Referring to what you know and do not know: Making Referring Expression Generation Models Generalize To Unseen Entities

Rossana Cunha, Thiago Castro Ferreira, Adriana Silvina Pagano, and Fabio Alves
Arts Faculty, Federal University of Minas Gerais (UFMG), Brazil
{rossanacunha,thiagocf05,apagano,fabio-alves}@ufmg.br

Abstract

Data-to-text Natural Language Generation (NLG) is the computational process of generating natural language in the form of text or voice from non-linguistic data. A core micro-planning task within NLG is referring expression generation (REG), which aims to automatically generate noun phrases to refer to entities mentioned as discourse unfolds. A limitation of novel REG models is not being able to generate referring expressions to entities not encountered during the training process. To solve this problem, we propose two extensions to NeuralREG, a state-of-the-art encoder-decoder REG model. The first is a copy mechanism, whereas the second consists of representing the gender and type of the referent as inputs to the model. Drawing on the results of automatic and human evaluation as well as an ablation study using the WebNLG corpus, we contend that our proposal contributes to the generation of more meaningful referring expressions to unseen entities than the original system and related work. Code and all produced data are publicly available[1][2].

1 Introduction

Data-to-text Natural Language Generation (NLG) is the computational process of generating natural language in the form of text or voice from non-linguistic data. A traditional micro-planning task within the pipeline data-to-text architecture is referring expression generation (REG) (Krahmer and van Deemter, 2019), which aims to automatically generate appropriate noun phrases (e.g., *The mathematician Ada Lovelace*) to refer to entities (e.g., `Ada_Lovelace`) mentioned as discourse unfolds (e.g., *"___ was the first to recognise that the machine had applications beyond pure calculation."*).

Traditionally, REG systems produce references to discourse entities in two explicit steps. First, they decide on the referential form, i.e., choosing whether a referring expression should be a pronoun (*She*), a proper name (*Ada Lovelace*), a description (*The mathematician*), etc. Once the choice is made, such systems textually realize the referring expression based on the chosen referential form and discourse context. If the first step selects a proper name as the form to refer to `Ada_Lovelace` for instance, the ensuing step is responsible for deciding, among *Ada*, *Ada Lovelace*, or another text realization of a proper name, i.e., the one that is the most appropriate referring expression to that entity in a given discourse context.

With the advent of large amounts of data, REG systems have undergone a significant change in their architecture. From being rule-based modular, they have become data-driven end-to-end systems that aim to perform the choice of referential form and surface realization jointly. An example of these more integrated approaches is NeuralREG (Castro Ferreira et al., 2018a), an end-to-end neural REG model that produces referring expressions deciding on form and content jointly based on representations of the referent and its surrounding context.

[1] https://github.com/rossanacunha/NeuralREG
[2] https://github.com/ThiagoCF05/NeuralREG/tree/improvements

Proceedings of the 28th International Conference on Computational Linguistics, pages 2261–2272
Barcelona, Spain (Online), December 8-13, 2020

Although NeuralREG is able to generate adequate referring expressions to discourse entities already seen during the training phase, the model does not generalize to unseen ones, i.e., it can not generate referring expressions to entities which were not seen during its training. This study aims to fill this gap by proposing two extensions to the model's original architecture. The first is a copy mechanism, which may decide at each decoding timestep whether the next token of the referring expression should be generated from the output vocabulary or copied from the input representation of the target entity. We thereby hypothesize that the model will be able to generate a token from the vocabulary for seen entities and to copy tokens from the input representation in the case of unseen ones. The second extension consists of representing the gender and type of the entity as input to the model. Such information can be easily extracted from the Semantic Web and may help the model to generate pronominal (e.g., *She*) and descriptive (e.g., *The country*) referring expressions to unseen entities.

To evaluate our approach, we conducted experiments relying on a delexicalized version (Castro Ferreira et al., 2018b) of the WebNLG corpus (Gardent et al., 2017b). We first compare our proposal with the original NeuralREG and other related approaches as ProfileREG (Cao and Cheung, 2019). Second, to assess the quality of the texts generated by our model, we conducted a supplementary evaluation with human judges. Next, we follow the rationale of ablation studies to analyze the importance of each feature in our model within the process of referring expression generation. Finally, we discuss some advantages of the introduced features and how they interact to improve accuracy, variety, and generalization.

2 Related work

Given an entity to be referred to in a particular context, traditional REG methods have addressed this task in two steps. The first one concerns the choice of referential form, i.e., deciding whether the target reference is more likely to be a proper name (*Belo Horizonte*), a description (*The city*), a pronoun (*It*), or another referential form. Regarding this step, Reiter and Dale (2000) suggested to always choose a full proper name as the first reference to a particular entity in a given context, whereas pronouns may be used for its subsequent references if there is no other entity with the same person, gender and number in-between the target reference and its antecedents. More recently, Castro Ferreira et al. (2016) proposed a naive Bayes method, which is able to non-deterministically choose a referential form to a particular reference. The model's choice is conditioned upon discourse features which studies in psycholinguistics have shown to impact this choice, such as grammatical position, givenness and recency of the target reference.

