Exploring Data Augmentation in Neural DRS-to-Text Generation

Anonymous ACL submission

Abstract

Neural networks are notoriously data-hungry. This represents an issue in cases where data are scarce such as in low-resource languages. Data augmentation is a technique commonly used in computer vision to provide neural networks with more data and increase their generalization power. When dealing with data augmentation for natural language, however, simple data augmentation techniques similar to the ones used in computer vision such as rotation and cropping cannot be employed because they would generate ungrammatical texts. Thus, data augmentation needs a specific design in the case of neural logic-to-text systems, especially for a structurally rich input format such as the ones used for meaning representation. This is the case of the neural natural language generation for Discourse Representation Structures (DRS-to-Text), where the logical nature of DRS needs a specific design of data augmentation. In this paper, we adopt a novel approach in DRS-to-Text to selectively augment a training set with new data by adding and varying two specific lexical categories, i.e. proper and common nouns. In particular, we propose using WordNet supersenses to produce new training sentences using both *in-* and *out-of-*context nouns. We present a number of experiments for evaluating the role played by augmented lexical information. The experimental results prove the effectiveness of our approach for data augmentation in DRS-to-Text generation.

1 Introduction

Data augmentation is a systematic way of increasing data examples by altering the original data with controlled variations (Feng et al., 2021). It is a prevalent technique in computer vision (CV) for increasing dataset size by introducing slightly different and contextually similar examples (Yang et al., 2022).

Augmentation approaches are also becoming popular in many Natural Language Processing (NLP) applications as well. The most commonly used approaches to augment textual data are based on random swapping, random insertion, random deletions, synonyms replacement, back translation, and using generative models to get new context-aware data (Feng et al., 2021; Shorten and Khoshgoftaar, 2019). Notice that data augmentation in NLP is a very challenging task due to the constraint of producing a grammatical augmented text (Hou et al., 2018). Moreover, given the continuous nature of images, in CV the augmented version of an image rarely is *pragmatically* incorrect. In contrast, in NLP, preserving the contextual meaning of the sentence is, usually, a hard constraint. Indeed, bad model performance can be the consequence of augmented textual data that is grammatically incorrect or out-of-scope (Dong et al., 2017).

Recently, researchers working on text generation from meaning representations, i.e., graph-based Abstract Meaning Representation (AMR) (Banarescu et al., 2013; Flanigan et al., 2016) or Discourse Representation Structure (DRS), have put their efforts into generating text from logical representations, and vice-versa, using transformers and encoder-decoder-based neural models (Basile and Bos, 2011; van Noord et al., 2018; Noord, 2019; Wang et al., 2021; Amin et al., 2022; Wang et al., 2023). In this paper, we consider the specific problem of augmenting data in the context of neural DRS-to-Text generation task. DRS represents textual information in the form of events, concepts, and entities, i.e., names as discourse referents usually represented as variables in DRS, and logical relations between these entities i.e., quantifiers, conjunctions, negations, disjunctions, etc. (Bos, 2021; Kamp and Reyle, 1993; Jaszczolt, 2023). In Figure 1 a graphical representation of DRS in box format (on the left), its flattened version i.e., clausal format (on the right), and its corresponding textual representation (on the bottom) is displayed.

Neural DRS-to-Text generation is a type of data-

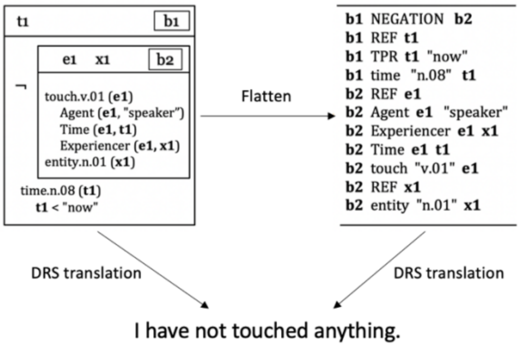

Figure 1: Box format and Clausal format of DRS along with their textual representation.

to-text generation task that takes the logical representation of a sentence as input and generates text as output (Wang et al., 2021; Amin et al., 2022). This is an application of text generation from structured input data similar to knowledge graphs (Flanigan et al., 2016), RDF triplets data (Gardent et al., 2017), and tables (Parikh et al., 2020). Note that, in contrast to tables and graphs, the ability to represent the structured logical nature of the input as a DRS generation allows for a more fine-grained investigation of the relation between input and output in DRS-to-Text. In other words *"changing the meaning of a DRS in a controlled way, the robustness of systems can be monitored in detail and assessed accordingly"* (Wang et al., 2021). However, this robustness property discourages the application of large language models (LLMs) for augmenting data because LLMs would generate noise in the augmented data (Feng et al., 2021; Hou et al., 2018; Dong et al., 2017) – see also Section 4.

In this paper, we exploit the robustness property of neural DRS-to-Text generation by designing and evaluating data augmentation for the specific categories of (i) proper nouns and (ii) common nouns. In particular, we have designed and evaluated a procedure for augmenting a DRS training dataset by adding *context-aware* new sentences that are produced by varying the proper and common nouns in the original sentences. We consider different strategies and propose to use Supersense Tagging (SST) for creating new training sentences using both *in- and out-of*-context nouns. In this way, we want to analyze the role played by lexical information in the performance of a neural DRS-to-Text system.

The research questions and contributions addressed in this paper are:

- Is it possible to augment a logical data representation such as DRS?
- How to generate new data that is contextually similar to the original one?
- What is the role played by the *in- and out-of*-contextual vocabulary for char-level and word-level decoder models? And what is the role of grammatical-semantic-pragmatic-world knowledge in learning?
- Does augmentation result in an increase or decrease in model performance by training a seq-to-seq model or fine-tuning a Transformer model?
- What is the behavior of pre-trained large language models (LLMs) i.e., ChatGPT and Claude, while analyzing DRS structures given as prompt?

To the best of our knowledge, apart from the preliminary work on augmentation of verbs presented in (Amin et al., 2022), this is the first work on data augmentation in DRS-to-Text generation analyzing its impact on model performance.

Notice that our augmentation techniques could generate factually incorrect texts (e.g., starting from "at dawn, the sun rises", "at night, the sun rises" could be generated. However, since humans can generate texts that are not factually correct (consider, for example, a sci-fi story), preventing this situation would actually be not only beneficial but detrimental for the system.

