Personality-dependent Neural Text Summarization

Pablo Botton da Costa
University of São Paulo
São Paulo, Brazil
pablo.botton.costa@gmail.com

Ivandré Paraboni
University of São Paulo
São Paulo, Brazil
ivandre@usp.br

Abstract

In Natural Language Generation (NLG) systems, personalization strategies - i.e., the use of information about a target author to generate text that (more) closely resembles human-produced language - have long been applied to improve results. The present work addresses one such strategy - namely, the use of Big Five personality information about the target author - applied to the case of abstractive text summarization using neural sequence-to-sequence models. Initial results suggest that having access to personality information does lead to more accurate (or human-like) text summaries, and paves the way for more robust systems of this kind.

1 Introduction

Computational approaches to text summarization may be divided into two general categories: abstractive and extractive summarization. Extractive summarization consists of selecting relevant pieces of text to compose a subset of the original sentences, whereas the more complex abstractive summarization involves interpreting the input text and rewriting its main ideas in a new, shorter version. Both strategies may be modelled as a machine learning problem by making use of unsupervised (Ren et al., 2017), graph-based and neural methods (Wan and Yang, 2006; Cao et al., 2015), among others. The present work focuses on the issue of neural abstractive summarization, addressing the issue of personalized text generation in systems of this kind.

Text-generating systems may in principle produce always the same fixed output from a given input representation. In order to generate more natural (or 'human-like') output, however, systems of this kind will often implement a range of stylistic variation strategies. Among these, the use of computational models of *human personality* has emerged as a popular alternative, and it is commonly associated with the rise of the Big Five model of human personality (Goldberg, 1990) in many related fields.

The Big Five model is based on the assumption that differences in personality are reflected in natural language use, and comprises five fundamental dimensions of personality: *Extraversion*, *Agreeableness*, *Conscientiousness*, *Neuroticism*, and *Openness to experience*. Given its linguistic motivation, the Big Five personality traits have been addressed in a wide range of studies in both natural language understanding and generation alike. Thus, for instance, the work in Mairesse and Walker (2007) introduces PERSONAGE, a fully-functional NLG system that produces restaurant recommendations. PERSONAGE and many of its subsequent extensions support multiple stylistic variations that are controlled by personality information provided as an input.

The use of personality information for text summarization, by contrast, seems to be far less common, and we are not aware of any existing work that addresses the issue of personality-dependent neural text summarization. Based on these observations, this paper introduces a personality-dependent text summarization model that makes use of a corpus of source and summary text pairs labelled with personality information about their authors. In doing so, our goal is to use personality information to generate summaries that more closely resemble those produced by humans.

The rest of this paper is structured as follows. Section 2 discusses the issues of sequence-to-sequence learning and attention mechanism for text summarization. These are the basis of our current work described in Section 3. Section 4

Proceedings of Recent Advances in Natural Language Processing, pages 205–212,
Varna, Bulgaria, Sep 2–4, 2019.
https://doi.org/10.26615/978-954-452-056-4_024

reports two experiments comparing the proposed models against a number of alternatives, and Section 5 presents final remarks and future work.

2 Background

Due to the capacity of neural language generation models to learn and automatically induce representations from text (Rush et al., 2015; Nallapati et al., 2016; Mikolov et al., 2013), neural abstractive summarization has attracted a great deal of attention in the field. Architectures of this kind may not only produce high-quality summaries, but may also embed external information easily (See et al., 2017). Accordingly, these models have achieved significant results, at least in terms of intrinsic evaluation measures such as BLEU (Papineni et al., 2002) or ROUGE (Lin and Hovy, 2003), when comparing to extractive approaches (Celikyilmaz et al., 2018).

