultiWOZ dataset. Also, while dialogue state tracking is the focal point of this work, SGD-X is applicable to evaluating the robustness of other schema-guided dialogue components (e.g. policy, NLG). We hope that releasing this paper and benchmark motivates further research in the area of schema robustness.

Ethical Impact

Crowdsourcing details: We hired 400+ Amazon Mechanical Turk crowdworkers from the U.S. and paid USD $1-2 per task, where each task consisted of paraphrasing either names or descriptions for every element in a single schema. The median submission time was 3 minutes, which equates to US$20-40/hr. In total, we spent ∼$2000 on data collection.

References

Bapna, A.; Tur, G.; Hakkani-Tur, D.; and Heck, L. 2017. Towards zero-shot frame semantic parsing for domain scaling. *arXiv preprint arXiv:1707.02363*.

Belinkov, Y.; and Bisk, Y. 2018. Synthetic and Natural Noise Both Break Neural Machine Translation. In *International Conference on Learning Representations*.

Budzianowski, P.; Wen, T.-H.; Tseng, B.-H.; Casanueva, I.; Ultes, S.; Ramadan, O.; and Gašić, M. 2018. MultiWOZ– A Large-Scale Multi-Domain Wizard-of-Oz Dataset for Task-Oriented Dialogue Modelling. *arXiv preprint arXiv:1810.00278*.

Campagna, G.; Foryciarz, A.; Moradshahi, M.; and Lam, M. 2020. Zero-Shot Transfer Learning with Synthesized Data for Multi-Domain Dialogue State Tracking. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 122–132.

Cao, J.; and Zhang, Y. 2021. A Comparative Study on Schema-Guided Dialogue State Tracking. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 782–796.

Cheng, M.; Wei, W.; and Hsieh, C.-J. 2019. Evaluating and enhancing the robustness of dialogue systems: A case study on a negotiation agent. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 3325–3335.

Dinan, E.; Humeau, S.; Chintagunta, B.; and Weston, J. 2019. Build it Break it Fix it for Dialogue Safety: Robustness from Adversarial Human Attack. 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)*, 4537–4546.

Einolghozati, A.; Gupta, S.; Mohit, M.; and Shah, R. 2019. Improving robustness of task oriented dialog systems. *arXiv preprint arXiv:1911.05153*.

Goel, K.; Rajani, N.; Vig, J.; Taschdjian, Z.; Bansal, M.; and Ré, C. 2021. Robustness Gym: Unifying the NLP Evaluation Landscape. *NAACL-HLT 2021*, 42.

Henderson, M.; Thomson, B.; and Young, S. 2014. Word-based dialog state tracking with recurrent neural networks. In *Proceedings of the 15th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*, 292–299.

Hendrycks, D.; Basart, S.; Mu, N.; Kadavath, S.; Wang, F.; Dorundo, E.; Desai, R.; Zhu, T.; Parajuli, S.; Guo, M.; et al. 2020. The many faces of robustness: A critical analysis of out-of-distribution generalization. *arXiv preprint arXiv:2006.16241*.

Honnibal, M.; Montani, I.; Van Landeghem, S.; and Boyd, A. 2020. spaCy: Industrial-strength Natural Language Processing in Python. https://doi.org/10.5281/zenodo.1212303. Accessed: 2022-03-28.

Hou, Y.; Liu, Y.; Che, W.; and Liu, T. 2018. Sequence-to-Sequence Data Augmentation for Dialogue Language Understanding. In *Proceedings of the 27th International Conference on Computational Linguistics*, 1234–1245.

Jia, R.; and Liang, P. 2017. Adversarial Examples for Evaluating Reading Comprehension Systems. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2021–2031.

Jiang, H.; He, P.; Chen, W.; Liu, X.; Gao, J.; and Zhao, T. 2020. SMART: Robust and Efficient Fine-Tuning for Pre-trained Natural Language Models through Principled Regularized Optimization. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*.

Kumar, A.; Ku, P.; Goyal, A.; Metallinou, A.; and Hakkani-Tur, D. 2020. Ma-dst: Multi-attention-based scalable dialog state tracking. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 8107–8114.

Lee, C.-H.; Cheng, H.; and Ostendorf, M. 2021. Dialogue State Tracking with a Language Model using Schema-Driven Prompting. *arXiv preprint arXiv:2109.07506*.