The statistical nature of the neural networks does not allow for an easy analysis of the kind of knowledge really learned by the system. When we provide a specific example as *Brad Pitt is an actor*, the network is learning that the verb follows the subject (e.g. grammatical competence), and/or that a man can be an actor (semantic and pragmatic knowledge), and/or that a specific man is an actor (world knowledge)? How can we exploit this multi-level nature of neural learning? A side effect of our study on data augmentation is to investigate these theoretical questions as well.

The paper is structured as follows: in Section 2, we describe the procedure adopted for noun augmentation; in Section 3, we give architectural insights on the neural DRS-to-Text pipeline; in Section 4, we describe the experimental results of DRS-to-text generation that uses (1) automatic metrics-based and pre-trained model-based evaluations on a standard test set, (2) a reduced test set comparing

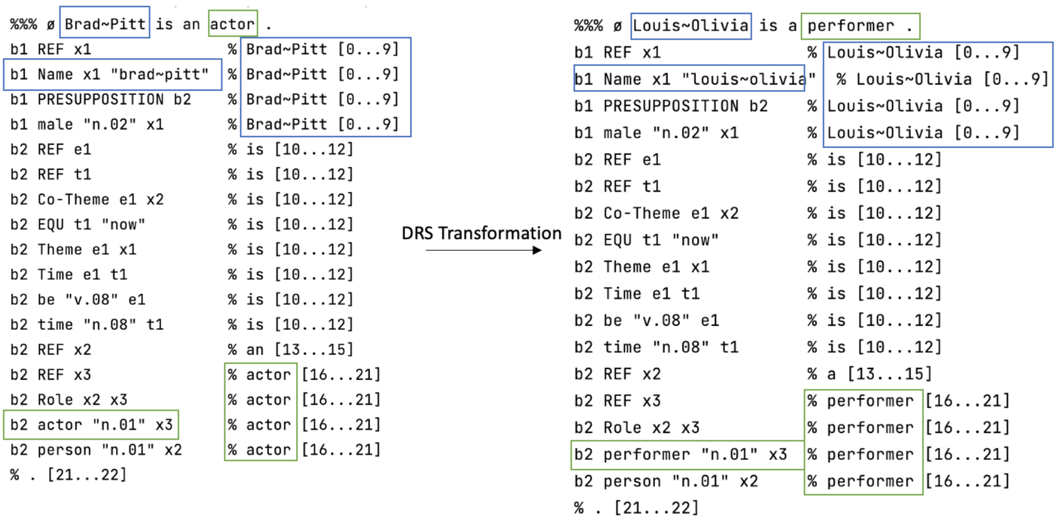

Figure 2: Graphical representation of the DRS transformation as a proper noun (in blue) and common noun (in green). The DRS on the left generates the sentence *Brad Pitt is an actor.*, while the DRS on the right generates *Louis Olivia is a performer.*

our neural systems with two general LLMs, and (3) applying both automatic and human evaluation metrics. Finally, in Section 5, we conclude the paper.

2 Logical Data Augmentation with Nouns

Data augmentation is a relatively complex task in the case of neural DRS-to-Text: each augmented example in the training set consists of a pair of new DRS structures together with a new corresponding sentence. While applying systematic transformations on training data, it is essential to keep track of both types of data representations as they are treated as *input value pairs* in the neural model. So, data transformations should be identical and symmetrical on both elements by considering the order of meaning representations and textual translations.

In the DRS-to-Text generation task, we applied different augmentation techniques to augment proper and common nouns. We have used the gold version of the Parallel Meaning Bank[1] (PMB) dataset, which is organized in the usual train-dev-test split.

A graphical representation of transformation for proper (highlighted in blue) and common (highlighted in green) nouns in DRS is shown in Figure 2: the DRS on the left generates the sentence *Brad Pitt is an actor*, while the DRS on the right generates *Louis Olivia is a performer* (see Table 1).

2.1 Proper Noun Augmentation

For proper nouns, we considered two specific name entity (NE) categories, which are the proper name of a person (PER) i.e., both male and female names, and of a place i.e., city, state, or country (GPE). We have used spaCy NE recognizer (https://spacy.io) to extract proper nouns from the text. There are a total amount of 3773 proper noun instances for PER and GPE. The proper nouns are divided as follows: person names 57%, city names 30%, state names 6%, country names 6%, and 1% of other types i.e., island names (see Figure 3 in Appendix).

We have used two procedures for replacing proper nouns to analyze the impact of adding external linguistic information to the dataset vocabulary[2]. (1) Replacing them with other proper nouns inside the same dataset, i.e., *inside context*. (2) Replacing them with proper nouns outside the dataset, i.e., *outside context*. *Outside context* refers to the fact that we chose just nouns different from the ones already present in the dataset. For replacing person names (PER) via *outside context* approach, we chose the person names based on the highest frequency of each name cited in the world (source: ChatGPT) that were not already in the dataset. For the city, state, and country names, we replaced them based on geographical distribution keeping in mind that the GPE names should not be in the

[1]The PMB is developed at the University of Groningen as part of the NWO-VICI project "Lost in Translation – Found in Meaning" (Project number 277-89-003), led by Johan Bos.

[2]While extracting NE, no offensive information was found.

Transf Type	Original Text	Transformed Augmented Text
Proper Noun	Brad Pitt is an actor.	Louis Olivia is an actor.
	Alice and Bob work for this company.	Maria and Tom work for this company.
	Turin is a beautiful city.	Venice is a beautiful city.
	Indiana is a very famous state.	Georgia is a very famous state.
	China is one of the top 5 populous countries in the world.	Indonesia is one of the top 5 populous countries in the world.
Common Noun	Brad Pitt is an actor.	Brad Pitt is a performer.
	Alice and Bob work for this company.	Alice and Bob work for this institution.
	Turin is a beautiful city.	Turin is a beautiful municipality.
	We painted the house green.	We painted the building green.
	The book rested on the table.	The novel rested on the furniture.
Proper and Common Noun	Brad Pitt is an actor.	Louis Olivia is a performer.
	The Mona Lisa hung above the antique table.	The Leonardo da Vinci hung above the antique furniture.
	Alice and Bob work for this company.	Maria and Tom work for this institution.
	Noah and Sophia watched a movie at the local theater.	Liam and Emma watched a show at the local edifice.
	Oliver and Isabella enjoyed the view of the mountains from the cabin.	Daniel and Lily enjoyed the view of the elevations from the compartment.

