Crowdsourcing syntactically diverse paraphrases with diversity-aware prompts and workflows

Jorge Ramírez[1][0000−0003−0678−8962], Marcos Baez[1][0000−0003−1666−2474], Auday Berro[1][0000−0003−2411−5761], Boualem Benatallah[2][0000−0002−8805−1130], and Fabio Casati[3][0000−0001−7591−9562]

[1] LIRIS – University of Claude Bernard Lyon 1, France
[2] University of New South Wales, Australia
[3] ServiceNow, USA
auday.berro@univ-lyon1.fr

Abstract. Task-oriented bots (or simply bots) enable humans to perform tasks in natural language. For example, to book a restaurant or check the weather. Crowdsourcing has become a prominent approach to build datasets for training and evaluating task-oriented bots, where the crowd grows an initial seed of utterances through *paraphrasing*, i.e., reformulating a given seed into semantically equivalent sentences. In this context, the resulting *diversity* is a relevant dimension of high-quality datasets, as diverse paraphrases capture the many ways users may express an intent. Current techniques, however, are either based on the assumption that crowd-powered paraphrases are naturally diverse or focus only on *lexical* diversity. In this paper, we address an overlooked aspect of diversity and introduce an approach for guiding the crowdsourcing process towards paraphrases that are *syntactically* diverse. We introduce a workflow and novel prompts that are informed by syntax patterns to elicit paraphrases avoiding or incorporating desired syntax. Our empirical analysis indicates that our approach yields higher syntactic diversity, syntactic novelty and more uniform pattern distribution than state-of-the-art baselines, albeit incurring on higher task effort.

Keywords: crowdsourcing · paraphrasing · diversity · task-oriented bots

1 Introduction

Task-oriented chatbots (or simply bots) allow users to interact with software-enabled services in natural language, for example, to perform tasks such as booking a restaurant or checking the weather. Such interactions require bots to process utterances (i.e., user input) like *"find restaurants in Milan"* to identify the user's intent (e.g., *"find restaurant"*) and the entities (i.e., slots) associated with the intent (e.g., *location="Milan"*). The success of intent recognition models depends entirely on the quality (and size) of the dataset of user utterances used for training and evaluating these models. A prominent approach to build datasets to support the training and evaluation of intent recognition models involves

expanding an initial set of seed utterances (for the intents) by means of *paraphrasing*. Paraphrasing is a task that aims to reformulate a given utterance into its many possible variations to generate semantically equivalent sentences [18].

An important dimension to measure quality in paraphrasing is *diversity*, i.e., the breath and variety of paraphrases in the resulting corpus, which dictates the ability to capture the many ways users may express an intent. In this context, paraphrasing techniques generally rely on approaches that aim at introducing *lexical* and *syntactic* variations [22]. Lexical variations refer to changes that affect individual words, such as substituting words by their synonyms (e.g., *"search restaurants in Milan"*). Syntactic variations, instead, refer to changes in sentence or phrasal structure, such as transforming the grammatical structure of a sentence (e.g., *"Where can we eat in Milan?"*). This richness of human language (and its potential ambiguity [24]) emphasizes the importance of paraphrasing for building diverse datasets. Missing to train models on such variations of language may result in bots failing to recognize intents and slots, thus performing tasks diverging from a user's actual intention and degrading their experience [27].

Having more control over the type of variations introduced can steer the paraphrase generation process towards more diverse and useful paraphrases for training and testing models for downstream tasks [6, 3]. For any real-world usage, it is critical to test the generalization capabilities of models. Adversarial examples, crafted by introducing syntactic variations (besides lexical changes) to seeds, can help "break" models and identify the boundaries of their capabilities [9]. Robustness may be increased by training models on augmented data, resulting from applying transformations (like paraphrasing) to training datasets. Thus having more control over the paraphrasing process can help increase the overall diversity of the training datasets to counteract adversarial examples [9].

However, while the development of specific techniques to guide the paraphrase generation process towards syntactic variations is the focus of ongoing work in automatic paraphrasing [3], they are currently greatly under-explored in the crowdsourcing literature. Among the few contributions towards diversity in crowdsourcing, a prominent data collection framework involves turning crowd-based paraphrasing into an iterative and multi-stage pipeline, chaining together multiple paraphrasing rounds. A different approach aims at increasing diversity by focusing on the task design itself [13, 27]. While valuable, these state of the art approaches assume workers would naturally produce diverse paraphrases or focus primarily on lexical variations (see Section 2).

In this paper, we present a multi-stage paraphrasing pipeline designed to guide the crowdsourcing process towards producing paraphrases that are syntactically diverse and balanced. Unlike prior work, the pipeline supports a *workflow* that can extract syntax patterns from crowdsourced paraphrases, and identify target patterns that should guide the generation task. We adopt the definition of syntax pattern, as the top two levels of a constituency parse tree [9], and select target patterns based on a pattern selection strategy (e.g., frequent or infrequent patterns). The paraphrase generation task then includes novel *prompts* that can elicit paraphrases conforming to the target syntax patterns (patterns by

example), or avoiding frequent patterns (taboo patterns). With this approach, we are exploring strategies to elicit more diverse paraphrases by steering the crowd away from over-represented syntax patterns, or guide workers towards less frequent patterns that should have more representation.

Contribution. The contributions of the work are in (i) an approach to guide the crowdsourcing process towards syntactically diverse paraphrases, (ii) workflows and prompts that can elicit paraphrases informed by syntax patterns, (iii) empirical evaluation of state-of-the art approaches, and the proposed strategies, for the generation of syntactically diverse paraphrases, and (iv) we contribute crowdsourced datasets to further study syntactic diversity.