Once the referential form is chosen, the second step of traditional REG models focuses on the surface realization of the reference. Most part of the literature on this step focuses on the generation of descriptions (Dale and Reiter, 1995) although some studies have approached the generation of proper names (Siddharthan et al., 2011; van Deemter, 2016; Castro Ferreira et al., 2017).

In contrast to previous proposals that have focused on selecting referential form or referential content, Castro Ferreira et al. (2018a) proposed an end-to-end approach: NeuralREG, a referring expression generator able to perform the choice of referential form and the surface realization in an end-to-end style using a neural encoder-decoder architecture. Given an entity to be referred to in a particular textual context, the approach first encodes the entity identifier and the text prior (pre-context) and subsequent to the reference (post-context) to later decode this representation into an appropriate referring expression using attention (Bahdanau et al., 2015).

Although NeuralREG (Castro Ferreira et al., 2018a) can generate suitable referring expressions to entities seen during training, it presents certain problems when referring to unseen ones. To overcome this limitation, Cao and Cheung (2019) presented a profile based model. Their solution uses information from both profile[3] (i.e., information retrieved from the entity's Wikipedia page) and context (pre- and post-contexts jointly) to generate suitable references to unseen entities. The authors conclude that their approach is more successful to determine the most suitable referring expression to a particular entity.

In contrast to Cao and Cheung (2019) solution, in order to address the limitations of dealing with unseen relations and entities, our proposal uses a combination of a copy mechanism together with rep-

[3] https://en.wikipedia.org/wiki/Ada_Lovelace

Triples		
Subject	**Predicate**	**Object**
Adenan_Satem	**birthPlace**	**Japanese_occupation_of_British_Borneo**
Abdul_Taib_Mahmud	successor	Adenan_Satem
Abdul_Taib_Mahmud	residence	Sarawak
Abdul_Taib_Mahmud	party	"Barisan Raayat Jati Sarawak"

Corresponding Text
Adenan Satem was born in **Japanese Occupied British Borneo**. His successor was Abdul Taib Mahmud, who, resides in Sarawak and is a member of the *"Barisan Raáyat Jati Sarawak"* party.

Tag	Entity/Constant	Referring Expression
BRIDGE-1	Adenan_Satem	Adenan Satem
PATIENT-1	Japanese_occupation_of_British_Borneo	Japanese Occupied British Borneo
BRIDGE-1	Adenan_Satem	His
AGENT-1	Abdul_Taib_Mahmud	Abdul Taib Mahmud
PATIENT-2	Sarawak	Sarawak
PATIENT-3	"Barisan Raayat Jati Sarawak"	Barisan Ra'ayat Jati Sarawak

Template
BRIDGE-1 was born in PATIENT-1. BRIDGE-1 successor was AGENT-1, who, resides in PATIENT-2 and is a member of the PATIENT-3 party.

Table 1: A set of RDF triples and their corresponding text. Followed by entities mapping and a delexicalized template.

resentations of gender and type of referent as input to the model. We expect both extensions to make the model able to produce suitable referring expressions in particular to unseen entities. We describe our model in more detail in the next sections, based on the data used to investigate and evaluate our approach.

3 Data

We evaluated our proposal based on an enriched version (Castro Ferreira et al., 2018b) of the WebNLG corpus (Gardent et al., 2017a). The original resource is a parallel corpus with sets of RDF (Resource Description Framework) triples and their corresponding verbalizations. Each RDF triple set consists of subject-predicate-object (e.g., `Adenan_Satem | birthPlace | Japanese_occupation_of_British_Borneo`), which is illustrated in Table 1 and can be verbalized in different forms. Each subject and object is a *Uniform Resource Identifier* (URI), which can be represented by a Wikipedia ID (e.g., `Adenan_Satem`, `Abdul_Taib_Mahmud`) or a literal value like a date, number, or constant (e.g., "`Barisan Raayat Jati Sarawak`"), followed by a predicate (e.g., `birthPlace`) which is a relation between these entities (Gardent et al., 2017a; Gardent et al., 2017b).

The WebNLG dataset is an NLG benchmark that differs from other datasets (Novikova et al., 2017; Mille et al., 2018) due to its data diversity in terms of attributes, patterns, and shapes (i.e., RDF tree shapes from DBPedia). The corpus contains 25,298 English texts verbalizing sets of 1 to 7 RDF triples in 15 different domains. The dataset has five domains exclusive to the test set, providing adequate means to evaluate our model's performance regarding the generation of referring expressions to unseen entities.

We used an enriched version of the WebNLG corpus obtained by a delexicalization process (i.e., mapping each entity to a generic tag and later replacing their corresponding referring expressions in discourse with these tags) which was created by Castro Ferreira et al. (2018b). Table 1 shows an example of a set of 4 triples and corresponding text, together with the intermediate representations obtained in the delexicalization process, such as general tags, Wikipedia IDs (entity/constant), referring expressions and the delexicalized template.