Table 1: Different flavors of augmentation applied to the dataset as single and blended data transformations.

dataset. For GPE, we again used the ChatGPT prompt which provided the list of available GPE entities extracted from the original dataset, to get a new list of GPE entities with the same geographical distribution. For example, in "The weather of Dubai is very hot and dry." we replace 'Dubai' with 'Sharjah' as the semantic correlation of hot weather holds true for both cities. Some examples listing proper noun augmentation are displayed in Table 1.

2.2 Common Noun Augmentation

Replacing a common noun without altering the contextual information of the sentence is a challenging task. To tackle this challenge, we adopt a novel SST approach to associate a category with the noun based on its contextual sense in the sentence. For the implementation of SST, we have used spaCy again. Based on data examples, we extracted 6193 common nouns belonging to the 26 lexicographic categories of WordNet, including act, artifact, body, cognition, communication, event, feeling, food, group, and motion (Ciaramita and Johnson, 2003) (see Figure 4 in Appendix).

In common noun augmentation, our approach considers two procedures: inside/outside dataset and preserving/not preserving supersenses (SS), thus resulting in four of the following combinations: (1) Replacing a common noun with any other common noun inside the dataset but not preserving SS: *"inside context without SS"*. (2) Replacing a common noun with another common noun having the same category of SS: *"inside context with SS"*. (3) Replacing a common noun with another common noun having the same category of SS but outside the dataset *"outside context with SS"*. (4) Replacing a common noun with another common noun not having the same category of SS but outside the dataset *"outside context without SS"*. For points (1) and (4), there is no guarantee of sustaining the contextual sense of the sentence as the noun replacement can happen between two different SS categories e.g., cat with chair[3]. For points (2) and (3), we make sure that the noun replacement has the same SS category e.g., cat with dog[4].

For points (3) and (4), for the sake of adding external lexical information for common nouns, we are taking the support of the WordNet lexical database. For point (3), we replace the common noun with its *WordNet hypernym* and then make sure that the new noun also belongs to the same SS category. For point (4), we just perform noun replacement through *WordNet synonyms*. All examples listed in Table 1 are also representing *outside context with SS* through *WordNet hypernyms*.

Note that in this work we have not performed other possible combinations for proper nouns, that is: changing GPE without considering the same class, i.e., changing city with state or country. The motivation lies in the fact that these combinations would radically change the semantics of the sen-

[3]As 'cat' belongs to 'noun animal' while 'chair' belongs to 'noun artifact' classes of SS. This can be grammatically true but not semantically and contextually.

[4]As 'cat' and 'dog' both belong to the same 'noun animal' class of SS. This type of substitution is grammatically, semantically, and contextually correct.

tence. In other words, we decided to follow a sort of *principle of minimum variation of the meaning* for choosing the augmentation strategy.

3 Three Neural DRS-to-Text Pipelines

DRS-to-Text generation is a complex logic-to-text generation task requiring computationally efficient neural models to transform logical representations. In our implementation pipelines, we have used three different neural architectures[5]. The first two models are based on an encoder-decoder oriented recurrent sequences-to-sequence neural networks with two bi-directional LSTM layers (Hochreiter and Schmidhuber, 1997; Junczys-Dowmunt et al., 2018), having (1) char-based lexical encoding (CB-bi-LSTM henceforth) and, (2) word-based lexical encoding (WB-bi-LSTM henceforth). Moreover, we have also used (3) a byT5 variant of *Transformer's* family (Xue et al., 2022) for fine-tuning the DRS-to-Text generation task (FT-byT5 henceforth).

We are aware that the state-of-the-art DRS-to-text generation models use sophisticated neural architectures (Liu et al., 2021; Wang et al., 2023), thus, encouraging us to use the Transformers-based model for our task as well. However, the goals of this paper are related to analyzing the effects of data augmentation in the context of neural DRS-to-text generation rather than providing a system with the best performances.

Note that the fundamental differences between CB-bi-LSTM and WB-bi-LSTM are based on input and output data representations, i.e., *characters or words* and their ability to handle out-of-vocabulary (OOV) words. The former deals with OOV words in a seamless way as it processes character sequences, while the latter could struggle to handle OOV words as it is dependent on the size of the included vocabulary. We believe that these two different approaches can drive the impact of specific techniques of data augmentation.

In our sequence-to-sequence implementation, the model architecture and hyperparameters used in our experiments are focused on LSTM-based encryption decryption cells having epochs-based learning decay strategy while using Adam as an optimizer. We have used cross entropy as the validation metric and ce-mean as the cost type function. Other important hyperparameters are in Table 5 of the Appendix. In our transformers-based implementation, we have used the default hyperparameter settings of byT5 with a little bit of change in batch size, update steps, and learning rates, while using AdamW as an optimizer and fine-tuning the model for 15 epochs. All hyperparameter settings of our FT-byT5 model are listed in Table 6 of the Appendix.

We have used the English version of the Parallel Meaning Bank (PMB) dataset. Among the different dataset types, i.e., gold, silver, and bronze, we have worked on the gold (fully manually annotated and corrected version) dataset. Gold-PMB follows the standard dataset division of training, development, and testing files having 6620, 885, and 898 data examples. In the process of augmenting the dataset, we have adopted two types of approaches to transform examples. (1) Apply one type of transformation and concatenate it with the original data examples. This approach will result in having more data with one type of data transformation, e.g., proper noun or common noun (indicated with the '+' sign in Table 2 and following). We have applied data augmentation to training examples only. Development and test files are the original ones, without any augmentation. (2) Apply multiple possible transformations (blend) on each example, e.g., apply proper noun and common noun augmentation on one example (indicated with the '-' sign in Table 2 and following). So, in this approach the training set size is smaller than in approach (1): in this way, we emphasize the role played by transformations rather than training set size. In Table 7 of the Appendix, we have listed the individual and blended data transformation along with training examples size.