2.1 Sequence-to-sequence Learning

Neural text summarization models are often grounded on a particular kind of neural network, the sequence-to sequence architecture (Sutskever et al., 2014a; Cho et al., 2014). In models of this kind, input text is modelled as a sequence of representations carrying any contextual information from end to end in the generation process. More formally, a sequence-to-sequence model is defined in Goodfellow et al. (2016) as a neural network that directly models the conditional probability $p(y|x)$ of a source sequence, $x_1, ..., x_n$, to a target sequence, $y_1, ..., y_m$[1].

A basic form of sequence-to-sequence model consists of two main components: (i) an encoder that computes a representation s for each source sequence; and (ii) a decoder that generates one target token at a time, decomposing the conditional probability as follows:

$$p(y|x) = \sum_{j=1}^{m}(y_j|y_{<j}, s)$$

A common strategy for learning sequence representations is by making use of Recurrent Neural Networks (RNN) (Rumelhart et al., 1986). According to Hochreiter and Schmidhuber (1997), a RNN generalizes the concept of feed-forward neural network to sequences. Given a temporal sequence of inputs $(x_1, ..., x_t)$, the standard RNN computes a sequence of outputs $(y_1, ..., y_t)$ mapped onto sequences using the following equation (Sundermeyer et al., 2012):

$$h_t = sigmoid(W^{hx}x_t + W^{hh}h_{t-1})$$
$$y_t = W^{yh}h_t$$

A simple strategy for general sequence learning is to map the input sequence to a fixed-sized vector using a RNN, and then map the vector to the target sequence by using a second RNN. This may in principle be successful, but long term dependencies may make the training of the two networks difficult (Bengio et al., 1994; Hochreiter, 1998). As an alternative, Long Short-Term Memory (LSTM) networks (Hochreiter and Schmidhuber, 1997), and their simplification known as Gated Recurrent Unit (GRU) (Cho et al., 2014) are known to learn problems with long range temporal dependencies, and may therefore succeed in this setting.

The goal of a LSTM/GRU network is to estimate the conditional probability $p(y|x)$, where $(x_1, ..., x_{t'})$ is an input sequence and $(y_1, ..., y_t)$ is its corresponding output sequence whose length t' may differ from t (Cho et al., 2014). The conditional probability is computed by first obtaining the fixed dimensional representation v of the input sequence $(x_1, ..., x_{t'})$ given by the last hidden state of the network, and by computing the probability of $(y_1, ..., y_t)$ with a standard LSTM/GRU formulation in which the initial hidden state is set to the representation v of $(x_1, ..., x_{t'})$. Finally, each $p(y_j|s, y_1, ..., y_{j-1})$ distribution is represented with a softmax over all the words in the vocabulary.

GRUs are distinct from LSTMs in that a GRU architecture contains only a single unit to control when the current states 'forgets' a piece of information (Goodfellow et al., 2016). Due to this simplification, GRUs can directly access all hidden states without bearing the price of a memory state (Cho et al., 2014).

GRU architectures model sequences as causal relationships through the input sequence by examining left-to-right relationships only (Goodfellow et al., 2016). However, many sequence classification problems may require predicting an output that depends (bidirectionally) on the entire input sequence, that is, from left to right and also from right to left. This is the case, for instance, of a large number of common NLP applications that

[1] Sentences are assumed to start with a special 'start-of-sentence' token $< bos >$ and end with an 'end-of-sequence' token $< eos >$.

need to pay regard to contextual dependency when modelling phrases and sentences.

Bidirectional GRUs (Bi-GRUs) are applied to a wide range of tasks to scan and learn both left-to-right and right-to-left dependencies, which can capture complementary types of information from its inputs. The left and right hidden representations produced by GRUs can be linearly combined (θ) to form a final representation (Goodfellow et al., 2016): $h_t = h_t^{\leftarrow} \; \theta \; h_t^{\rightarrow}$.

2.2 Attention Mechanism

Sequence-to-sequence architectures have been successfully applied to a wide range of tasks, including machine translation and natural text generation (Cho et al., 2014; Sutskever et al., 2014a) and, accordingly, have been subject to a great deal of extensions and improvements. Among these, the use of more context-aware sequence generation methods (Cho et al., 2014) and the use of attention mechanism to score and select words that best describe the intended output are discussed below.