2 Related Work

Crowdsourcing is a widely used approach to paraphrase generation [23]. It is a popular strategy as it can help scale the paraphrases generation efforts while reducing the costs, compared to hiring experts [14]. Two important aspects when it comes to diversity in this context are the *workflow* and *task design*.

In a crowdsourced process, an initial seed utterance, usually provided by an expert or generated using generative models or grammars [21], is presented as a starting point, and workers are asked to paraphrase the seed to new variations. A standard approach to introduce diversity in this context is to see the crowdsourcing task as a brainstorming session [1], where different perspectives are sought after. Here the assumption is that relying on crowd workers from different countries, backgrounds and demographics, will introduce diversity in the paraphrase generation process [1].

An improvement over the standard process involves turning crowd-based paraphrasing into an iterative and multi-stage workflow. This approach chains together multiple rounds of paraphrasing. The seed utterances for a round come from a previous round by using different seed selection strategies [17, 10, 12]. An approach to this is to use random sampling [10], which replaces a seed by randomly selecting one of its paraphrases from the previous round. An alternative is to replace the seed using semantic outliers [12], where the idea is to look for unique yet valid paraphrases to show to workers. This strategy uses sentence embeddings to represent each paraphrase to then scores these based on their distance to the mean vector. The paraphrases further away from the mean vector are defined as outliers. Another strategy is to just choose all the paraphrases from the previous round as seeds, looking for a multiplier effect, an approach known as Chinese whispers [17]. The focus of these strategies is to ultimately reduce the priming effect of seed utterances and examples [23] that would influence workers towards similar sentences.

Task design is another important aspect that affects the diversity of crowd-based paraphrasing[17, 10]. Jian et al. [10] explored relevant task design dimensions including number and type of examples in the prompts, number of paraphrases requested and workflows, assessing their impact on general diversity. The study found that the number of paraphrases requested did not significantly

affect the diversity of the outcome, but that workflows and prompts do have a significant contribution. Prompts providing only lexical examples lead to higher semantic relevance but lower diversity than showing a mix of syntactic and lexical examples. They also observed that a workflow based on Chinese whispers [17] can increase the diversity but at the cost of a lower semantic relevance. Overall, although the study did not focus on syntactic diversity, it provides further support for exploring prompts and workflows to improve diversity.

Early work on prompts focused on exploring general priming effects of different types of prompts. Wang et al. [23] explored paraphrasing prompts such as *sentence-based*, based on presenting seed sentences, *scenario-based*, that adopts a story-telling approach instead of directly showing a seed, and *list-based*, where only the goal is presented along with the required slot values. This study found that the list method is the one introducing less priming, with the other two priming workers with their choice of words and language. Most recent work has explored prompts to guide workers towards lexical diversity. Some words in the seed may be swapped with images (e.g., replacing the entity flight with the image of an airplane) [20]. However, this may be hard to apply beyond entities (e.g., finding images for verbs). *Taboo words* have been used instead to constraint the crowd from using frequently-used words [13], and *word recommendations* to help workers in choosing words to incorporate [27] Although these works focus on lexical diversity, they inspired our approach to steer workers towards or away from syntactic patterns to drive the process towards syntactic diversity.

In contrast to the crowdsourcing literature, the controlled generation of syntactic paraphrases has been the focus of research in *automatic* paraphrasing (e.g., [9, 8, 4]). Works on automatic paraphrasing models have proposed different architectures to disentangle semantic and syntactic properties, and allow for an additional input denoting the target syntax. Relevant to our discussion is the representation of syntactic templates guiding the generation. As syntactic specification, these approaches have leveraged explicit representations, such as constituency parse tree [9, 8] or learn more abstract syntax representations from the data [4]. In the latter, the input to the model are exemplars (i.e., sentences providing example expressions to mimic). We stress, however, that despite automatic approaches being a promising and emerging direction (with its own quality issues to address [25, 18, 2]), crowdsourcing is still a very important technique that can greatly benefit from the research on diversity. Crowdsourcing is actively used for collecting training data, generating adversarial examples for intent recognition models, and even to support the training and evaluation of automatic paraphrasing techniques.

The above tells us that using workflows and prompts for syntactic diversity is an unexplored area in crowdsourced paraphrasing. This also means that it is unclear to what extent current state-of-the-art approaches are able to deliver on syntactic diversity. In this paper, we draw inspiration from automatic paraphrasing based on syntactic patterns, as well as lessons learned from crowdsourced paraphrasing. We explore whether crowdsourcing workflows and prompts can guide the paraphrase acquisition process towards syntactic diversity.

3 Crowdsourcing Syntactically Diverse Paraphrases

In this section, we present our approach to guide the paraphrase generation process towards syntactically diverse paraphrases. In what follows we describe the syntax-aware workflows and prompts that are at the core of our approach.

3.1 Paraphrase generation workflow

Figure 1 depicts our approach that *abstracts* and *extends* state of the art paraphrasing workflows [17, 11, 12] into an iterative and multi-stage pipeline targeting syntactic diversity. We can define the typical data collection process as broken into multiple rounds of three main steps: *paraphrase generation*, *paraphrase validation*, and *seed selection*.