4 Experimental Results

Evaluation with automatic metrics. We have conducted a series of different experiments that focus on analyzing the model performance based on systematic alterations in lexical semantics-based input representations. We have listed CB-bi-LSTM (see Table 2), WB-bi-LSTM (see Table 2), FT-byT5 (see Table 2), experimental results with the BLEU, NIST, METEOR, ROUGE, CIDEr, and BERTScore-based automatic evaluation measures (Wang et al., 2021; Amin et al., 2022; Zhang et al., 2020). In these experiments, we have used the standard split of PMB in train-dev-test sets. Note that the baseline of the experiment (1) is consistent with the results reported in (Amin et al., 2022), but is no-

[5]The code will be released upon acceptance.

tably inferior to the value reported in (Wang et al., 2021) because this latter study considers mixed gold-silver training data.

If we compare the overall performance of CB-bi-LSTM and WB-bi-LSTM, we found as expected that CB-bi-LSTM always wins in all aspects of input data. This reflects the fact that the char-level model with the ability to handle OOV words is performing very well in capturing micro-level aspects and data patterns of input DRS. This also shows the effectiveness and morphological accuracy of the char-level model in generating correct output sequences. However, the FT-byT5 model outperforms the bi-LSTM-based models in most experiments.

In the proper noun augmentation, our experiments are twofold: (1) *inside context* and (2) *outside context* as discussed in Section 2. Exp. $2-3$ (see Table 2), $11-12$ (see Table 2) and $20-21$ (see Table 2) list the results obtained after performing two flavors of proper noun augmentation. Considering only LSTM architectures, the experimental findings show that vocabulary plays a vital role in the case of CB-bi-LSTM model as this is more independent in sequence generation. Therefore, we have the highest score in CB-bi-LSTM for the proper noun augmentation *outside context* to the dataset (Exp. 3). On the other hand, the word-level decoder is more focused on vocabulary, therefore it has the highest scores in proper noun augmentation *inside context* to the dataset (Exp. 11). The latter represents the effectiveness of word-level models in generating coherent and grammatically correct output sequences while capturing correct syntax and semantic meanings of input DRS. P.N. augmentation shows the best results in the case of the FT-byT5 modes. In particular, we note that FT-byT5 with P.N. produced the highest values over all the metrics over all the experiments. We speculate that this result could depend on the peculiarities of the T5 original model. However, the important point for our study is to note that also in pretrained LLMs, data augmentation can play an important role in performance.

For common noun augmentation, our experiments are fourfold: (1) *inside context with SS*, (2) *inside context without SS*, (3) *outside context with SS*, and (4) *outside context without SS*: Exp. 4-7 (see Table 2) , 13-16 (see Table 2) and 22-25 (see Table 2) regard these four different flavors of common noun augmentation. We believe that the important role played by the vocabulary holds for common nouns as well, with the highest scores of CB-bi-LSTM decoder for *outside context with SS* (Exp. 6) and best WB-bi-LSTM score for *inside context with SS* (Exp. 13). Again, the FT-byT5 shows the best results with augmentation.

Finally, in Exp. 8-9 (see Table 2), 17-18 (see Table 2) and 26-27 (see Table 2), we have applied the best augmentation techniques of proper and common nouns (i.e., outside context for CB-bi-LSTM, and inside context for WB-bi-LSTM and outside context for FT-byT5) as blended and individual data examples. In Exp. 8, 17, and 26, the augmentation techniques have been applied simultaneously to each input data example (i.e., as we are applying 2 data transformations on one example, we name it blended, see proper and common noun example in Table 1). Here dataset examples are concatenated as *(original + P.N.-with-C.N.)*. While in Exp. 9, 18 and 27, these augmentation techniques have been applied separately and concatenated as *(original + P.N + C.N)* augmentation data examples. Comparing all experimental results, we achieved the highest scores for LSTM models while applying the best augmentation flavors of P.N and C.N concatenated as separate individual training examples (see Exp. 9 and 18). Similarly, in FT-byT5 the best value is for concatenated examples (see Exp. 27). However, surprisingly, in contrast with LSTM, we do not achieve the best values in FT-byT5 in the experiment 27 (see experiment 21). Again, we believe that this different pattern of T5 w.r.t. LSTM depends on the peculiarities of the original model.

Finally, in Exp. 28, we preliminarily evaluated the impact of the size of the augmented data. So, we repeated Exp. 21, that is the best on for Ft-byT5 model, by halving the size of the augmented part of the training set. The results, with scores that are intermediate w.r.t. the baseline and the best model, suggest that there is a linear increase w.r.t. the size of the augmented training set. However, more experiments are necessary to verify this hypothesis. We tested the statistical significance of the results with a *Wilcoxon Signed Rank Test* (Dror et al., 2018).

Comparing neural DRS-to-Text and LLMs. We compare the quality of the generated text of our neural DRS-to-Text systems with two recent general LLMs, ChatGPT 3.5 (OpenAI, 2023) and Claude 2.0 (Turpin et al., 2023) in order to provide a preliminary insight in the performance of our approach with respect to a general LLM that was not fine-tuned on this specific task. We have applied