To train and evaluate our approach, we have a pre-processing stage where we extract a collection of referring expression entries from the enriched version of WebNLG (Castro Ferreira et al., 2018b). This stage is performed once, where we map the WebNLG corpus information as the basis to obtain external information without adding new features nor changing the WebNLG structure. We avoid a possible impact on the evaluation results since all entities are available on DBPedia. Each entry consists of a Wikipedia ID, i.e., a target entity (`Adenan_Satem`), a truecased tokenized referring expression (`Adenan Satem` or `His`), and lowercased tokenized pre- (`Adenan_Satem was born in`) and post-contexts (`Adenan_Satem successor was Abdul_Taib_Mahmud, who, resides`

in Sarawak and is a member of the barisan raayat jati sarawak party.), indicating the surrounding context of the target reference.

4 Model

Our approach was based on NeuralREG (Castro Ferreira et al., 2018a) and aims to generate a referring expression $y = \{y_1, y_2, ..., y_N\}$ with N tokens to refer to a target entity, given the textual context prior to the reference $X^{(pre)} = \{x_1^{(pre)}, x_2^{(pre)}, ..., x_M^{(pre)}\}$ with M tokens (e.g., pre-context) and subsequent to the reference $X^{(post)} = \{x_1^{(post)}, x_2^{(post)}, ..., x_L^{(post)}\}$ with L tokens (e.g., post-context). Unlike Castro Ferreira et al. (2018a), where the target entity is represented by a single token, our approach describes the referent by an identifier $X^{(wiki)} = \{x_1^{(wiki)}, x_2^{(wiki)}, ..., x_T^{(wiki)}\}$ with T tokens and its entity type E and gender G. To generate the referring expression given the description of the target entity and its surrounding context, we implemented an *encoder-attention-decoder* architecture with a copy mechanism, sharing the same input word-embedding matrix V, as explained in the following sections.

4.1 Encoder

In order to generate feature representations for the inputs, the model starts by encoding the identifier of the target entity as well as the pre- and post-contexts, using three different bidirectional Long-Short Term Memory layers (LSTM) (Hochreiter and Schmidhuber, 1997). The identifier of the target entity $X^{(wiki)} = \{x_1^{(wiki)}, x_2^{(wiki)}, ..., x_T^{(wiki)}\}$ is represented by the forward and backward hidden-state vectors $(\overrightarrow{h}_1^{(wiki)}, \cdots, \overrightarrow{h}_m^{(wiki)})$ and $(\overleftarrow{h}_1^{(wiki)}, \cdots, \overleftarrow{h}_m^{(wiki)})$. To form its final feature representation, forward and backward hidden-state representations at each timestep t are concatenated as $h_t^{(wiki)} = [\overrightarrow{h}_t^{(wiki)}, \overleftarrow{h}_t^{(wiki)}]$. Using the two remaining bidirectional LSTMs, the same process is repeated for the textual context surrounding the reference, resulting in the final pre- and post-context representations $([\overrightarrow{h}_1^{(pre)}, \overleftarrow{h}_1^{(pre)}], \cdots, [\overrightarrow{h}_m^{(pre)}, \overleftarrow{h}_m^{(pre)}])$ and $([\overrightarrow{h}_1^{(post)}, \overleftarrow{h}_1^{(post)}], \cdots, [\overrightarrow{h}_m^{(post)}, \overleftarrow{h}_m^{(post)}])$, respectively. Finally, the type and gender of the target entity is encoded into their respective vector representations, V_{type} and V_{gender}, by looking up their entry in the sharing word-embedding matrix V.

4.2 Decoder

Once the information about the target entity and its surrounding contexts are encoded, their vector representations are fed into an LSTM decoder, augmented with attention and copy mechanisms, in order to produce an adequate referring expression to the target entity according to the context. The process is explained in detail in the following sections.

Attention Mechanism The decoder process starts by the attention mechanism (Bahdanau et al., 2015), which aims to compute a vector c_t at each timestep t. The mechanism first computes the energies $e_{tj}^{(wiki)}$, $e_{tj}^{(pre)}$ and $e_{tj}^{(post)}$ based on the encoder states $h_t^{(wiki)}$, $h_t^{(pre)}$ and $h_t^{(post)}$, together with the decoder state s_{t-1}. The softmax function is then applied over these energies, resulting in the final attention probabilities $\alpha_t^{(wiki)}$, $\alpha_t^{(pre)}$ and $\alpha_t^{(post)}$. Equations 1 and 2 show the computation of the energies and final attention probabilities, where $k \in \{wiki, pre, post\}$ and the matrices $W_a^{(k)}$ and $U_a^{(k)}$ as well as the attention vectors $v_a^{(k)}$ are training parameters.

$$e_{tj}^{(k)} = v_a^{(k)T} \tanh(W_a^{(k)} s_{t-1} + U_a^{(k)} h_j^{(k)}) \tag{1}$$

$$\alpha_{tj}^{(k)} = \frac{\exp(e_{tj}^{(k)})}{\sum_{n=1}^{N} \exp(e_{tn}^{(k)})} \tag{2}$$

At each decoding step t, a final context vector $c_t^{(k)}$ is computed based on the sum of the encoder states $h_t^{(k)}$ weighed by the attention probabilities $\alpha_{tj}^{(k)}$, as the following equation expresses:

$$c_t^{(k)} = \sum_{j=1}^{N} \alpha_{tj}^{(k)} h_j^{(k)} \tag{3}$$

Finally, in order to obtain the final context vector c_t, we follow the concatenative approach of NeuralREG, where the attention vectors $c_t^{(wiki)}$, $c_t^{(pre)}$ and $c_t^{(post)}$ are simply concatenated, such as $c_t = [c_t^{(wiki)}, c_t^{(pre)}, c_t^{(post)}]$.