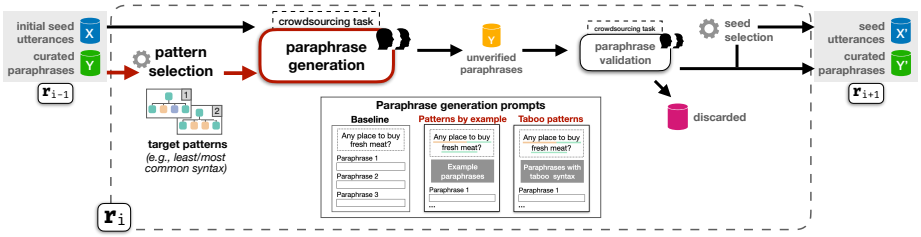

Fig. 1. Our approach consolidates and extends state-of-the-art paraphrasing workflows into an iterative and multi-stage pipeline aiming for syntactic diversity.

A data collection round r, for a typical workflow, takes as input a dataset of seeds utterances X and a curated collection of paraphrases Y (initially, Y can be empty) and proceeds by querying the crowd for paraphrases via predefined prompts. The prompts in the paraphrase generation step ask workers to provide a set m paraphrases for an utterance $x \in X$. These prompts generally rely on instructions and examples explaining the concept of paraphrasing, expecting workers to provide valid paraphrases according to the instructions. The resulting collection of unverified paraphrases $\bar{Y}$ is fed to the paraphrase validation step, where another crowd helps to assess the quality of candidate paraphrases, typically assessing semantic relevance[4]). The valid paraphrases are then appended to the collection of curated paraphrases Y. Finally, the seed selection step updates (or fully replaces) the seeds in X by sampling from the correct paraphrases to create the set of seeds for the next round. In this abstraction, we can model state-of-the-art workflows as instances implementing different seed selection strategies (e.g., random sampling [10], or identifying outliers [12]).

To steer the process towards syntactic diversity, we introduce the notion of syntax patterns into the workflow as well as pipeline components that can extract and identify target patterns from an input paraphrase corpus to guide the generation task. In this extended workflow, an input paraphrase corpus Y is provided along with the initial seed utterances X. The input paraphrase corpus

[4] Refer to [26, 28] for other relevant quality aspects in crowdsourced paraphrases

provides a curated list of paraphrases from where syntactic exemplars will be derived. Note that this curated list can be the output from a previous round (Y_{r_i-1}) or provided by experts. The *pattern selection* component then extracts the syntactic *patterns* for each paraphrase in the input paraphrase corpus, capturing the different syntactic variations present in the corpus. To direct the crowd away from (or towards) specific syntax, the pattern selection step proceeds by narrowing down this list to a set of *target patterns*, according to a selection strategy. These target patterns and the associated paraphrases in the corpus are handed over to the paraphrase generation step, where novel *prompts* take advantage of this additional input to query the crowd for paraphrases — ensuring workers conform with (or avoid) specific syntax. In the following subsections, we expand on these components and the notion of patterns in more detail.

3.2 Pattern representation and selection

To capture and control syntax, we follow [9] and define a pattern as the top two levels of a constituency parse tree, as shown in Figure 3a. For example, the extracted syntax pattern for *"search for a restaurant"* in bracket notation would be `(S (VP))`. This pattern denotes a simple declarative clause, with a verb and dependants. Instead, the syntax pattern for *"where to eat?"* would be `(SBARQ(WHADVP)(S))` which can be interpreted as a direct question introduced by a wh-word or a wh-phrase.[5] The pattern is thus a relaxed version of the full syntax tree, since the nodes at the top two levels are mostly clause/phrase level nodes. This takes syntax comparisons at a higher level of abstraction, which was deemed appropriate for guiding syntactic variations in prior work [9, 8].

Based on this definition of pattern, the *pattern selection* component aims at identifying target patterns that would inform the generation process. The component starts by first extracting the underlying syntax pattern for each paraphrase in Y. To do so, we obtain the linearized parse tree for the paraphrase using the Stanza NLP toolkit [19] and the Stanford CoreNLP package [16]. The pattern (the top two levels) is extracted from the full syntax tree based on the algorithm and code shared in [9]. As a result of this step, we have K unique syntax patterns, where each p_k is associated with one or more paraphrases in the input corpus Y of curated paraphrases.

To support the selection of the pattern, the component then builds a pattern frequency table. This is calculated by looking at each unique syntax pattern p_k, and counting the number of paraphrases in the corpus it is associated with. The pattern selection component then selects a subset of n patterns as the target patterns, by applying a pattern *selection strategy*. In the context of this work these strategies are based on pattern frequency table a) *least-frequent patterns* (bottom-n) or *most-frequent patterns* (top-n). Either choice (top-n or bottom-n) informs the paraphrase generation step differently, producing different prompts.

It is relevant to mention that for practical reasons, this component expects well-formed and grammatically correct paraphrases as part of the input corpus

[5] Reference for bracket labels at https://gist.github.com/nlothian/9240750

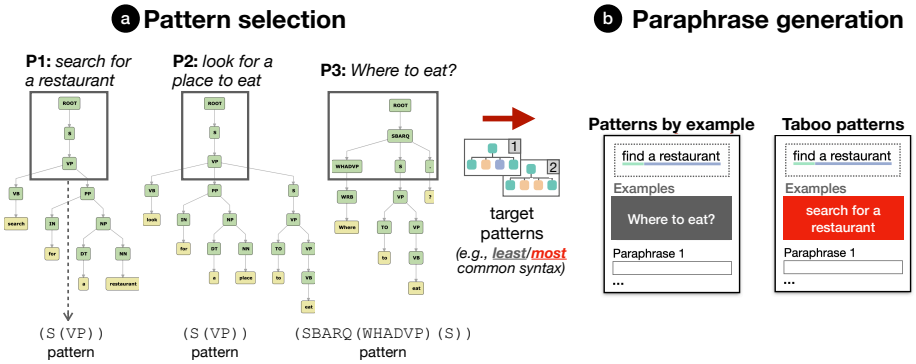

Fig. 2. A pattern is defined as the top two levels of a constituency parse tree. Patterns identified with pattern selection strategy inform paraphrase generation prompts.