Exp	Implementation Type	BLEU	NIST	METEOR	ROUGE	CIDEr	BERT Score
01	Gold-PMB (no Aug)	47.72	7.68	39.42	72.59	4.84	95.3
02	Orig + P.N. (in ctx) Aug	51.37 †	7.96 †	41.19 †	74.78 †	5.15 †	95.8
03	Orig + P.N. (out ctx) Aug	*53.16* †	*8.11* †	*42.00* †	*75.30* †	*5.27* †	*95.9*
04	Orig + C.N. (in ctx with SS) Aug	50.28 †	7.94	40.90 †	74.24 †	5.02 †	*95.7*
05	Orig + C.N. (in ctx w.o. SS) Aug	49.99 †	7.91	40.14 †	74.06 †	4.96 †	95.6
06	Orig + C.N (out ctx with SS) Aug	*50.89* †	*7.98* †	*40.70* †	*74.38* †	*5.08*	*95.7*
07	Orig + C.N (out ctx w.o. SS) Aug	50.63 †	7.93 †	40.39 †	74.33 †	5.06 †	*95.7*
08	Orig + P.N (out ctx)-with-C.N (out ctx with SS) Aug	52.51 †	8.06 †	41.23 †	75.28 †	5.24 †	96.0
09	Orig + P.N (out ctx) + C.N (out ctx with SS) Aug	**54.00** †	**8.19** †	**42.32** †	**76.15** †	**5.35**	**96.1**
10	Gold-PMB (no Aug)	32.91	5.80	29.99	61.39	3.49	94.4
11	Orig + P.N. (in ctx) Aug	*44.37* ‡	*7.37* ‡	*36.56* ‡	*69.54* ‡	*4.38* ‡	95.1
12	Orig + P.N. (out ctx) Aug	42.70 ‡	7.16 ‡	35.39 ‡	67.69 ‡	4.18	94.9
13	Orig + C.N. (in ctx with SS) Aug	***44.41*** ‡	*7.28* ‡	*36.22* ‡	*68.78* ‡	***4.34*** ‡	95.1
14	Orig + C.N. (in ctx w.o. SS) Aug	42.94 ‡	7.14 ‡	35.11 ‡	67.56 ‡	4.19	94.8
15	Orig + C.N (out ctx with SS) Aug	41.84 ‡	6.97 ‡	34.25 ‡	66.38 ‡	4.05	94.6
16	Orig + C.N (out ctx w.o. SS) Aug	42.41 ‡	7.13 ‡	35.01 ‡	67.47 ‡	4.16 ‡	94.8
17	Orig + P.N. (in ctx)-with-C.N. (in ctx with SS) Aug	43.78 ‡	7.21 ‡	35.87 ‡	68.52 ‡	4.27 ‡	95.0
18	Orig + P.N (in ctx)+C.N. (in ctx with SS) Aug	44.39 ‡	**7.36** ‡	**36.63** ‡	**69.53** ‡	4.29 ‡	**95.2**
19	Gold-PMB (no Aug)	51.88	7.94	43.55	76.04	5.63	96.7
20	Orig + P.N. (in ctx) Aug	55.72 ◇	8.23 ◇	45.05 ◇	77.81 ◇	5.91 ◇	97.1
21	Orig + P.N. (out ctx) Aug	**57.15** ◇	**8.33** ◇	**45.90** ◇	**78.81** ◇	**6.08** ◇	**97.2**
22	Orig + C.N. (in ctx with SS) Aug	53.08	8.04	44.20	76.64	5.68	96.8
23	Orig + C.N. (in ctx w.o. SS) Aug	52.85	8.00	44.50	76.32	5.69	96.8
24	Orig + C.N (out ctx with SS) Aug	*54.71* ◇	*8.13* ◇	*44.77*	*77.27*	*5.84*◇	*97.0*
25	Orig + C.N (out ctx w.o. SS) Aug	52.78	8.02	44.29	76.31	5.66 ◇	96.8
26	Orig + P.N. (out ctx)-with-C.N. (out ctx with SS)	52.89	8.03	44.68	76.60	5.76	96.9
27	Orig + P.N. (out ctx) + C.N. (out ctx with SS) Aug	53.34	8.02	44.60	77.05	5.71	96.9
28	Orig + half P.N. (out ctx) (randomly sampled) Aug	53.42	8.04	44.44	76.50	5.74	97.0

Table 2: CB-bi-LSTM (Exp. 01-09), WB-bi-LSTM (Exp. 10-18), FT-byT5 (Exp. 19-28) individual and blended proper noun (P.N.) and common noun (C.N.) augmentation experiments. †, ‡ and ◇ show that the model is statistically significant using *Wilcoxon Test* on all evaluation metrics scores w.r.t. the baselines (Exp. 01, 10 and 19 respectively). All experiments are an average of 5 runs.

Model Type	Data Type	BLEU	NIST	METEOR	ROUGE	CIDEr	BERT Score
CB-bi-LSTM	Gold without Aug	45.42	6.43	38.42	71.70	4.75	95.4
	PN Aug	*50.64*	6.69	*40.67*	*74.22*	*5.22*	*95.9*
	CN Aug	48.70	*6.70*	39.67	73.38	5.03	95.7
Claude-2.0	zero-shot	11.33	3.05	29.39	42.43	1.69	92.3
	few-shot	*27.25*	*5.39*	*38.58*	*64.25*	*3.51*	*95.3*
ChatGPT-3.5	zero-shot	*9.82*	*2.63*	*27.91*	*39.80*	*1.59*	*91.9*
	few-shot	9.58	2.51	26.01	37.40	1.53	91.5
byT5	Gold without Aug	47.55	6.46	42.90	74.56	5.49	96.5
	PN Augmentation	**54.28**	**6.86**	**45.81**	**78.25**	**5.96**	**97.1**
	CN Augmentation	53.04	6.73	45.21	76.97	5.90	96.9

Table 3: Evaluation of DRS-to-Text by LLMs reporting scores for the baseline (without augmentation), ChatGPT 3.5, Claude 2.0, and our best (augmented) models.

both *zero-shot* and *few-shot* learning to analyze the LLMs performance.

To capture performance insights, we considered a sample of 215 sentences from the test set, both (1) on the best neural DRS-to-Text models i.e., CB-bi-LSTM and FT-byT5 (see Table 3), and (2) to the prompt of ChatGPT 3.5 and Claude 2.0 to get model-generated texts (see the exact prompts in the Appendix). We evaluated the output with automatic evaluation metrics scores (see Table 3). The experimental evaluation clearly states that LLMs being general-purpose generative models do not perform well for the low-resource domain-specific task thus highlighting the need for task-specific neural models for the DRS-to-Text generation task.

Error Analysis based Expert Evaluation. Our final evaluation is based on the human evaluation of one expert, who evaluated the generated text by analyzing the model-generated systematic errors in the form of ill-formed semantics, grammaticalization, and phenomenon and produced a ROSE (Robust Overall Semantic Evaluation) score. Table 13 in the Appendix lists some interesting examples generated by our best augmentation model. As defined in (Wang et al., 2021), the ROSE score is the conjunction of three 0-1 evaluation scores: (1) a Boolean *Semantic* measure that checks if the generated text preserves the true meaning w.r.t. to the gold reference; (2) a Boolean *Grammatical* measure that checks if the generated text is not containing any spelling or grammatical errors; (3) a Boolean *Phenomenon* measure that checks if in the generated text the "phenomenon of control is generated at all" w.r.t. to the gold reference (Wang et al., 2021). If the text passes all these three scores (conjunction), it gets a 1 score, otherwise, it gets a 0 score. In Table 4, we have reported the ROSE scores on a sample of 50 sentences from the test set. This evaluation confirms the quality of our best augmentation model in producing good-quality texts, showing the best results in the ROSE measure too.

