Natural Language Generation through Character-Based RNNs with Finite-State Prior Knowledge

Raghav Goyal[1*]; Marc Dymetman[2]; Eric Gaussier[3]
[1]Twenty Billion Neurons, [2]Xerox Research Centre Europe, [3]LIG, University Grenoble Alpes

Abstract

Wen et al. (2015) [1] proposed a Recurrent Neural Network (RNN) approach to the generation of utterances from dialog acts. It employs generation at the word-level, which requires one to pre-process the data by substituting named entities with placeholders. This prevents the model from handling some contextual effects and from managing multiple occurrences of the same attribute.

Our approach :
1. We use a *character-level* model, which unlike the word-level model makes it posssible to learn to “copy” information from the dialog act to the target without having to pre-process the input.
2. In order to avoid generating non-words and inventing information not present in the input, we propose a method for incorporating *prior knowledge* [2] into the RNN in the form of a weighted finite-state automaton over character sequences.

Task

[Dialog Act]: *inform(name=‘phoenix hotel’; area=‘civic center’; accepts credit cards=‘yes’)*
[Realization]: *the phoenix hotel is near the civic center and accepts credit card -s .*

Previous Work and Criticism

Word-based model requires *de-lexicalisation* and *re-lexicalisation*

[Dialog Act]: *inform(name=<NAME>; area=<AREA>; accepts credit cards=‘yes’)*
[Realization]: *the <NAME> is near the <AREA>and accepts credit card -s .*

- Requires a reliable mechanism for Named-Entity Recognition (NER)
- Unable to account for subtle morphological or lexical effects that a specific named entity may have on its context
  - *le ritz (la belle époque) est situé (est située) …)*
  - *the HOTEL NAME is … → the the renaissance is …*
- Does not address multiple slots of the same type, e.g. names of two hotels

Proposed Approach

Character-based Model:
- Attention-based [4] character-to-character model based on *seq2seq*[3] models
- Enables copying at the character level, this prevents us from having to de-lexicalize some aspects of the input

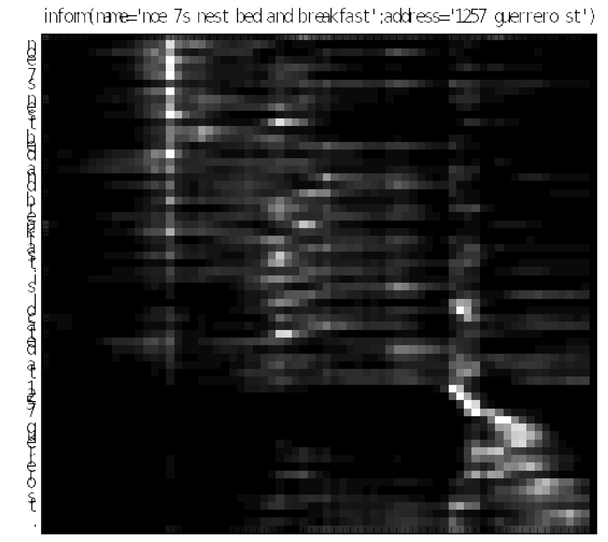

Figure 1. Attention heatmap for a selected example. The x-axis and y-axis denote the input Dialog Act and realization respectively.

[Dialog Act]: *inform(name=’noe 7s nest bed and breakfast’;address=’1257 guerrero st’)*
[Reference]: ** the address is 1257 guerrero st for noe 7s nest bed and breakfast. #*
[Realization]: *noe 7s nest bed and breakfast is located at 1257 guerrero st .*

Background-Adaptor RNNs
- Standard RNN: $a_\theta(x_{t+1}|x_1, x_2, ..., x_t; C)$
- Our model:

$$p_\theta(x_{t+1}|x_1, x_2, ..., x_t; C) \propto \underbrace{a_\theta(x_{t+1}|x_1, x_2, ..., x_t; C)}_{\text{adaptor}} . \underbrace{b(x_{t+1}|x_1, x_2, ..., x_t; C)}_{\text{background}}$$

- Intuition with extreme cases:
  - Background: Uniform Distribution
  - Background: True distribution over the data
- Finite State Machine as the Background Process:

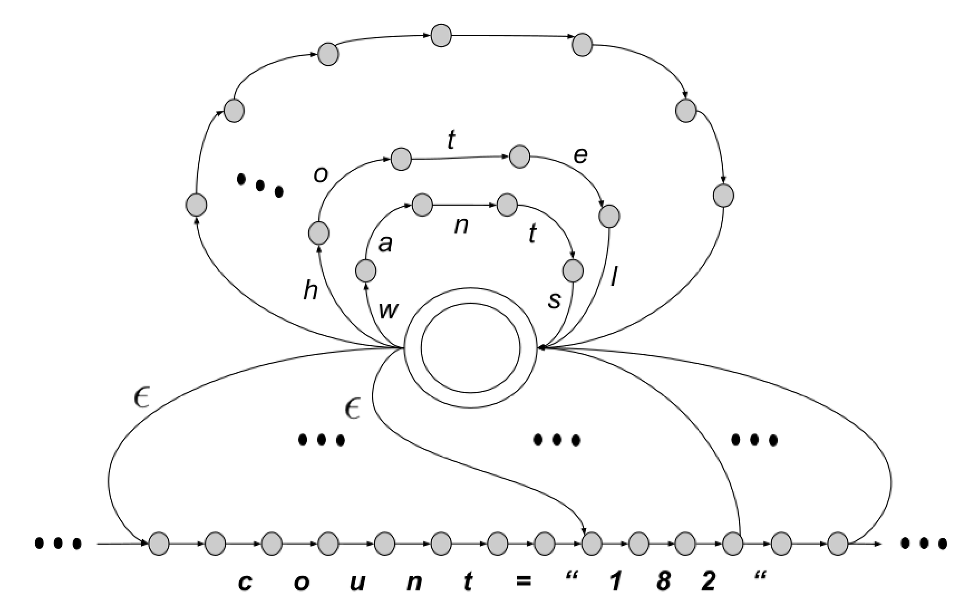

Figure 2. The Background FSA which shows the inclusion of target vocabulary words such as “hotel” and “wants”. Also, it is capable of accepting any substring from the dialog act such as the number “182” (bottom of the figure). The large central state is both initial and final.