Y. The tools we employ for extracting the parse tree and deriving the patterns [19, 16] may interpret errors in the paraphrases (e.g., typos, misused verb forms) as part of new patterns. As relevant literature suggests, crowdsourced paraphrases are subject to such errors [26].

3.3 Paraphrase generation prompts

In a crowdsourced paraphrasing process, the paraphrase generation is modeled as a crowsourcing task, generally deployed on a crowdsourcing platform. The typical task provides instructions explaining the concept of paraphrasing, and prompting crowd workers to provide x paraphrases for a given seed (refer to Figure 1 for an illustrative example of this baseline task). Except for the work on lexical diversity (see Section 2), current paraphrasing prompts rely on the assumption that engaging workers of diverse demographic and background will naturally lead to diversity. In this work, we propose two novel prompts that aim at eliciting workers paraphrases that conform or differ from a target pattern (informed by the *pattern selection* strategy). We posit that by designing syntax-aware prompts we can more effectively guide the crowd towards syntactic variations.

Patterns by example. The *patterns by example* prompt (Figure 3b) aims to guide crowd workers towards providing paraphrases featuring desired target patterns. While these patterns could also be provided by experts, in this work we explore the use of least-frequent patterns inferred from the corpus Y. In feeding the prompt with target patterns identified with the least-frequent pattern selection strategy (bottom-n patterns), the idea is to elicit paraphrases conforming with patterns that are currently unrepresented in the input corpus. The design of the prompt, as seen in the figure, incorporates elements of the baseline task, but includes additional instructions, syntactic examples, and validators. The *instructions* request workers to provide (m=3) paraphrases inspired by the sentence structures illustrated in concrete example sentences. These *syntactic examples* are k example paraphrases (in our experiments $k = 2$),[6] randomly

[6] We set $k = 2$ as prompts from prior art typically include two examples [17].

sampled from the list of paraphrases in Y, featuring the target syntax patters (one example per target pattern). The *validators* are a list of checks we built into the task to ensure workers do not provide paraphrases with the same pattern as the input seed x, but did not enforce strict compliance with the target patterns. Thus, we relied on the priming effect of the examples but still allowed for novel syntax (i.e., a patterns not found in any example).

Taboo patterns. The taboo patterns prompt is inspired by existing work on lexical diversity [13], and it aims to steer the crowd away from over-represented syntax patterns. The prompt is thus informed by the target patterns obtained with the most-frequent pattern selection strategy (top-n patterns). As with the previous prompt, the design of taboo pattern prompt extends the baseline with additional instructions, example taboo syntax and validators. The *instructions* in this case instructs workers to provide (m=3) paraphrases featuring structures different from those given in syntactic examples. These *taboo pattern examples*, are k example paraphrases (in our experiments $k = 2$), selected for each taboo pattern by randomly sampling one paraphrase featuring the given pattern. The *validators* then ensure that workers do not provide paraphrases with the same pattern as the input seed x and any of the taboo patterns.

To avoid certain well known issues in crowdsourced paraphrasing [26], we incorporated additional checks among the task validators: (i) checking for duplicates, by comparing the workers' contribution to the input seed and examples, after preprocessing (lemmatizing and lowercasing), (ii) avoiding gibberish, as in [15], (iii) ensuring the paraphrases feature the parameters (or slots) in the input seed. If a worker's contribution failed to pass these checks, and the prompt-specific checks, the worker was informed of the issue and reminded of the instructions.

4 Experiment Design

The experiment is set to explore whether our approach can effectively steer the crowd towards syntactic variations. We focus on the impact of the syntactic control introduced on relevant dimensions such a syntactic diversity and pattern distribution as well as important crowdsourcing metrics such as task effort. We compare our approach to state-of-the art workflows and assess their ability to generate syntactic variations. In what follows we summarize the design of the experiment, and provide additional details on datasets, annotations, and data analysis as supplementary materials.[7]

Datasets. We considered seed utterances representing a broad set of intents and domains, drawn from three relevant datasets. We selected 20 seeds from ParaQuality [26], a dataset that contains seed utterances (and their paraphrases) for intents from domains including Scopus, Spotify, Open Weather, Gmail, AWS, among other services. All seven intents from the SNIPS dataset [5] were also considered, randomly sampling three utterances per intent to be used as seeds. Finally, we used utterances from the ATIS dataset [7], where intents corresponds

[7] Online supplementary material available at https://tinyurl.com/caise-2022-diversity

to interactions with a flight-booking bot. We considered the top-5 intents from the training set[8] and sampled 10 utterances (2 per intent) as seeds for our experiment. The resulting dataset contains 51 seed utterances for a total of 24 intents. We provide the list of input seeds as part of our supplementary material.

Experimental conditions. For the experimental conditions, we chose as baselines the three multi-stage approaches to diversity from the literature (see Section 2). We considered these as meaningful baselines as they had the potential to introduce syntactic variations, whereas the approaches focused on lexical diversity were not considered, due to their focus on word-level changes (e.g., replacing words by synonyms) rather than syntax. We compared these baselines to our two strategies based on taboo and example patterns.