Implementation	Sem.	Gram.	Phen.	ROSE
Gold-PMB	54%	60%	70%	52%
ChatGPT 3.5	28%	86%	46%	24%
Claude 2.0	34%	86%	44%	34%
C.N. Aug	58%	68%	62%	58%
P.N. Aug	62%	66%	68%	58%
P.N. + C.N. Aug	**64%**	**72%**	**72%**	**62%**

Table 4: Expert evaluation based on Semantics, Grammatical Structure, and Phenomenon for the baseline, ChatGPT, Claude, and our best (augmented) CB-bi-LSTM model.

5 Conclusion

In this paper, we have performed a series of experiments on data augmentation for neural DRS-to-Text generation. Using the PMB dataset, we adopted a novel approach to augment lexical information in DRS for proper and common nouns along with the ability to sustain contextual similarly through *SS* approach on different *in- and out-of*-context transformations. Experimental results reflect both individual and blended implementation scores for our seq-to-seq models (from a training perspective) and Transformer model (from a fine-tuning perspective). The significantly improved results for the char, word, and transformer level models prove the effectiveness and reliability of our proposed approach.

Limitations

As PMB is a multilingual dataset, we have not expanded our implementation on other low-resource languages like *Italian*, *Dutch*, and *French*. We are also exploring other possible augmentation strategies to transform verb phrases.

References

Muhammad Saad Amin, Alessandro Mazzei, and Luca Anselma. 2022. Towards data augmentation for drs-to-text generation. In *Proceedings of the Sixth Workshop on Natural Language for Artificial Intelligence (NL4AI 2022) co-located with 21th International Conference of the Italian Association for Artificial Intelligence (AI*IA 2022), Udine, November 30th, 2022*, volume 3287 of *CEUR Workshop Proceedings*, pages 141–152. CEUR-WS.org.

Laura Banarescu, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Philipp Koehn, Martha Palmer, and Nathan Schneider. 2013. Abstract meaning representation for sembanking. *in Proc.*, 7:178–186.

Valerio Basile and Johan Bos. 2011. Towards generating text from discourse representation structures. *in ENLG'*, 11:145–150.

Johan Bos. 2021. *Quantification annotation in discourse representation theory*. in ISA 2021-17th Workshop on Interoperable Semantic Annotation, Groningen/Virtual, Netherlands.

Massimiliano Ciaramita and Mark Johnson. 2003. Supersense tagging of unknown nouns in wordnet. In *Proc*, pages 168–175. 2003 Conference on Empirical Methods in Natural Language Processing.

Hao Dong, Jingqing Zhang, Douglas McIlwraith, and Yike Guo. 2017. I2t2i: Learning text to image synthesis with textual data augmentation. In *IEEE international conference on image processing (ICIP)*, pages 2015–2019. vol. 2017. IEEE.

Rotem Dror, Gili Baumer, Segev Shlomov, and Roi Reichart. 2018. The hitchhiker's guide to testing statistical significance in natural language processing. *in Proc.*, 56:1383–1392.

Steven Y. Feng, Varun Gangal, Jason Wei, Sarath Chandar, Soroush Vosoughi, Teruko Mitamura, and Eduard Hovy. 2021. A survey of data augmentation approaches for NLP. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 968–988, Online. Association for Computational Linguistics.

Jeffrey Flanigan, Chris Dyer, Noah A Smith, and Jaime G Carbonell. 2016. Generation from abstract meaning representation using tree transducers. *in Proc.*, 2016:731–739.

Claire Gardent, Anastasia Shimorina, Shashi Narayan, and Laura Perez-Beltrachini. 2017. The webnlg challenge: Generating text from rdf data. In *Proceedings of the 10th International Conference on Natural Language Generation*, pages 124–133.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Yutai Hou, Yijia Liu, Wanxiang Che, and Ting Liu. 2018. Sequence-to-sequence data augmentation for dialogue language understanding. arxiv. Preprint.

Katarzyna Jaszczolt. 2023. *Semantics, Pragmatics, Philosophy: A Journey Through Meaning*. New York, NY: Cambridge University Press.

Marcin Junczys-Dowmunt, Kenneth Heafield, Hieu Hoang, Roman Grundkiewicz, and Anthony Aue. 2018. Marian: Cost-effective high-quality neural machine translation in c++. *Arxiv. /abs/*, 1805:12096.

Hans Kamp and Uwe Reyle. 1993. *From Discourse to Logic: Introduction to Modeltheoretic Semantics of Natural Language, Formal Logic and Discourse Representation Theory*. Kluwer Academic Publishers, Dordrecht.

Jiangming Liu, Shay B. Cohen, and Mirella Lapata. 2021. Text generation from discourse representation structures. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 397–415, Online. Association for Computational Linguistics.

Rik van Noord. 2019. *Neural boxer at the IWCS shared task on DRS parsing*. in Proc. IWCS Shared Task on Semantic Parsing, Gothenburg, Sweden. Association for Computational Linguistics[.

OpenAI. 2023. Gpt-4 technical report.

Ankur Parikh, Xuezhi Wang, Sebastian Gehrmann, Manaal Faruqui, Bhuwan Dhingra, Diyi Yang, and Dipanjan Das. 2020. Totto: A controlled table-to-text generation dataset. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1173–1186.

Connor Shorten and Taghi M Khoshgoftaar. 2019. A survey on image data augmentation for deep learning. *Journal of big data*, 6(1):1–48.

Miles Turpin, Julian Michael, Ethan Perez, and Samuel R. Bowman. 2023. Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting.

Rik van Noord, Lasha Abzianidze, Hessel Haagsma, and Johan Bos. 2018. Evaluating scoped meaning representations. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Chunliu Wang, Huiyuan Lai, Malvina Nissim, and Johan Bos. 2023. Pre-trained language-meaning models for multilingual parsing and generation.

Chunliu Wang, Rik van Noord, Arianna Bisazza, and Johan Bos. 2021. Evaluating text generation from discourse representation structures. In *Proceedings of the 1st Workshop on Natural Language Generation, Evaluation, and Metrics (GEM 2021)*, pages 73–83, Online. Association for Computational Linguistics.