In natural language generation, attention models as introduced in Cho et al. (2014) and Sutskever et al. (2014a) are intended to generalize the text generation task so as to handle sequence pairs with different sizes of inputs and outputs. This approach, subsequently called sequence-to-sequence with attention mechanism, applies a mapping strategy from a variable-length sentence to another variable-length sentence. This mapping strategy is a scoring system over the contextual information from the input sequence (Cho et al., 2014), making a set of attention weights.

Attention-based models (Sutskever et al., 2014b; Luong et al., 2015) are sequence-to-sequence networks that employ an encoder to represent the text utterance and an attention-based decoder that generates the response, one token at a time. More specifically, neural text summarization can be viewed as a sequence-to-sequence problem (Sutskever et al., 2014a), where a sequence of input language tokens $x = x_1, ..., x_m$ describing the input text are mapped onto a sequence of output language tokens $y_1, ..., y_n$ describing the target text output. The encoder is a GRU unit (Cho et al., 2014) that converts $x_, ..., x_m$ into a sequence of context-sensitive embeddings $b_1, ..., b_m$. A general-attention decoder generates output tokens one at a time. At each time step j, the decoder generates y_j based on the current hidden state s_j, and then updates the hidden state s_{j+1} based on s_j and y_j. Formally, the attention decoder is defined by original equation proposed in Cho et al. (2014):

$$s_1 = tanh(W^{(s)} b_m)$$
$$p(y_j = w | x, y_{1:j-1}) \; \alpha \; exp(U[s_j, c_j])$$
$$s_{j+1} = GRU([\phi^{(out)}(y_j), c_j], s_j)$$

where $i \; \epsilon \; \{1, ..., m\}$, $j \; \epsilon \; \{1, ..., m\}$ and the context vector c_j, is the result of general attention (Luong et al., 2015). The matrices $W^{(s)}$, $W^{(\alpha)}$, U and the embedding function $\phi^{(out)}$ are decoder parameters.

3 Current Work

Our basic model is generally inspired from the architecture in Cho et al. (2014), with an added personality embedding layer. As in many other sequence-to-sequence models with attention, our model takes as an input a sentence, and produces as an output a set of words that summarizes the given input. The actual rendering of this output as structured text is presently not addressed.

The proposed architecture is illustrated in Figure 1, which is adapted from Cho et al. (2014), and further discussed below.

In this example, $B \; B \; B \; B$ represent the input sequence from the target sequence $Z \; X$, and C is the personality embedding representation. The five main components of the architecture are as follows.

(A) a bidirectional GRU that maps words to personality types

(B) a word embedding layer

(C) a personality embedding layer

(D) an attention mechanism

(E) a bidirectional GRU that outputs word encodings

The input bidirectional GRU (A) produces a word-to-personality compositional representation of each word. This serves two main purposes: combining the composite sequences of words and personality information, and combining attention weights over sequences in our decoder model.

The word embeddings layer (B) produces a typical word-level representation of each input word. In the present work, we make use of both random