Decoding After attending the representations of the target entity and its surrounding contexts, the resulting attention vector c_t is concatenated with the previous decoding state s_{t-1}, the word-embedding of the previous generated token V_{y-1} and the vector representations of the type and gender of the target entity, V_{type} and V_{gender}. This concatenation is then fed into the decoding layer, which produces its next state s_t. Finally, a softmax layer is applied over the decoding state s_t to generate a probability distribution over the output vocabulary. Equations 4, 5 and 6 summarize this process:

$$s_t = \Phi_{\text{dec}}(s_{t-1}, [c_t, V_{y_{t-1}}, V_{type}, V_{gender}]) \tag{4}$$

$$z_t = W_b s_t + b \tag{5}$$

$$P_{vocab}(w) = \frac{\exp(z_{ti})}{\sum_{j=1}^{J} \exp(z_{tj})} \tag{6}$$

Copy Mechanism To make the approach able to generate referring expressions to unseen entities, we also implemented a copy mechanism during the decoding process, similar to the one presented by See et al. (2017). This mechanism first computes a probability p_{gen} based on the attention vector of the target entity $c_t^{(wiki)}$, the decoding state s_{t-1} and the word-embedding of the previously generated token $V_{y_{t-1}}$, as the following equation expresses:

$$p_{gen} = \text{sigmoid}(W_c c_t^{(wiki)} + W_d s_{t-1} + W_e V_{y_{t-1}} + b) \tag{7}$$

p_{gen} is used to decide between (1) choosing the token with the highest probability in the softmax probability distribution $P_{vocab}(w)$ in Equation 6 or (2) copying the token from the description of the entity $X^{(wiki)}$ with the highest probability according to the attention weights $\alpha_t^{(wiki)}$. The final probability distribution to choose the next token at each timestep t is given by the following Equation:

$$P(w) = p_{gen} P_{vocab}(w) + (1 - p_{gen}) \sum_{i:w_i=w} \alpha_{ti}^{(wiki)} \tag{8}$$

In this context, we expect the model to learn that p_{gen} should have a higher value when the target entity was seen during training, and a lower one when a referring expression should be generated for an unseen entity.

Loss During training time, the approach has its training parameters updated in order to minimize the following loss function:

$$J(\theta) = -\sum_t P(y_t)) \tag{9}$$

5 Automatic Evaluation

5.1 Data

We used the delexicalized version of the WebNLG corpus described in Section 3. In particular, we used version 1.5 of the corpus, which is publicly available[4]. This version of the corpus contains 67,027, 8,278 and 19,210 referring expression instances in training, development and test sets, respectively. Training and development domains have instances of 10 semantic domains, whereas the test set has instances of those 10 domains, plus 5 unseen ones in the former sets.

Each instance of the sets is formed by the target entity, a referring expression, and pre- and post-contexts. Pre- and post-contexts are represented in their lowercased and tokenized forms, whereas the referring expression in its truecased and tokenized one. Moreover, references to different discourse entities are represented by their Wikipedia IDs. In contrast, numbers, dates, and other constants are

[4] https://github.com/ThiagoCF05/webnlg

represented by one-word ID replacing white spaces with underscores and eliminating double-quotes (Castro Ferreira et al., 2018a; Castro Ferreira et al., 2018b). To represent the target entity $X^{(wiki)}$ as described in Section 4, we lowercase the Wikipedia ID of the target entity, remove all special characters and split it in a list based on underscores (e.g., `Abdul_Taib_Mahmud` → `[abdul, taib, mahmud]`). Accordingly, all target entities' gender (female, male, neutral) and type (person, organization, etc.), used by our approach, were automatically retrieved from DBpedia[5].

5.2 Model Settings

Regarding the model parameters, we followed most of the settings from NeuralREG's set-up of Castro Ferreira et al. (2019). We trained the model with 60 epochs with a dropout of 0.2. Furthermore, we set the early stopping of the neural networks to 10 and the beam size to 1. We applied a maximum output limit generation of 30. Moreover, we set the batch, state, and attention sizes to 80, 256, and 256, respectively. Additionally, we set pre-context, post-context, and entity word embeddings to be 128D each.

5.3 Baselines

We compared our proposal (*NeuralREG+Copy*) against three baselines: the concatenative attention version of the original model *NeuralREG*, OnlyNames (Castro Ferreira et al., 2016), and ProfileREG (Cao and Cheung, 2019).

OnlyNames correlates an entity that will be referred to by its Wikipedia ID. This baseline exclusively works with proper names by replacing entities underscores with white spaces (e.g., Ada_Lovelace to "Ada Lovelace"). Instead of working exclusively with proper names, our approach implements other referential forms, such as pronouns and descriptions, consequently yielding a more natural discourse flow in the texts produced.