Linting Xue, Aditya Barua, Noah Constant, Rami Al-Rfou, Sharan Narang, Mihir Kale, Adam Roberts, and Colin Raffel. 2022. ByT5: Towards a token-free future with pre-trained byte-to-byte models. *Transactions of the Association for Computational Linguistics*, 10:291–306.

Suorong Yang, Wei-Ting Xiao, Mengcheng Zhang, Suhan Guo, Jian Zhao, and Shen Furao. 2022. Image data augmentation for deep learning: A survey. *ArXiv*, abs/2204.08610.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with BERT. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Appendix

In this appendix, we report:

- Hyperparameters for CB-bi-LSTM and WB-bi-LSTM (Table 5)
- Hyperparameters for T5 experiments (Table 6)
- Dataset size (Table 7)
- The Graphical distribution of Named Entities for Proper Noun Augmentation (Figure 3)
- SS-based graphical distribution of Common Noun Entities for Common Noun Augmentation (Figure 4)
- Prompt for ChatGPT-3.5 (Table 8)
- Prompt for Claude-2.0 (Table 9)
- BERT-Score for CB-bi-LSTM (Table 10)
- BERT-Score for WB-bi-LSTM (Table 11)
- BERT-Score for FT-byT5 (Table 12)
- A preliminary error analysis (Table 13)

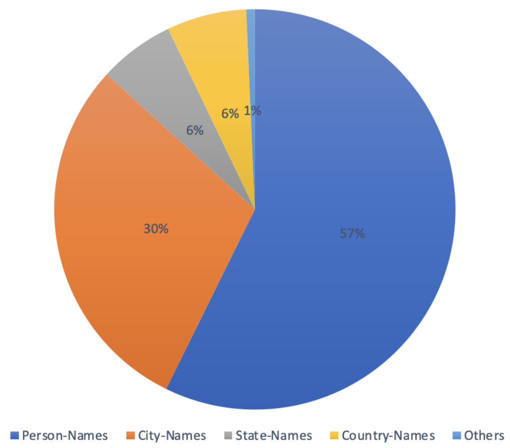

Figure 3: Distribution of proper noun entities in Gold-PMB dataset.

HyperParameters	Values
Embedding Dimensions	300
Enc/Dec Cell	LSTM
Enc/Dec Depth	2
Mini-batch	48
Normalization Rate	0.9
lr-decay	0.5
lr-decay-strategy	Epoch
Optimizer	Adam
Validation Metric	Cross-Entropy
Cost-Type	ce-mean
Beam Size	10
Learning Rate	0.002

Table 5: Hyperparameter settings for CB-bi-LSTM and WB-bi-LSTM.

HyperParameters	Values
Batch size	15
Update steps	8
Max learning Rate	1e-4
Min learning Rate	1e-5
Warmup updates	3000
Max decay steps	30000
No. of epochs	15
Optimizer	AdamW

Table 6: Hyperparameter settings for FT-byT5.

Transformation Type	Size	Examples
Orig Training Examples	x1	6620
Orig + P.N. Aug	x2	13240
Orig + C.N. Aug	x2	13240
Orig + P.N.-with-C.N. Aug	x2	13240
Orig + P.N. + C.N. Aug	x3	19860
Validation Examples		885
Test Examples		898

Table 7: Dataset size ('+' indicates augmentation applied in individual form and '-' in blended form).

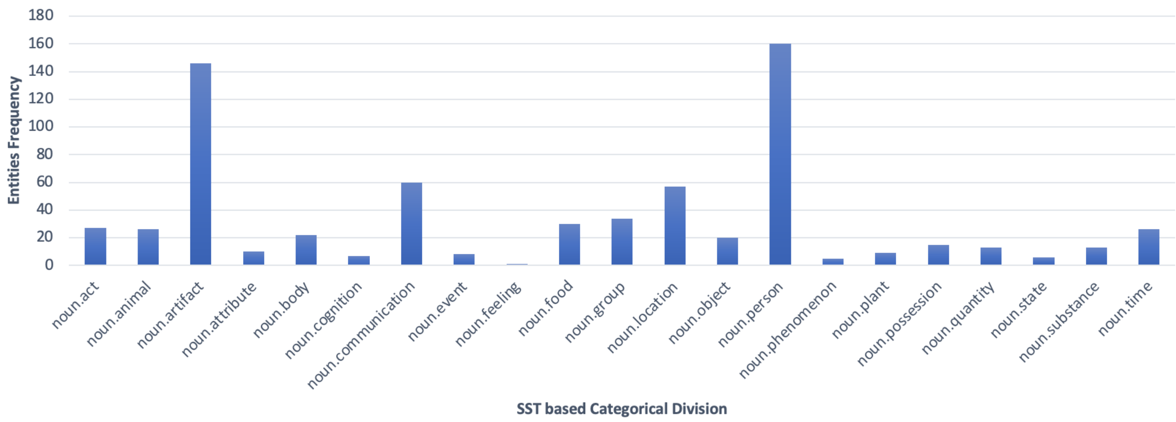

Figure 4: SS tagging based categorical division of common noun entities in Gold-PMB dataset.

Prompt
Discourse Representation Structure (DRS) refers to a formal representation used to capture the meaning of a sentence or a piece of text in a way that can be processed and analyzed by computational systems. Given a clausal format of DRS, can you generate an English translation of the given DRS?
ChatGPT Response
Absolutely, I can assist you in generating English translations from a clausal format of Discourse Representation Structure (DRS). Please provide me with the specific DRS you have in mind, and I'll do my best to provide you with the corresponding English translation.
Prompt
b2 REF s1 b2 Attribute x1 s1 b2 Value s1 "?" b2 far "a.01" s1 b2 REF t1 b2 EQU t1 "now" b2 Time s1 t1 b2 time "n.08" t1 b1 REF x1 b1 PRESUPPOSITION b2 b1 entity "n.01" x1
ChatGPT Response
The provided Discourse Representation Structure (DRS) corresponds to the following English sentence: "Is it far now in time?" Explanation: The lines in the DRS represent the following: b2 REF s1 introduces a reference to a situation "s1." b2 Attribute x1 s1 attributes "x1" (later determined to be "it") to situation "s1." b2 Value s1 "?" indicates that the value of situation "s1" is a question mark, implying an interrogative context. b2 far "a.01" s1 specifies that situation "s1" is associated with being "far" (distant). b2 REF t1 introduces a reference to a time "t1." b2 EQU t1 "now" equates time "t1" with "now," indicating the present time. b2 Time s1 t1 associates situation "s1" with time "t1." b2 time "n.08" t1 relates the concept of time "t1" to the noun "time." b1 REF x1 introduces a reference to an entity "x1." b1 PRESUPPOSITION b2 establishes a presupposition relationship with "b2." b1 entity "n.01" x1 indicates that "x1" is an entity, often referred to as "it." Putting it all together, the English translation is "Is it far now in time?" This DRS representation and its translation capture the structure and meaning of the given sentence.
Prompt
The translation should be without explanation. Just one line English sentence.
ChatGPT Response
My time is ?.
Actual Translation
How far is it?