All baseline conditions follow the reference state-of-the-art workflow (see Section 3.1) with two rounds ($r = 2$). They rely on the same baseline prompt that simply queries for three paraphrases for a given input seed, but implement different seed selection strategies. The ❶ *baseline* condition comprises a workflow that uses the baseline prompt for paraphrase generation and random sampling [10] for seed selection, which substitutes the each seed from the previous round (r_{i-1}) with one correct paraphrase. The ❷ *baseline-cw* condition represents a similar workflow but the seed selection step chooses 8 valid paraphrases per input seed (instead of one), mimicking the Chinese whispers approach [17]. Similarly, the ❸ *baseline-outliers* condition constitute a workflow that for seed selection, it selects one semantic outlier (but correct) paraphrase per input seed, resembling [12].

The two other experimental conditions constitute our approach. Both shared the same extended workflow, and as the baseline conditions were conducted in two rounds. The ❹ *patterns by example* condition sets a workflow where the pattern selection step chooses the bottom-k patterns as targets, and these are used to set up the pattern by example prompt for paraphrase generation. The ❺ *taboo patterns* condition, instead, sets the top-k patterns as targets, and these are used to set up the taboo patterns prompt to elicit paraphrases from workers. We set $k = 2$ for both prompts (i.e., workers are shown two example paraphrases). In these workflows, the seed selection step does not update the input seeds in X (i.e., subsequent rounds use the same seeds). We should note that the specific task designs were refined through internal and external pilots.

Procedure. We conducted two full rounds of the pipeline shown in Figure 1. The first round (r_1) helped to bootstrap a dataset of curated paraphrases for the 51 seeds of the experiment. This round was shared by all the experimental conditions, using the baseline prompt for paraphrase generation. The bootstrap round collected 1224 paraphrases in total (24 per seed). The paraphrase validation was performed manually by the researchers as explained in the next subsection.

The second and main round (r_2) of the experiment ran all the experimental conditions in parallel, using as input the output dataset from the bootstrap round. In this context, we applied the seed selection, pattern selection and prompts configured based on the specific experimental condition.

[8] We used the dataset available at https://www.kaggle.com/siddhadev/atis-dataset-from-ms-cntk. The top-5 intents are those with the highest number of training items.

Table 1. Relevance criteria used in manual paraphrase validation [28]. Paraphrases correspond to the intent BookTaxi and seed *"Request a taxi from the airport to home"*

Criteria	**Examples mistake**
Semantic similarity to seed	*Where do I need to go to pick a taxi from airport to home?* (asking for location)
No extra parameter / slots should be added	*I need a taxi from the airport to home for tomorrow*
Generalizations and specializations beyond the scope of the intent are not allowed	*How can I get home from the airport?* (generalization)
Only spelling mistakes such as missing / duplicated articles and typos are tolerated	*Now get a taxi to make me from the airport to my home*
Slot values should not be swapped	I need a taxi from home to the airport
The paraphrase should contain the action	~~*Request*~~ *A taxi from the airport to home*
Turning the original intent into a "composite intent" is not allowed	*Search a taxi from the airport to home and book it for me*
Example valid paraphrase:	*Please book a taxi from the airport to home*

We ran the experiment on Toloka and recruited workers who had passed an English test (set by the platform) and were ranked top-40%. In all conditions, except for *baseline-cw*, each seed was assigned to 8 workers, and each worker wrote 3 paraphrases for a seed (yielding 24 paraphrases per input seed). Since *baseline-cw* relies on 8 seeds per intent, instead of one, we assigned one seed (yielding a total of 24 paraphrases as in the other conditions). Workers were paid 0.15 USD per solved prompt. This reward stems from multiple pilots aimed to estimate task completion time and target a minimum hourly wage.

Paraphrase validation. A manual validation of the two rounds was performed based on as set of criteria informed by previous work on crowdsourced paraphrasing mistakes [28], which we summarize in Table 1. Two researchers first annotated a small sample of paraphrases to calibrate the criteria, to then tag a 20% of randomly selected seeds, from which a random 20% of paraphrases were selected. The resulting inter-coder agreement was 95%. After this, the researchers split the rest of the dataset and independently performed the annotation. The researchers were condition-blinded, meaning that all paraphrases from all conditions were mixed together with the associated condition hidden, to avoid any condition induced bias. As part of this process, the researchers also labeled borderline cases, and potentially valid paraphrases with minor typos or grammar mistakes. Borderline cases were discussed between the researchers resolved by consensus. The minor typos and mistakes were fixed in a clean version to minimize the chances of generating incorrect parse trees [19].

Data analysis. In understanding the effectiveness of our approach, we focus first on its ability to inform syntactic variations, i.e., the ability of the proposed prompts to steer workers away or towards the target prompts. For this, we use a measure of *syntactic similarity* proposed in [4] to compare target patterns with produced paraphrases. This metric applies the tree edit distance algorithm (TED) between two (full) syntax parse trees, after removing word tokens. Here, a low value suggests a high syntactic similarity, i.e., less number of edits in the syntax tree of one sentence to transform into the other.

We then focus on the impact of prompts on the quality of the resulting paraphrases as measured by metrics of semantic relevance, general and syntactic diversity, and the resulting pattern distribution. We should note that quality in crowdsourced paraphrasing is a much more involved concept (see [28]), and that here we focus specifically on how the elicited variations influence diversity and syntactic properties, while also assessing that resulting paraphrases are still semantically related and valid. For *semantic relevance* we rely on the manual paraphrase evaluation criteria previously described, and a complementary metric, BertScore [29], which is an automatic text similarity metric[9] based on contextual embeddings. For general *diversity*, we adopted DIV [11], computes diversity at corpus level by calculating n-grams changes between all pairs of utterances sharing the same intent. To measure *syntactic diversity* we relied on the syntactic similarity metric and applied to all pairs of utterances sharing the same intent to compute the mean syntactic distance. We also characterized the resulting *pattern distribution*, and observed to what extent the number of paraphrases per pattern were balanced. In particular, the mean distance of the paraphrase count per pattern in a seed to a uniform distribution.