`NeuralREG+Catt` is as an end-to-end deep neural network model that uses both form and content to generate texts. The model works with a delexicalized version of the WebNLG corpus by first encoding pre- and post-contexts as a reference. In contrast to our proposal, `NeuralREG+Catt` does not implement a copy mechanism and does not consider any external knowledge when selecting the best referring expression.

ProfileREG encodes information from a local context and an external profile to generate references to a given entity. This model is able to determine the best reference to an entity by selecting from existing vocabulary, pronouns, or entity profile. Contrary to ProfileREG, our model uses selected entity features and different architectures in order to evaluate the best scenario for generating referring expressions.

5.4 Metrics

We calculated Accuracy and String Edit Distance (Levenshtein, 1966) in order to measure the quality of the generated referring expressions in comparison with the gold-standard ones. To evaluate the models' performance in realizing pronouns, we also computed the accuracy, precision, recall, and F1-score, based on a concise difference between the gold-standard referring expressions to the ones produced by the model. Finally, we compared the original texts against the references lexicalized through the models by computing text accuracy and BLEU score (Papineni et al., 2002).

5.5 Results

Table 2 presents the results of our model in comparison with the baselines for *all* entities as well as for *seen* and *unseen* ones. In terms of referring expression accuracy, string edit distance, text accuracy, and BLEU score, our proposed approach outperforms the three baselines considering *all* entities and *seen* ones only. Regarding *unseen* entities, our model presents higher results for the same metrics in comparison with all models, except for OnlyNames one. Regarding pronouns, ProfileREG introduces

[5] `http://dbpedia.org/page/Ada_Lovelace`

Entities	Model	RE Acc.	SED	BLEU	Txt Acc.	Precision	Recall	F1-score
All	OnlyNames	0.51	4.21	65.48	0.14	-	-	-
	NeuralREG	0.38	9.86	48.10	0.11	0.72	0.63	0.67
	NeuralREG+Copy	**0.59**	**3.53**	**66.14**	**0.16**	0.79	0.53	0.63
	ProfileREG	0.42	7.05	54.08	0.09	**0.82**	**0.86**	**0.84**
Seen	OnlyNames	0.53	4.32	66.39	0.16	-	-	-
	NeuralREG	0.70	3.07	70.21	0.20	0.78	0.76	0.77
	NeuralREG+Copy	**0.73**	**2.50**	**71.74**	**0.24**	**0.79**	0.71	0.75
	ProfileREG	0.69	3.11	69.11	0.17	**0.79**	**0.90**	**0.84**
Unseen	OnlyNames	**0.50**	**4.10**	**63.97**	**0.11**	-	-	-
	NeuralREG	0.07	16.71	25.42	0.00	0.67	0.55	0.61
	NeuralREG+Copy	0.46	4.57	59.15	0.08	0.79	0.42	0.54
	ProfileREG	0.14	11.03	36.45	0.00	**0.84**	**0.83**	**0.84**

Table 2: (a) Referring Expressions' Accuracy and String-edit distance (SED), (b) BLEU and Text Accuracy scores of the models, (c) Pronoun - Precision, Recall, and F1-Score of the models in the automatic evaluation. Best results are **boldfaced**, whereas the second best are underlined.

Original: **Adenan Satem** was born in **Japanese Occupied British Borneo**. **His** successor was **Abdul Taib Mahmud**, who, resides in **Sarawak** and is a member of the "*Barisan Raáyat Jati Sarawak*" party.
OnlyNames: **Adenan Satem** was born in **Japanese Occupied British Borneo**. **Adenan Satem** successor was **Abdul Taib Mahmud**, who, resides in **Sarawak** and is a member of the "*Barisan Raáyat Jati Sarawak*" party.
NeuralREG: **The Boeing light combat** was born in **Abilene, in Texas**. **They** successor was **the hal of the astronaut**, who, resides in **the state of Grenada** and is a member of the "*Barisan Raáyat Jati Sarawak*" party.
NeuralREG+Copy: **Adenan Satem** was born in the **Japanese Occupied British Borneo**. **His** successor was **Abdul Taib Mahmud**, who, resides in **Sarawak** and is a member of "*Barisan Raáyat Jati Sarawak*" party.
ProfileREG: **258.2 Satem** was born in the **Japanese**. **Its** successor was the **Taib the Moro**, who, resides in the **Sarawak** and is a member of the "*Barisan Raáyat Jati Sarawak*" party.

Table 3: Sample outputs of an unseen domain (Politician) - original and generated text of each model. Referring expressions are boldfaced, and constants are double-quoted and italicized.

the best results, while the OnlyNames model is not considered, since this model is not able to generate this form of reference.