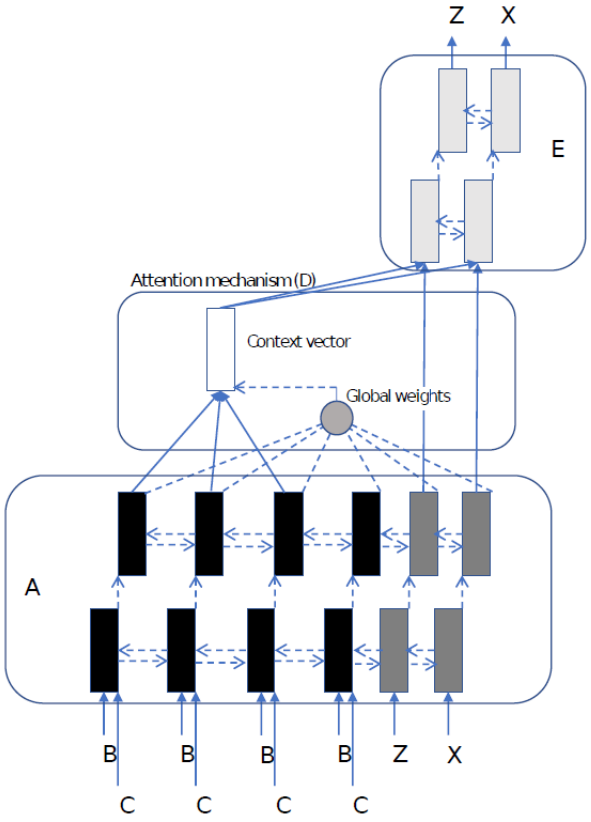

Figure 1: Model architecture

and pre-trained word embeddings. The latter are Skip-gram 300 word embeddings taken from Hartmann et al. (2017).

Word embeddings are complemented with induced personality embeddings (C) for each target author. The role of this layer is twofold. First, it is intended to learn the probability $P(Y|X, personality)$, that is, the $personality$ representation of each author for each word in the vocabulary. Second, this layer is also intended to decide which profile value should be selected (from the corpus gold standard annotation) in order to generate a summary.

The attention mechanism (D) attempts to learn a general representation from the most important parts of the input text at each time step. To this end, the experiments described in the next section will consider two score function alternatives: general attention and dot product.

Finally, the output bidirectional GRU (E) combines the attention weight representations, and produces a final encoding for each word. A loss function describe the overall generation probability, and it is intended to optimize the above parameters. This function is described as follows.

$$\ell_1(\theta, D(c), D(x,y)) = -\sum_{(X,Y)\ \epsilon\ D^c \cup D^{pr}} logP(Y|X, < k_i, v_i >)$$

$$= -\sum_{(X,Y)\ \epsilon\ D^c} logP^{fr}(Y|X)$$

The first term of the function is the negative log likelihood of observing $D^{(c)}$ and the second term for $D^{(pr)}$. $D^{(pr)}$ consists of pairs where a summary is related to a profile key and its response match to the summary, and $D^{(c)}$ has only general text-summary pairs. $< k_i, v_i >$ is the personality representation. The decoder P^{fr} does not have shared parameters. A simple epoch-based training strategy using gradient descendent is performed.

4 Evaluation

We envisaged two experiments on neural text summarization based on the model described in the previous section. The first experiment aims to assess whether a general or a dot product attention mechanism is more suitable to the task. The second experiment focuses on our main research question, that is, on whether the use of personality information does improve summarization results.

As in many (or most) sequence-to-sequence approaches to text generation, our work focuses on the selection of text segments to compose an abstract summary, but it does not address the actual rendering of the final output text, which would normally require additional post-processing. Each of the two experiments is discussed in turn in the following sections, but first we describe the dataset taken as their basis.

4.1 Data

We make use of the text and caption portions of the b5 corpus in Ramos et al. (2018), called *b5-text* and *b5-caption*. The corpus conveys 1510 multi- and single-sentence image description pairs, all of which labelled with Big Five personality information about their authors. Table 1 summarizes the corpus descriptive statistics.

The corpus was elicited from a crowd sourcing task in which participants were requested to provide both long and short descriptions for 10 stimulus images taken from GAPED, a database of images classified by valence and normative significance designed to elicit various reactions (Dan-

Table 1: Corpus descriptive statistics.

Data	Words	Average	Types	Average
text	84463	559.4	37210	246.4
caption	4896	32.4	4121	27.3

Figure 2: Stimulus image from *GAPED* (Dan-Glauser and Scherer, 2011).