Table 8: Example of a prompt and the related response from ChatGPT 3.5 for DRS-to-Text generation.

Prompt
Discourse Representation Structure (DRS) refers to a formal representation used to capture the meaning of a sentence or a piece of text in a way that can be processed and analyzed by computational systems. Given a clausal format of DRS, can you generate an english translation of the given DRS?
Claude Response
Sure, here is an example of generating an English translation from a clausal DRS format: Given DRS: x, y, e book(x) person(y) own(e, y, x) English translation: "There is a book x and a person y such that y owns x." The key aspects are: The discourse referents x, y, e represent objects or events introduced in the DRS conditions. The conditions like book(x) and person(y) provide information about these referents. own(e, y, x) represents a relation between the referents. These conditions can be translated into English by referring back to the introduced referents and expressing the relations between them. Additional discourse context like tense, temporal relations, modality etc. can also be represented in the DRS and translated appropriately. So in essence, I would analyze the DRS conditions and referents, identify the objects, events and relations they represent, and generate corresponding English sentences to verbalize those meanings. Let me know if you need any other examples of translating DRS representations into natural language!
Prompt
b2 REF s1 b2 Attribute x1 s1 b2 Value s1 "?" b2 far "a.01" s1 b2 REF t1 b2 EQU t1 "now" b2 Time s1 t1 b2 time "n.08" t1 b1 REF x1 b1 PRESUPPOSITION b2 b1 entity "n.01" x1
Claude Response
Here is my attempt at translating the given DRS into English: There is a referent s1. s1 has the attribute x1. The value of s1 is "?". s1 is far away "a.01". There is a referent t1. t1 is equated to "now". s1 occurs at time t1. t1 is the present time "n.08". There is a referent x1. x1 presupposes referent b2. x1 is an entity "n.01". In plain English: There is something s1. It has some attribute x1 and its value is unknown (?). s1 is far away. There is a time t1 which is the present. s1 occurs at this present time. There is something x1 which presupposes something else b2. x1 is an entity. Let me know if this translation captures the meaning represented in the DRS or if you have any other feedback! I'm still learning how to translate from DRS to natural language.
Prompt
The translation should be without explanation. Just one line English sentence.
Claude Response
Here is a one line English translation of the given DRS: There is a far away entity with an unknown attribute occurring now which presupposes something else.
Actual Translation
How far is it?

Table 9: Example of a prompt and its related response from Claude 2.0 for DRS-to-Text generation.

Exp. No	Implementation Type	Precision	Recall	F1-Score
01	Gold (without augmentation)	95.2	95.4	95.3
02	Gold + PN (inside context)	95.8	95.9	95.8
03	Gold + PN (outside context)	95.9	95.9	95.9
04	Gold + CN (inside context with SS)	95.7	95.7	95.7
05	Gold + CN (inside context without SS)	95.7	95.5	95.6
06	Gold + CN (outside context with SS)	95.5	95.8	95.7
07	Gold + CN (outside context without SS)	95.8	95.7	95.7
08	Gold + PN-with-CN	96.1	95.9	96.0
09	Gold + PN + CN	96.1	96.1	96.1

Table 10: BERT-Score for all Char-level-based implementation of augmentation experiments with Precision, Recall, and F1-Score.

Exp. No	Implementation Type	Precision	Recall	F1-Score
01	Gold (without augmentation)	94.6	94.3	94.4
02	Gold + PN (inside context)	95.2	95.0	95.1
03	Gold + PN (outside context)	95.0	94.8	94.9
04	Gold + CN (inside context with SS)	95.2	95.0	95.1
05	Gold + CN (inside context without SS)	94.9	94.7	94.8
06	Gold + CN (outside context with SS)	94.6	94.5	94.6
07	Gold + CN (outside context without SS)	95.0	94.7	94.8
08	Gold + PN-with-CN	95.1	94.9	95.0
09	Gold + PN + CN	95.3	95.1	95.2

Table 11: BERT-Score for all Word-level-based implementation of augmentation experiments with Precision, Recall, and F1-Score.

Exp. No	Implementation Type	Precision	Recall	F1-Score
01	Gold (without augmentation)	96.6	96.9	96.7
02	Gold + PN (inside context)	96.9	97.3	97.1
03	Gold + PN (outside context)	97.0	97.4	97.2
04	Gold + CN (inside context with SS)	96.6	97.1	96.8
05	Gold + CN (inside context without SS)	96.5	97.1	96.8
06	Gold + CN (outside context with SS)	96.8	97.2	97.0
07	Gold + CN (outside context without SS)	96.6	97.1	96.8
08	Gold + PN-with-CN	96.6	97.2	96.9
09	Gold + PN + CN	96.6	97.2	96.9
10	Gold + half (randomly sampled) P.N. (outside context) Aug	96.7	97.2	97.0

Table 12: BERT-Score for all byT5-based implementation of augmentation experiments with Precision, Recall, and F1-Score.

Reference Text	Generated Text	Sem.	Gram.	Phen.	ROSE
I am milking my goat.	I'm milking my squirrel.	0	1	1	0
The train leaves at 2:30 p.m.	The train leaves at 2:30.	0	0	1	0
We arrived two days ago.	I arrived two days ago.	1	1	1	1
Three times five is fifteen.	3 times 5 is 15.	1	1	1	1
An elephant has a long nose.	The elephant has a long nose.	1	1	1	1

Table 13: Error analysis of model-generated examples w.r.t reference text for ROSE evaluation.