Table 3 shows an example of a text lexicalized with referring expressions generated by our proposal and the three baselines. The text was extracted from the test set of the data in the Politician domain, not present in the training and development sets. By comparing our approach (*NeuralREG+Copy*) to the baseline OnlyNames, we can see that our model is able to generate more variation in referring mechanisms since it makes use of a pronoun as a referential form, while OnlyNames uses repetition of proper names. The outputs for the `Adenan_Satem` for NeuralREG and ProfileREG models show generation problems, namely entities completely unrelated to the references (e.g., `The Boeing light combat` and `258.2 Satem`, respectively).

6 Human Evaluation

To assess the quality of the texts generated by our proposal and the three baselines, we conducted a supplementary evaluation with human judges.

Method Two applied linguists were recruited to rate the texts. They are proficient in English and have over 20 years' expertise as translators and language advisers.

We selected 75 instances of the delexicalized version of the WebNLG corpus, considering a unique occurrence for each combination between the number of triples (ranging from 1 to 7) and domain (10 seen and 5 unseen ones). After selecting the set of triples, we collected the corresponding produced versions of each investigated model introduced in this study (our proposal and three baselines). Finally, we randomly ordered the final trial set of ($4 \times 75 =$) 300 sentences to decrease the bias of having the 4 generated texts together during the evaluation.

The performed evaluation followed the best practices suggested by Van der Lee et al. (2019) and the guidelines in Novikova et al. (2018) regarding human evaluations of NLG systems. For instance, we

Metric	Model	All	Seen	Unseen
Fluency	OnlyNames	**4.03 ±0.16**	**4.05 ±0.18**	**3.95 ±0.36**
	NeuralREG	3.31 ±0.30	3.91 ±0.21	1.53 ±0.40
	NeuralREG+Copy	4.00 ±0.16	**4.05 ±0.17**	3.84 ±0.39
	ProfileREG	3.53 ±0.26	3.87 ±0.21	2.53 ±0.64
Grammaticality	OnlyNames	3.99 ±0.14	**4.05 ±0.16**	3.79 ±0.28
	NeuralREG	3.36 ±0.29	3.92 ±0.22	1.71 ±0.34
	NeuralREG+Copy	**4.02 ±0.16**	**4.05 ±0.18**	**3.92 ±0.32**
	ProfileREG	3.51 ±0.24	3.83 ±0.20	2.55 ±0.59
Semantic Adequacy	OnlyNames	**4.85 ±0.10**	**4.88 ±0.08**	**4.76 ±0.34**
	NeuralREG	3.70 ±0.37	4.50 ±0.24	1.34 ±0.25
	NeuralREG+Copy	4.63 ±0.15	4.75 ±0.14	4.29 ±0.44
	ProfileREG	3.89 ±0.32	4.46 ±0.24	2.18 ±0.55

Table 4: Mean and $\pm 95\%$ Confidence Intervals on Fluency, Grammaticality, and Semantic Adequacy results for All, Seen, and Unseen entities of the Human Evaluation. Best results are **bolded**, whereas the second best are underlined.

chose well-defined criteria to assess text quality and a well-established scale for assessment. The participants were asked to rate the automatically generated sentences with respect to three criteria: fluency, i.e., whether the text flow was acceptable; grammaticality, i.e., whether grammatical and lexical patterns were close to human language patterns; and semantic adequacy, i.e., whether the information in the output text matched that of the input representation. In addition, a 5 point Likert scale was used (1 – very low, 2 – low, 3 – medium, 4 – high, and 5 – highly/fully adequate).

Results Table 4 summarizes the results of human evaluation regarding fluency, grammaticality, and semantic adequacy for all, seen, and unseen entities. Our proposed model outperformed the previous version of NeuralREG and presented competitive results compared to the current state-of-the-art in the literature. Regarding grammaticality, our model presents the best results for all, seen, and unseen entities considering the three baselines. Regarding fluency and semantic adequacy, human evaluation showed similar scores for our proposal and OnlyNames. Despite its limitations in referring expression generation, OnlyNames baseline performed very well, which can be accounted for by the fact that WebNLG is a corpus made up of texts potentially used to yield encyclopedia entries, which allow for repetition of proper nouns unlike other types of text. Hallucination and repetition often present on neural models (Rohrbach et al., 2018; Moryossef et al., 2019; Holtzman et al., 2018) can also account for OnlyNames good performance which does not suffer from this problem. An example of this can be seen in the RDF triple set: [Alfa_Romeo_164 | assembly |Milan. Alfa_Romeo_164 | relatedMeanOfTransportation | Saab_9000]. OnlyNames output was "*Alfa Romeo 164*, which is assembled in *Milan*, is a related means of transportation to *Saab 9000*, in that they are both cars", whereas *NeuralREG+Copy* produced the following output "*Romeo Romeo 164*, which is assembled in *Milan*, is a related means of transportation to *Saab*, in that they are both cars". Despite eventual hallucination problems, human evaluation showed that our model has more consistent performance, improving overall quality.

7 Ablation Study

We also performed an ablation study in order to analyze the performance of the different features used by our proposal.

Method We evaluated the copy mechanism, pre- and post-contexts, as well as gender and type embeddings in order to determine which feature best influences the model. The performance of every single feature was analyzed by running the model without it and measuring loss, according to the referring expression accuracy metric in the test part of the data, considering all entities as well as only *seen* and only

unseen ones. When removing the copy mechanism, the model has a similar performance to the original NeuralREG though with the target entity also represented by the entity embeddings for gender and type.