Glauser and Scherer, 2011). From a set of 10 selected images with valence degrees in the 3 to 54 range, participants were first instructed to describe everything that they could see in the scene (e.g., as if helping a visually-impaired person to understand the picture) and, subsequently, were requested to summarize it in a single sentence (similar to a picture caption.)

An example of stimulus image is illustrated in Figure 2. We notice however that in the present work we only consider the text elicited from these images, and not the images themselves.

Based on scenes as in Figure 2, the following is a possible long description (translated from the Portuguese original text) of the kind found in the corpus.

> '*A black baby, about one year old. He's in a cradle. He is dressed in a dirty blue blouse, on a pink sheet, without a pillow. A blue blanket is next to the baby. It seems that he has not taken a shower for a while.*'

A single-sentence summary for the same scene (and which would have been written by the same participant in the data collection task) may be represented as the following example.

> '*A sad-looking baby.*'

In the experiments described in the next sections, texts were pre-processed by removing punctuation and numerical symbols. In addition to that,

Table 2: Data split

Split	Samples
Train	1358
Validation	152
Total	1510

the first data split performed for the purpose of cross-validation is shown in Table 2.

4.2 Experiment 1: Basic Neural Summarization with Attention Mechanism

In Encoder-Decoder Recurrent Neural Networks, the global attention mechanism may be seen as a model-inferred context vector computed as a weighted average of all inputs by making use of a score function. The choice of score function may have a great impact on the overall performance of the model, and for that reason in what follows we examine two alternatives: using the dot product over the context vectors of the source, and using the learned representation over the context states.

To this end, our first experiment evaluates our basic summarization model (cf. the previous section) in two versions, namely, using general and dot product attention mechanisms. Both of these models, hereby called *sDot* and *sGen*, will make use of encoder/decoder randomized word embedding of size 300, and two encoder/decoder hidden units of size 600.

Both models were trained using Adam optimization with mini batches of size 128. The initial learning rate was set to 0.0001 with a gradient clipping based on the norm of the values. We also applied different learning rates for the decoder module, set to five times the learning rate of the encoder. In order to reduce over-fitting, a 0.5 drop-out regularization was applied to both embedding layers.

Model optimization was performed by using gradient descendent with masked loss, and by applying early stopping when the BLEU scores over the evaluation dataset did not increase for 20 epochs. Except for the embedding layer, all other

Table 3: 10-fold cross validation BLEU scores for text summarization using dot product (sDot) and general (sGen) attention. the best result is highlighted.

Model	BLEU
sGen	**13.88**
sDot	13.63

Table 4: 10-fold cross validation BLEU scores for text summarization with (sPers) and without (sBase) personality information. The best result is highlighted.

Model	BLEU
sBase	14.21
sPers	**14.58**

parameters were initialized by sampling from a uniform distribution $U(-sqrt(3/n), sqrt(3/n))$, where n is the parameter dimension.

We performed 10-fold cross-validation over our corpus data, and we compared the output summaries produced by both models using BLEU[2]. Results are presented in Table 3.

From these results, we notice that the attention mechanism based on the general function in *sGen* outperforms the use of dot function in *sDot*. Although the difference is small, the use of a generalized network to learn how to align the contextual information is superior to simply concatenating contextual information obtained from the global weights. Based on these results, the general attention strategy will be our choice for the next experiment.

4.3 Experiment 2: Personality-dependent Summarization

Our second and main experiment assesses the use of personality information in text summarization. To this end, two models are considered: the full personality-aware model presented in Section 3, hereby called *sPers*, and a simplified baseline version of the same architecture without access to personality information, hereby called *sBase*. In doing so, our goal is to show that summaries produced by *sPers* resemble the human-made texts (as seen in the corpus) more closely than those produced by *sBase*.

Both *sPers* and *sBase* make use of pre-trained skip-gram 300 word embeddings for the Brazilian Portuguese language taken from Hartmann et al. (2017). Both models also make use of encoder/decoder randomized word embedding of size 300, and two encoder/decoder hidden units of size 600 with general attention.