Finally, we also assessed task completion time and task abandonment as a proxies for the percieved and actual effort incurred on workers.

5 Results

We collected a total of 7344 paraphrases from 877 workers, obtained from the bootstrap round (1224 paraphrases) and the main experiment (6120 paraphrases). We made the full crowdsourced dataset available[10], and summarize the distribution by condition in Table 2. While we have an overall high participation and representation of workers across conditions, some conditions attracted more participants (we discuss some reasons in Section 5.3).

5.1 Impact on the relevance of crowdsourced paraphrases

The results of the relevance by condition can be seen in Table 2 for the manual and automatic assessment. While improving relevance was not the focus of this work, we analyzed relevance to understand whether the experimental conditions impacted negatively on this quality dimension.

Human judgement. Even though the baseline conditions rely on the same baseline prompt, the seed selection strategy had an effect on the relevance of resulting paraphrases. We can see that `BASE-OUT` featured the lowest number of relevant paraphrases among the baselines (56.29%), which we attribute to having a semantic outlier as input seed. Albeit still relevant, the outlier might be pushing workers to contribute paraphrases that get semantically further from

[9] We stress that BertScore was not designed specifically for assessing paraphrases, so it does not capture the full range of criteria of the more specific manual evaluation.

[10] The datasets can be found at https://github.com/jorgeramirez/syntactic-diversity

Table 2. Summary of metrics and dataset distribution for the experimental conditions

Dataset	BASE	BASE-OUT	BASE-CW	PAT-TABOO	PAT-EXAMP
N	1224	1224	1224	1224	1224
Workers	203	209	166	164	135
Relevance	BASE	BASE-OUT	BASE-CW	PAT-TABOO	PAT-EXAMP
%Manual	**67.24**	56.29	63.56	53.10	65.60
BertScore	0.516	0.489	0.522	0.501	**0.528**
Diversity	BASE	BASE-OUT	BASE-CW	PAT-TABOO	PAT-EXAMP
S-Novel	3	2	3	**5**	**5**
$\text{S-TED}_{\text{main}}$	12.06	11.99	7.82	**15.58**	15.35
$\text{S-TED}_{\text{workflow}}$	12.36	12.95	11.27	**14.02**	13.82
DIV_{main}	0.677	0.672	0.494	0.729	**0.730**
$\text{DIV}_{\text{workflow}}$	0.691	0.706	0.666	**0.710**	0.703

the original seed (i.e., seed in the bootstrap round). In the literature, `BASE-CW` has also shown to produce less relevant paraphrases from the initial seed at each iteration [10], but in our experiments having based the seed selection on valid paraphrases reduced this effect (63.56%). We can see having `BASE` rely on random sampling for seed selection strategy resulted in a higher percentage of relevant paraphrases (67.24%) We see `PAT-EXAMP` coming second (65.6%) to the performance of the baseline (`BASE`), hinting that the specific steering strategy did not affect negatively on the relevance. However, `PAT-TABOO` came last (53.1%), suggesting that workers experienced difficulties contributing with paraphrases that avoided the taboo patterns. We expand on this aspect in Section 5.3.

Automatic assessment. By applying BertScore to the paraphrases we can see similar performances across conditions, but again with the conditions with the semantic outliers (`BASE-OUT`) and taboo patterns (`PAT-TABOO`) ranked last. We note that while BertScore has shown to correlate well with human judgement [29], the manual assessment relied on more specific criteria (see Table 1).

5.2 Guiding the crowd towards syntactic variations

We analyze the effectiveness of the proposed pipeline to steer the process towards syntactic variations by assessing: the syntactic control introduced by the prompts, the impact on diversity, and the overall pattern distribution. This analysis is performed over the subset of relevant paraphrases (manual evaluation).

Syntactic control. We started by assessing the level of syntactic control introduced by the proposed prompts. In the case of *taboo patterns*, the validators incorporated as part of the prompt design were effective in avoiding paraphrases featuring the given patterns. We observed no paraphrases matching the patterns presented as taboo. As for *patterns by example*, we also look at the conformity of paraphrases with the syntactic examples introduced by the prompt. Recall that patterns by example does not enforce conformity with the target patterns but use them to prime workers. Indeed, we see only a 19% of paraphrases in this condition matching the exact pattern of the examples shown to the workers. Taking the baseline prompt as a reference, the priming effect in this case results in 15% of paraphrases featuring the same pattern as the seed seen by the workers.

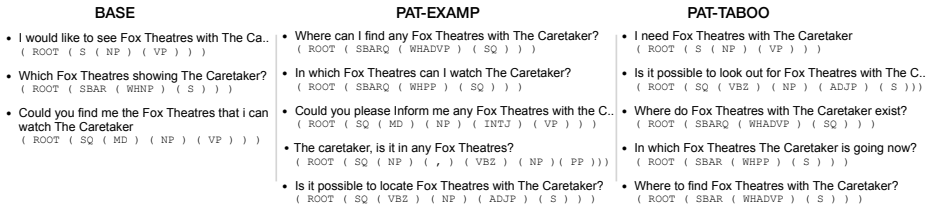

Fig. 3. Novel patterns and representative paraphrases for the seed *"find Fox Theatres with The Caretaker"*, illustrating results generated by BASE and the proposed prompts.