Model	Setup	Accuracy All	Seen	Unseen
Proposal	pre, copy, post, entity type and gender embeddings	0.59	0.73	0.46
Ablation 1	without copy	**0.40**	<u>0.70</u>	**0.09**
Ablation 2	without entity embeddings	<u>0.50</u>	0.71	<u>0.28</u>
Ablation 3	without post-context	0.54	0.77	0.42
Ablation 4	without pre-context	<u>0.50</u>	**0.64**	0.36

Table 5: Results of the Ablation Analysis. Best features (with the highest drop) are **boldfaced**, whereas the second best are <u>underlined</u>.

Results Table 5 depicts the results of our ablation analysis. The removal of the copy mechanism feature (Ablation 1) causes the highest decrease in the referring expression accuracy for *all* entities as well as only *seen* and *unseen* ones, validating this feature as the most efficient within the model. In addition, the copy mechanism relevance for generalizing to unseen entities is proved by a high accuracy drop in the analysis. Furthermore, removing entity embeddings for the entities' gender and type (Ablation 2) causes a negligible drop in all scores, particularly when generating referring expressions to unseen entities. Regarding context, pre-context (Ablation 3) causes the second highest decrease, being validated as the second best feature. Post-context (Ablation 4) does not yield the expected performance regarding accuracy for seen entities, since the produced referring expressions to this type of entities proved better without this feature. Nevertheless, we can stress the importance of post-context for unseen entities, since referring expression accuracy for this kind of entity decreases with the removal of this feature.

8 Discussion

This study set out to address a limitation of NeuralREG, a state-of-the-art encoder-decoder referring expression generation system, which is its failure to generate references to entities not previously seen during its training. To solve the problem, we proposed two extensions to the original approach: a copy mechanism and using a multi-token representation for the referent as well as its gender and type.

Considering pre- and post-contexts where an entity should be referred to and information about the entity's gender and type, at each decoding step our model decides whether the next token of the referring expression should be generated from the output vocabulary or copied from the multi-token input representation of the entity.

Although our approach set out to improve the generation of referring expressions to unseen entities only, an automatic evaluation shows that it presents competitive results regarding all the models when comparing overall performance and also for seen entities. Regarding generation of pronouns and references to unseen entities, our model outperforms ProfileREG and OnlyNames. Furthermore, human evaluation, conducted to rate the automatically generated sentences showed that our model achieved the best results regarding grammaticality. Regarding fluency and semantic adequacy, *NeuralREG+Copy* and OnlyNames presented similar results as shown in table 4. The similarities between both models are striking, which demonstrates that OnlyNames remains a competitive baseline in REG. In order to understand these results and have a deeper insight on the performance of each feature, we also conducted an ablation analysis, which showed different results in referring expression generation for seen and unseen entities.

Surrounding Context Pre- and post-contexts seem to perform different roles when used as input features to generate referring expressions. Based on our ablation analysis, pre-context plays a crucial role, being ranked the most important feature when generating referring expressions to seen entities and third to unseen ones. On the other hand, post-context seems to have a slight contribution only for the generation of references to unseen entities. In fact, when not used, the approach performs better for generation of referring expressions to seen entities.

Copy Mechanism Among the input features, the copy mechanism proved an essential feature of the model. Its importance was supported by the results in the ablation analysis, which pointed to this feature as the most important for the generation of referring expressions to unseen entities. This confirms the copy mechanism to be a productive addition to NeuralREG in order to make it able to work with entities not seen during training.

Gender and Type Entity Representations Besides the copy mechanism, we sought to make NeuralREG generalize to unseen entities by feeding it with embedding representations of the referent's gender and type. Among the motivations to use these features, we considered how easy it is to access this information in the Semantic Web, since entities in WebNLG are represented by their `URIs`. Second, we hypothesized that such representations would allow the model to generate pronominal and descriptive referring expressions to unseen entities. To some extent, the pronominal reference **his** to the unseen entity `Adenan_Satem` produced by our approach and depicted in Table 3 shows that the representations may indeed help. Ablation results also showed that they are the second most important feature in the referring expression generation to unseen entities, only behind the copy mechanism (another extension proposed by our study). Such result confirms our hypothesis.

Future Work Although our two proposed extensions allowed NeuralREG to perform better when compared to its previous version and also generate superior referring expressions to unseen entities, OnlyNames performed slightly better than our approach in the generation of referring expressions to this kind of entities as well as produced similar results to our model during human evaluation. We assume that part of this result is related to known issues in semantic neural models, such as hallucination, which could influence both automatic and human evaluation results, in particular for unseen entities. To fix this issue, we aim to investigate the generation of synthetic referring expression data to augment the training data and better tune our approaches.