[2]We are aware that, although popular in machine translation and text generation, BLEU may not be the ideal metrics for the present task (Liu et al., 2011; Song et al., 2013), and that it may not co-relate well with, e.g., human judgments (Reiter and Belz, 2009).

All optimization, training and other basic procedures are the same as in the previous experiment. Results are presented in Table 3.

We notice that personality-dependent summarization as provided by *sPers* outperforms standard summarization (i.e., with no access to personality information) as provided by *sBase*. Although the difference is once again small (which may be explained by the limited size of our dataset), this outcome offers support to our main research hypothesis by illustrating that the use of author personality information may improve summarization accuracy.

4.4 Selected Examples

As a means to illustrate the kinds of output that may be produced by our models, Table 5 presents a number of examples taken from the original corpus summaries, and the corresponding summaries obtained from the same input by making use of the *sBase* baseline and by the personality-dependent *sPers* models. For ease of illustration, the examples are informally grouped into three error categories (small, moderate and large) according to the distance between the corpus summaries and their *sPers* counterparts, and are presented in both original (Portuguese) and translated (English) forms.

5 Final Remarks

This paper addressed the use of Big Five personality information about the target author to generate personalized summaries in neural sequence-to-sequence text summarization. The model - consisting of two bidirectional GRUs, word embeddings and attention mechanism - was evaluated in two versions, namely, with and without an additional personality embedding layer. Initial results suggest that having access to personality information does lead to more accurate (or human-like) text summaries.

The use of personality information is of course only one among many possible personalization

Table 5: Selected examples taken from the corpus, baseline (sBase) and personality-dependent (sPers) summarization models, grouped by distance (small, moderate or large) between sPers and the expected (corpus) summary in original Portuguese (Pt) and translated English (En).

Error	Model	Summary (Pt)	Summary (En)
small	corpus	*homem na cerca*	*man by fence*
	sBase	*homem idoso*	*elderly man*
	sPers	*homem na cerca*	*man by fence*
moderate	corpus	*pessoas pedindo ajuda*	*people asking for help*
	sBase	*pessoas esperando*	*people waiting*
	sPers	*pessoas aguardam atendimento*	*people waiting for help*
large	corpus	*menino com um balde de terra*	*boy with a bucket full of soil*
	sBase	*crianca com balde*	*child with bucket*
	sPers	*crianca com balde de terra*	*child with bucket full of soil*

strategies for text summarization. In particular, we notice that the increasing availability of text corpora labelled with author demographics in general (e.g., gender, age, education information etc.) may in principle support a broad range of speaker-dependent summarization models. Thus, as future work we intend to extend the current approach along these lines, and provide additional summarization strategies that may represent more significant gains over the standard, fixed-output summarization approach.

Acknowledgements

The authors acknowledge support by FAPESP grant 2016/14223-0 and from the University of São Paulo.

References

Yoshua Bengio, Patrice Simard, and Paolo Frasconi. 1994. Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks* 5(2):157–166.

Ziqiang Cao, Furu Wei, Sujian Li, Wenjie Li, Ming Zhou, and Houfeng Wang. 2015. Learning summary prior representation for extractive summarization. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*. volume 2, pages 829–833.

Asli Celikyilmaz, Antoine Bosselut, Xiaodong He, and Yejin Choi. 2018. Deep communicating agents for abstractive summarization. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. pages 1662–1675.

Kyunghyun Cho, Bart van Merrienboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the properties of neural machine translation: Encoder–decoder approaches. In *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*. Association for Computational Linguistics, Doha, Qatar, pages 103–111.

Elise S. Dan-Glauser and Klaus R. Scherer. 2011. The Geneva affective picture database (GAPED): a new 730-picture database focusing on valence and normative significance. *Behavior Research Methods* 43(2):468–477.