The syntactic similarity metric (Section 4) shows the mean edit distance to be 14.70 for `PAT-TABOO`, which is a higher distance than the 11.92 for `PAT-EXAMP`. This indicates the ability of these prompts to guide the syntactic variations.

Syntactic novelty. We then looked at the mean number of *unique syntax patterns* underlying the paraphrases contributed in each condition (main round only). Among the baselines, those where same seed is presented to all workers resulted in the lower mean number of unique patterns (`BASE`=5 and `BASE-OUT`=5). Chinese whispers did better (`BASE-CW`=6) and we attribute this to workers being primed with different seeds. The syntax-aware conditions were more effective, with patterns by example taking the first spot (`PAT-EXAMP`=8) followed by taboo patterns (`BASE-CW`=7). This suggests that examples priming workers towards less represented syntax might steer workers towards more unique syntax variations.

Taking the entire workflow perspective, we then looked at the *novel syntax patterns* introduced by the conditions with respect to the bootstrap round. That is, we calculated for each condition how many unique patterns were not present in the bootstrap round. As shown in Table 2 (S-Novel row), both syntax-aware approaches were more effective than the baselines in eliciting novel syntactic variations. The Friedman test shows the differences between the conditions are statistically significant ($X^2_F(5) = 90.06269$, $p < .0001$). According to pairwise comparisons using the Wilcoxon signed-rank test (with Bonferroni correction), the number of novel patterns for `PAT-TABOO` and `PAT-EXAMP` is significantly higher than the baseline conditions ($p < .001$).

Diversity. We now look at the syntactic and general diversity metrics. Considering the paraphrases for the main round only ($\text{S-TED}_{\text{main}}$), we observe the syntactic-aware conditions featuring higher *syntactic diversity* than the baseline counterparts. Taboo patterns performed only slightly better (15.58) than patterns by example (15.35). Looking at the entire workflow ($\text{S-TED}_{\text{workflow}}$), the general trend still favors the syntactic-aware conditions. These results ultimately highlight the benefits of introducing the notion of patterns into the workflow, as seen by the syntactic diversity reached by the conditions rendering our approach.

We also assessed diversity with DIV. Focusing on the paraphrases for the main round (DIV_{main}), we see the syntax-aware conditions resulted in a higher mean DIV score than the baselines, with both conditions featuring virtually the same scores (DIV=0.730). Among the baselines, `BASE` and `BASE-OUT` featured very similar scores (DIV=0.678 and DIV=0.672, respectively), leaving `BASE-CW`

with a way lower performance than the rest of the conditions (DIV=0.494). This means that when considering general diversity (lexical and syntactic) the proposed prompts still result in higher performance. Taking the workflow perspective to assess the contribution of the conditions to the bootstrap round, we see the scores balancing out ($\text{DIV}_{\text{workflow}}$). This indicates that the focus on syntactic variations might produce less lexically diverse paraphrases.

Pattern distribution. Our analysis of the pattern distribution showed that the syntactic-aware conditions lead to a distribution that is closer than the baselines to an equal representation of syntax patterns. `PAT-TABOO` displayed the overall lowest mean distance (1.94). In prompting users to avoid the top two common patterns, taboo patterns elicited paraphrases distributed among other patterns. Pattern by example, instead, contributed to some extent those specific syntax patterns shown to the workers (19% conformity as discussed part of syntactic control). In general, in providing no syntactic guidance, the baselines contributed more to a long tail-type distribution, with fewer patterns dominating the dataset.

5.3 Impact on task effort

Overall, the syntactic-aware prompts demanded a higher level of effort from workers. The median task completion time was 287s for `BASE`, 251s for `BASE-OUT`, 244s for `BASE-CW`, 321s for `PAT-TABOO`, and 326s for `PAT-EXAMP`. A Kruskal–Wallis test indicates the differences are significant ($H(4) = 42.56$, $p < .001$), with the Dunn's test of multiple comparisons (with Benjamini-Hochberg adjustment) showing `PAT-TABOO` and `PAT-EXAMP` were significantly slower than the baselines.

A high task abandonment was observed in the different experimental conditions — ranging from 45% to 67%. While this is common in crowdsourcing tasks, the task abandonment topped at 47% for the different baselines but was higher for patterns by example (57%) and taboo patterns (67%). Both `PAT-EXAMP` and `PAT-TABOO` introduced additional requirements to the task, making the it more challenging. Especially for `PAT-TABOO`, paraphrases needed to feature a pattern different than a seed and examples, and judging by the resulting relevance, this led to comparatively fewer valid paraphrases.

6 Discussion & Conclusion

Our results provided insights into the effectiveness of the syntactic-aware approach, and shed light into the extent to which the assumptions of diversity of the baseline approaches apply. We summarize our main findings below.

The syntactic control is effective in eliciting unique and novel syntax patterns. The proposed prompts were effective in guiding workers towards (or away from) specific syntax, as indicated by the conformity and syntax similarity metrics. This control ultimately yields a higher number of unique syntax patterns, showing the potential of running our approach in a unique round (e.g., with input from experts). When taking a workflow perspective, our approach elicited more novel syntax (almost double) when compared to the literature.

Effective in eliciting syntactically diverse paraphrases. Our results confirm the added benefits of steering individual workers towards (or away) specific syntax patterns, in eliciting paraphrases featuring more diverse syntax structures (S-TED); these results applied when considering one or two rounds.

Reduced long-tail effect with syntactic guidance. We have seen that our syntactic-aware approach is able to elecit more uniform pattern distributions, while the baselines with no guidance lead to paraphrases accumulated around certain patterns. In particular, we observed taboo patterns to contribute more to this uniformity than patterns by example.