Parameterization of the FSA by $\alpha \in [0, 1]$:
- All common words from the target vocabulary: α
- All substrings from the source dialog act: *1 -* α
- Fit using maximum likelihood

Experimental Results

Dataset [1]:
- Consists of 2 domains with ~5K samples each: *hotel* and *restaurant*
- 8 different dialog act types: *inform*, *reject*, *confirm* etc
- 13 different possible slots such as *name*, *pricerange*, *address* etc.

Model	BLEU	
	Hotel	Restaurant
WORD	0.4495	0.4322
C	0.4200**	0.3699**
C-NWFSA	0.4109**	0.3971**
C-WFSA	**0.4655**	**0.4381**

Table 1. BLEU scores of the models computed on the test set. $^{**}p < 0.01$

	Model	Adequacy		Fluency		
		Precision	Recall	No non-words	Non-redundant	Naturalness
Hotel	WORD	0.952*	**0.844**	0.989	0.941*	1.841*
	C	0.844**	0.633**	0.911**	0.974	1.674**
	C-NWFSA	0.844**	0.615**	0.974*	0.974	1.756**
	C-WFSA	**0.978**	0.815	**0.996**	**0.978**	**1.926**
Restaurant	WORD	0.956	**0.793**	0.994	0.976	1.908
	C	0.846**	0.530**	0.926**	**0.988**	1.787**
	C-NWFSA	0.820**	0.609**	0.959**	0.935**	1.731**
	C-WFSA	**0.973**	0.778	**0.997**	0.982	**1.932**

Table 2. Human evaluation of top realization of the models. Statistical significance is computed through a pairwise difference one-tailed Student’s t-test between the model with maximum score against the others. $^{*}p < 0.05$, $^{**} p < 0.01$.

	Selected samples from the Hotel domain
1	inform(name=’the inn san francisco’;address=’943 s van ness ave’;phone=’4156410188’) [C]: the address of the inn san francisco is 943 s van ness ave . their phone number is 4156410188 . [C-NWFSA]: the inn san francisco ’s phone number is 4156410188 [C-WFSA]: the address of the inn san francisco is 943 s van ness ave . the phone number is 4156410188 . [WORD]: **the the** inn san francisco ’s address is 943 s van ness ave and the phone number is 4156410188 .
2	inform(name=’hotel des arts’;price_range=’moderate’) [C]: the hotel des **artea** hotel is in the moderate price range . [C-NWFSA]: the hotel des arts is in the moderate price range . [C-WFSA]: hotel des arts is in the moderate price range . [WORD]: hotel des arts is moderate -ly priced .
	Selected samples from the Restaurant domain
3	inform(name=’yummy yummy’;price_range=cheap;good_for_meal=dinner) [C]: i have found a restaurant called **mimmmmey** is good for dinner . [C-NWFSA]: **mmy** much is a good restaurant is good for dinner . [C-WFSA]: yummy yummy is cheap and good for dinner . [WORD]: yummy yummy is a cheap dinner restaurant with a cheap price range .
4	inform(name=’straits restaurant’;price_range=expensive;food=singaporean;good_for_meal=dinner) [C]: straits restaurant is an expensive restaurant that serves singaporean food . [C-NWFSA]: straits restaurant is expensive and serves singaporean food for dinner . [C-WFSA]: straits restaurant is expensive and is good for dinner . [WORD]: straits restaurant is an expensive restaurant that serves singaporean food and is good for dinner .

Table 3. Example realizations of the models. The most probable realization from a beam of length 5 is shown in each case.

Conclusions

- We proposed a character-level generator which is able to “copy” information from the source dialog act to the target utterance, and which uses original data without requiring pre-processing
- By incorporating prior knowledge in the form of a finite-state automaton, exploiting a notion of “background-augmented” RNN, we discourage the character-level model from generating non-existing words or information for which there is no evidence in the input.

In Proceedings of the *26th International Conference on Computational Linguistics* **(COLING)**. Osaka, Japan. 2016.

* Work performed during Raghav Goyal’s internship at XRCE in 2016.

References

1 Tsung-Hsien Wen et al. Semantically conditioned lstm-based natural language generation for spoken dialogue systems. EMNLP 2015.
2 Marc Dymetman, Chunyang Xiao. 2016. Log-Linear RNNs: Towards Recurrent Neural Networks with Flexible Prior Knowledge. arXiv: 1607.02467, pages 1–22.
3 Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In Advances in Neural Information Processing Systems, pages 3104–3112.
4 Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. In ICLR 2015.

Contact

Name: Raghav Goyal
Email: raghav.goyal@twentybn.com

Name: Marc Dymetman
Email: marc.dymetman@xerox.com

Name: Eric Gaussier
Email: eric.gaussier@imag.fr