Lewis R. Goldberg. 1990. An alternative description of personality: The Big-Five factor structure. *Journal of Personality and Social Psychology* 59:1216–1229.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. 2016. *Deep Learning*. MIT Press.

Nathan Hartmann, Erick Fonseca, Christopher Shulby, Marcos Treviso, Jéssica Rodrigues, and Sandra Aluísio. 2017. Portuguese word embeddings: Evaluating on word analogies and natural language tasks. In *11th Brazilian Symposium in Information and Human Language Technology - STIL*. Uberlândia, Brazil, pages 122–131.

Sepp Hochreiter. 1998. The vanishing gradient problem during learning recurrent neural nets and problem solutions. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 6(02):107–116.

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

Chin-Ye Lin and Eduard Hovy. 2003. Automatic evaluation of summaries using n-gram co-occurrence statistics. In *Proceedings of HLT-NAACL 2003*. Association for Computational Linguistics, Edmonton, Canada, pages 71–78.

Chang Liu, Daniel Dahlmeier, and Hwee Tou Ng. 2011. Better evaluation metrics lead to better machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, pages 375–384.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Lisbon, Portugal, pages 1412–1421. https://doi.org/10.18653/v1/D15-1166.

François Mairesse and Marilyn Walker. 2007. PERSONAGE: Personality generation for dialogue. In *45th Annual Meeting-Association For Computational Linguistics*. Association for Computational Linguistics (ACL), Sheffield, pages 496–503.

Tomas Mikolov, Scott Wen-tau, and Geoffrey Zweig. 2013. Linguistic regularities in continuous space word representations. In *Proc. of NAACL-HLT-2013*. Association for Computational Linguistics, Atlanta, USA, pages 746–751.

Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Caglar Gulcehre, and Bing Xiang. 2016. Abstractive text summarization using sequence-to-sequence RNNs and beyond. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*. Association for Computational Linguistics, Berlin, Germany, pages 280–290. https://doi.org/10.18653/v1/K16-1028.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceddings of ACL-2002*. Association for Computational Linguistics, Philadelphia, PA, USA, pages 311–318.

Ricelli Moreira Silva Ramos, Georges Basile Stavracas Neto, Barbara Barbosa Claudino Silva, Danielle Sampaio Monteiro, Ivandré Paraboni, and Rafael Felipe Sandroni Dias. 2018. Building a corpus for personality-dependent natural language understanding and generation. In *11th International Conference on Language Resources and Evaluation (LREC-2018)*. ELRA, Miyazaki, Japan, pages 1138–1145.

Ehud Reiter and Anja Belz. 2009. An investigation into the validity of some metrics for automatically evaluating natural language generation systems. *Computational Linguistics* 35(4):529–558.

Pengjie Ren, Zhumin Chen, Zhaochun Ren, Furu Wei, Jun Ma, and Maarten de Rijke. 2017. Leveraging contextual sentence relations for extractive summarization using a neural attention model. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, pages 95–104.

David E. Rumelhart, Geoffrey Hinton, and Ronald J. Williams. 1986. Learning representations by back propagating errors. *Nature* 323:533–536. https://doi.org/10.1038/323533a0.

Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Lisbon, Portugal, pages 379–389. https://doi.org/10.18653/v1/D15-1044.

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.

Xingyi Song, Trevor Cohn, and Lucia Specia. 2013. Bleu deconstructed: Designing a better mt evaluation metric. *International Journal of Computational Linguistics and Applications* 4(2):29–44.

Martin Sundermeyer, Ralf Schlüter, and Hermann Ney. 2012. Lstm neural networks for language modeling. In *Thirteenth annual conference of the international speech communication association*.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014a. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*. pages 3104–3112.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014b. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*. pages 3104–3112.

Xiaojun Wan and Jianwu Yang. 2006. Improved affinity graph based multi-document summarization. In *Proceedings of the human language technology conference of the NAACL, Companion volume: Short papers*. Association for Computational Linguistics, pages 181–184.