Higher perceived and actual task effort in syntax-aware prompts. The proposed prompts were generally more challenging for workers as indicated by the higher task completion time and abandonment, especially for taboo patterns. Asking workers to avoid popular structures incur in effort that can lead to higher abandonment as well as more non-relevant results.

The above results tell us that improvements in syntactic diversity will come at the price of an increased task effort (23%-25% more effort and, therefore, budget). This makes our proposal a suitable approach when looking to effectively inject novel and more diverse syntactic structures, but not necessarily as a general approach. However, having specialized mechanisms, as the ones proposed in the paper, can provide workflow designers with more control over the type of variations introduced depending on the goal (e.g., generating adversarial examples, training a model). Indeed, combining techniques in paraphrasing workflows and ensembles is an emerging strategy in paraphrase acquisition [2].

Limitations. Despite the systematic approach we followed to the experimentation, we should note some existing limitations. The differences in perceived task difficulty affected participant distribution between conditions, but in eliciting high number of paraphrases per conditions we ensured a high and representative minimum. The experiments were run on the crowdsourcing platform Toloka, which has a majority of crowd workers from east European countries. We mitigated this limitation by engaging workers with proven English level.

Conclusion. This paper empirically showed how a pipeline that incorporates a workflow and prompts informed by syntax patterns could guide the crowdsourcing process towards producing syntactic variations. Comparing to state-of-the-art baselines, our approach results in higher syntactic diversity and more uniform pattern distribution in the generated dataset, albeit with demanding more effort from the crowd. Our ongoing and future work investigates workflows that rely on combinations of techniques and prompts, including automatic approaches.sii

References

1. Bapat, R., Kucherbaev, P., Bozzon, A.: Effective crowdsourced generation of training data for chatbots natural language understanding. In: ICWE (2018)
2. Berro, A., et al.: Automated paraphrase generation with over-generation and pruning services. ICSOC 2021 (2021)
3. Berro, A., et al.: An extensible and reusable pipeline for automated utterance paraphrases. Proc VLDB Endow. (2021)

4. Chen, M., et al.: Controllable paraphrase generation with a syntactic exemplar. In: ACL (2019)
5. Coucke, A., et al.: Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces. CoRR **abs/1805.10190** (2018)
6. Goyal, T., Durrett, G.: Neural syntactic preordering for controlled paraphrase generation. In: ACL (2020)
7. Hemphill, C.T., et al.: The ATIS spoken language systems pilot corpus. In: Workshop Held at Hidden Valley, Pennsylvania, USA (1990)
8. Huang, K.H., Chang, K.W.: Generating syntactically controlled paraphrases without using annotated parallel pairs. arXiv preprint arXiv:2101.10579 (2021)
9. Iyyer, M., et al.: Adversarial example generation with syntactically controlled paraphrase networks. In: NAACL (2018)
10. Jiang, Y., Kummerfeld, J.K., Lasecki, W.S.: Understanding task design trade-offs in crowdsourced paraphrase collection. In: ACL (2017)
11. Kang, Y., et al.: Data collection for dialogue system: A startup perspective. In: Proc. HLT, Vol 3. pp. 33–40 (2018)
12. Larson, S., et al.: Outlier detection for improved data quality and diversity in dialog systems. In: NAACL-HLT (2019)
13. Larson, S., et al.: Iterative feature mining for constraint-based data collection to increase data diversity and model robustness. In: EMNLP (2020)
14. Lee, W., et al.: Effective quality assurance for data labels through crowdsourcing and domain expert collaboration. In: EDBT (2018)
15. Liu, P., Liu, T.: Optimizing the design and cost for crowdsourced conversational utterances. In: KDD-DCCL (2019)
16. Manning, C.D., et al.: The Stanford CoreNLP natural language processing toolkit. In: ACL (2014)
17. Negri, M., et al.: Chinese whispers: Cooperative paraphrase acquisition. In: LREC (2012)
18. Park, S., et al.: Paraphrase diversification using counterfactual debiasing. In: AAAI (2019)
19. Qi, P., et al.: Stanza: A Python natural language processing toolkit for many human languages. In: ACL (2020)
20. Ravichander, A., et al.: How would you say it? eliciting lexically diverse dialogue for supervised semantic parsing. In: SIGDIAL (2017)
21. Su, Y., et al.: Building natural language interfaces to web apis. In: CIKM (2017)
22. Thompson, B., Post, M.: Paraphrase generation as zero-shot multilingual translation. arXiv:2008.04935 (2020)
23. Wang, W.Y., et al.: Crowdsourcing the acquisition of natural language corpora: Methods and observations. In: (SLT) (2012)
24. Wasow, T., Perfors, A., Beaver, D.: The puzzle of ambiguity. Morphology and the web of grammar: Essays in memory of Steven G. Lapointe pp. 265–282 (2005)
25. Xu, Q., et al.: D-page: Diverse paraphrase generation. arXiv:1808.04364 (2018)
26. Yaghoub-Zadeh-Fard, M., et al.: A study of incorrect paraphrases in crowdsourced user utterances. In: NAACL-HLT (2019)
27. Yaghoub-Zadeh-Fard, M., et al.: Dynamic word recommendation to obtain diverse crowdsourced paraphrases of user utterances. In: IUI (2020)
28. Yaghoub-Zadeh-Fard, M., et al.: User utterance acquisition for training task-oriented bots: A review of challenges, techniques and opportunities. IC (2020)
29. Zhang, T., et al.: Bertscore: Evaluating text generation with bert. arXiv:1904.09675 (2019)