Li, M.; Xiong, H.; and Cao, Y. 2020. The sppd system for schema guided dialogue state tracking challenge. *arXiv preprint arXiv:2006.09035*.

Li, S.; Cao, J.; Sridhar, M.; Zhu, H.; Li, S.-W.; Hamza, W.; and McAuley, J. 2021. Zero-shot Generalization in Dialog State Tracking through Generative Question Answering. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, 1063–1074.

Lin, Z.; Liu, B.; Moon, S.; Crook, P. A.; Zhou, Z.; Wang, Z.; Yu, Z.; Madotto, A.; Cho, E.; and Subba, R. 2021. Leveraging Slot Descriptions for Zero-Shot Cross-Domain Dialogue StateTracking. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 5640–5648.

Liu, J.; Takanobu, R.; Wen, J.; Wan, D.; Li, H.; Nie, W.; Li, C.; Peng, W.; and Huang, M. 2020. Robustness Testing of Language Understanding in Task-Oriented Dialog. *arXiv preprint arXiv:2012.15262*.

Ma, Y.; Zeng, Z.; Zhu, D.; Li, X.; Yang, Y.; Yao, X.; Zhou, K.; and Shen, J. 2019. An end-to-end dialogue state tracking system with machine reading comprehension and wide & deep classification. *arXiv preprint arXiv:1912.09297*.

Mehri, S.; and Eskenazi, M. 2021. Schema-Guided Paradigm for Zero-Shot Dialog. *arXiv preprint arXiv:2106.07056*.

Mosig, J. E.; Mehri, S.; and Kober, T. 2020. Star: A schema-guided dialog dataset for transfer learning. *arXiv preprint arXiv:2010.11853*.

Mrkšić, N.; Séaghdha, D. Ó.; Wen, T.-H.; Thomson, B.; and Young, S. 2017. Neural Belief Tracker: Data-Driven Dialogue State Tracking. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1777–1788.

Noroozi, V.; Zhang, Y.; Bakhturina, E.; and Kornuta, T. 2020. A Fast and Robust BERT-based Dialogue State Tracker for Schema-Guided Dialogue Dataset. arXiv:2008.12335.

Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; and Liu, P. J. 2020. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. arXiv:1910.10683.

Rastogi, A.; Zang, X.; Sunkara, S.; Gupta, R.; and Khaitan, P. 2020a. Schema-guided dialogue state tracking task at DSTC8. *arXiv preprint arXiv:2002.01359*.

Rastogi, A.; Zang, X.; Sunkara, S.; Gupta, R.; and Khaitan, P. 2020b. Towards scalable multi-domain conversational agents: The schema-guided dialogue dataset. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 8689–8696.

Ruan, Y.-P.; Ling, Z.-H.; Gu, J.-C.; and Liu, Q. 2020. Fine-tuning bert for schema-guided zero-shot dialogue state tracking. *arXiv preprint arXiv:2002.00181*.

Sennrich, R.; Haddow, B.; and Birch, A. 2016. Improving Neural Machine Translation Models with Monolingual Data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 86–96.

Shah, D.; Gupta, R.; Fayazi, A.; and Hakkani-Tur, D. 2019. Robust Zero-Shot Cross-Domain Slot Filling with Example Values. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 5484–5490.

Wei, J.; Bosma, M.; Zhao, V. Y.; Guu, K.; Yu, A. W.; Lester, B.; Du, N.; Dai, A. M.; and Le, Q. V. 2021. Finetuned Language Models Are Zero-Shot Learners. arXiv:2109.01652.

Wu, C.-S.; Madotto, A.; Hosseini-Asl, E.; Xiong, C.; Socher, R.; and Fung, P. 2019. Transferable Multi-Domain State Generator for Task-Oriented Dialogue Systems. arXiv:1905.08743.

Yoo, K. M.; Shin, Y.; and Lee, S.-g. 2019. Data augmentation for spoken language understanding via joint variational generation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, 7402–7409.

Zhang, Y.; Noroozi, V.; Bakhturina, E.; and Ginsburg, B. 2021. SGD-QA: Fast Schema-Guided Dialogue State Tracking for Unseen Services. *arXiv preprint arXiv:2105.08049*.