Results also show that ProfileREG outperformed our model in the generation of pronouns. We hypothesize this result as an impact of incorrect gender and type information for some entities extracted from DBpedia. For instance, the entity `BBC` in DBpedia[6] is also considered of the type `Person`, leading to the generation of inaccurate descriptions (*The person*) and pronominal (*She* or *He*) outputs. In future work, we aim to manually inspect all type and gender information extracted from DBpedia in order to avoid errors. Additionally, to generate better pronominal referring expressions, we will enhance our approach by using the "profile" computed by the ProfileREG model.

Conclusion We have proposed extensions to the NeuralREG model to overcome shortcomings in not being able to generalize to entities not seen during the training process when generating referring expressions. We can conclude that our proposal contributes to generating more significant referring expressions to unseen entities, besides seen ones. Furthermore, our study provides a new version of a strong baseline within the NLG area. A future direction in our work is to implement the improvements discussed in this study in order to match OnlyNames performance for unseen entities.

Acknowledgments

Research funded by the National Council for Scientific and Technological Development (CNPQ) under grant No. 310630/2017-7), the Foundation for the Coordination and Improvement of Higher Education Personnel (CAPES) (Post doctoral grant No. 88887.508597/2020-00) and the State Funding Agency of Minas Gerais (FAPEMIG) under grant APQ-01.461-14.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

[6] `http://dbpedia.org/page/BBC`

Meng Cao and Jackie Chi Kit Cheung. 2019. Referring expression generation using entity profiles. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3154–3163.

Thiago Castro Ferreira, Emiel Krahmer, and Sander Wubben. 2016. Towards more variation in text generation: Developing and evaluating variation models for choice of referential form. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 568–577.

Thiago Castro Ferreira, Emiel Krahmer, and Sander Wubben. 2017. Generating flexible proper name references in text: Data, models and evaluation. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 655–664, Valencia, Spain, April. Association for Computational Linguistics.

Thiago Castro Ferreira, Diego Moussallem, Ákos Kádár, Sander Wubben, and Emiel Krahmer. 2018a. Neuralreg: An end-to-end approach to referring expression generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1959–1969.

Thiago Castro Ferreira, Diego Moussallem, Emiel Krahmer, and Sander Wubben. 2018b. Enriching the webnlg corpus. In *Proceedings of the 11th International Conference on Natural Language Generation*, pages 171–176.

Thiago Castro Ferreira, Chris van der Lee, Emiel van Miltenburg, and Emiel Krahmer. 2019. Neural data-to-text generation: A comparison between pipeline and end-to-end architectures. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 552–562.

Robert Dale and Ehud Reiter. 1995. Computational interpretations of the gricean maxims in the generation of referring expressions. *Cognitive science*, 19(2):233–263.

Claire Gardent, Anastasia Shimorina, Shashi Narayan, and Laura Perez-Beltrachini. 2017a. Creating training corpora for nlg micro-planners. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 179–188.

Claire Gardent, Anastasia Shimorina, Shashi Narayan, and Laura Perez-Beltrachini. 2017b. 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.

Ari Holtzman, Jan Buys, Maxwell Forbes, Antoine Bosselut, David Golub, and Yejin Choi. 2018. Learning to write with cooperative discriminators. *arXiv preprint arXiv:1805.06087*.

Emiel Krahmer and Kees van Deemter. 2019. Computational generation of referring expressions: An updated survey. In *The Oxford Handbook of Reference*. Oxford University Press.

Vladimir I Levenshtein. 1966. Binary codes capable of correcting deletions, insertions, and reversals. In *Soviet physics doklady*, volume 10, pages 707–710.

Simon Mille, Anja Belz, Bernd Bohnet, Yvette Graham, Emily Pitler, and Leo Wanner. 2018. The first multilingual surface realisation shared task (sr'18): Overview and evaluation results. In *Proceedings of the First Workshop on Multilingual Surface Realisation*, pages 1–12.

Amit Moryossef, Yoav Goldberg, and Ido Dagan. 2019. Step-by-step: Separating planning from realization in neural data-to-text generation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2267–2277.

Jekaterina Novikova, Ondřej Dušek, and Verena Rieser. 2017. The e2e dataset: New challenges for end-to-end generation. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, pages 201–206.

Jekaterina Novikova, Ondřej Dušek, and Verena Rieser. 2018. Rankme: Reliable human ratings for natural language generation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 72–78.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

Ehud Reiter and Robert Dale. 2000. *Building Natural Language Generation Systems*. Cambridge, UK: Cambridge University Press.

Anna Rohrbach, Lisa Anne Hendricks, Kaylee Burns, Trevor Darrell, and Kate Saenko. 2018. Object hallucination in image captioning. *arXiv preprint arXiv:1809.02156*.

Abigail See, Peter J Liu, and Christopher D Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083.

Advaith Siddharthan, Ani Nenkova, and Kathleen McKeown. 2011. Information status distinctions and referring expressions: An empirical study of references to people in news summaries. *Computational Linguistics*, 37(4):811–842.

Kees van Deemter. 2016. Designing algorithms for referring with proper names. In *Proceedings of the 9th International Natural Language Generation conference*, pages 31–35.

Chris van der Lee, Albert Gatt, Emiel van Miltenburg, Sander Wubben, and Emiel Krahmer. 2019. Best practices for the human evaluation of automatically generated text. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 355–368.