Generating Formality-tuned Summaries Using Input-dependent Rewards

Kushal Chawla*
University of Southern California
Los Angeles, CA, USA
kchawla@usc.edu

Balaji Vasan Srinivasan, Niyati Chhaya
Adobe Research
Bangalore, India
balsrini,nchhaya@adobe.com

Abstract

Abstractive text summarization aims at generating human-like summaries by understanding and paraphrasing the given input content. Recent efforts based on sequence-to-sequence networks only allow the generation of a single summary. However, it is often desirable to accommodate the psycho-linguistic preferences of the intended audience while generating the summaries. In this work, we present a reinforcement learning based approach to generate formality-tailored summaries for an input article. Our novel input-dependent reward function aids in training the model with stylistic feedback on sampled and ground-truth summaries together. Once trained, the same model can generate formal and informal summary variants. Our automated and qualitative evaluations show the viability of the proposed framework.

1 Introduction

Efficient content consumption is not only driven by the contained information but also by the tone/style of the presentation. The style in text is its non-informational or non-factual aspect and usually drives the quality of response from its audience. A *persuasive* snippet or a teaser might drive up sales for a marketing message and a piece of *formal* text will appeal better to corporate executives as against informal communication. Similarly, not all long form content is easy to read. A succinct representation of the content, i.e. the summary, plays an important role for its quick and efficient consumption. While, text summarization and to some extent *style-understanding* have been independently studied, approaches to generate style-tailored summaries are limited.

Models for style predictions (Pavlick and Tetreault, 2016; Brooke et al., 2010; Danescu-Niculescu-Mizil et al., 2013) are limited to measuring style in text. There have been multiple attempts towards transfer of style (Artetxe et al., 2017; Han et al., 2017; Shen et al., 2017; Tikhonov and Yamshchikov, 2018). Rao and Tetreault (2018) introduced a parallel corpus for formality style transfer with neural machine translation benchmarks. Niu et al. (2017) generate automatic translations tailored towards formality. However, none of these approaches account for the length/succinctness of the created content, and hence do not address the stylized summarization task. In this work, we propose an approach to generate summaries while simultaneously tailoring towards formality preferences (Table 1).

Tunable or controlled summary generation has picked up pace in recent times. Algorithms allow for controlling various dimensions of the output summary such as the length or entities (Fan et al., 2017) and topics (Krishna and Srinivasan, 2018). Since these approaches primarily rely on the diversity in the given dataset, extending these approaches for formality tailored summarization would require a diverse summarization corpus that captures subtleties in various formal variants. Since such a dataset is difficult to curate, it is non-trivial to use these methods as is. Reinforcement learning (RL) based loss functions have recently shown promise in tuning the output on rewards such as ROUGE (Paulus et al., 2018). In such methods, the model receives explicit feedback on the sampled sequences while training. If directly applied for controlling stylistic parameters like formality, such a method would need two separately trained models for generating formal and informal summaries and thus, may miss out on the common learnings.

In this work, we propose a method to incor-

* Work done when author was a full time employee at Adobe Research

Proceedings of the 23rd Conference on Computational Natural Language Learning, pages 833–842
Hong Kong, China, November 3-4, 2019.

porate formality in abstractive text summarization. To build a formality-rich training dataset, we merge data from two domains: news and social media - the first one representing more formal language and the latter, informal. We define a novel input-dependent reward function which aids in training the model with stylistic feedback on sampled and ground-truth summaries together. Once trained, the same model can generate formal and informal summary variants. We show the effectiveness of our approach through automated and crowd-sourced experiments, evaluating both the quality and formality levels of the generated summaries. Table 1 shows sample formal and informal summary variants, generated from our approach on an instance from the testset.

Input Article: katrina had been expressing anxiety for a while now about how worried she was about the bridal shower and being the center of attention of a bunch of people she did n't know . that 's completely normal . she has been so worried that she determined to send him a short email outlining what she would do if she were in his shoes . that absolutely counts as meddling . that 's literally the definition of meddling sticking your nose where it does n't belong . katrina talked to my mom for about five minutes and then sat about twenty feet away from her during this tournament while my mom sat alone . and katrina is n't obligated to entertain your mother . your mom could 've talked to other people instead ...
Informal: katrina had been so worried that she was abt the bridal shower & being the center of attention of a bunch of people she did n't know . katrina 's n't obligated to entertain ur mom .
Formal: katrina had been expressing anxiety for the bridal shower and being the center of attention of a bunch of people she did not know . she has been therefore worried that she determined to transmit him a short email outlining what she were in his shoes .

Table 1: Example formal and informal summary variants generated on an instance from our testset.

2 Related Work

Early abstractive summarization efforts were either template-based (Wang and Cardie, 2013; Genest and Lapalme, 2011) or employed ILP-based sentence compression (Filippova, 2010; Berg-Kirkpatrick et al., 2011; Banerjee et al., 2015). With the advent of deep sequence-to-sequence models (Sutskever et al., 2014), attention-based neural models have been proposed for long text summarization (Rush et al., 2015; Chopra et al., 2016). Recent approaches (Nallapati et al., 2017; See et al., 2017) have focused on larger datasets such as the CNN/DailyMail corpus (Hermann et al., 2015; Nallapati et al., 2016). Gulcehre et al. (2016) introduced the ability to copy out-of-vocabulary words from the article to incorporate rarely seen words like names in the generated text. Tu et al. (2016) included the concept of coverage, to prevent the models from repeating the same phrases while generating a sentence. See et al. (2017) proposed a pointer-generator framework which incorporates these improvements, and also learns to switch between generating new words and copying them from the source article. We use this pointer-generator framework as the underlying architecture.

2.1 Incorporating additional constraints

Controlled summary generation has only recently gained popularity. Variational auto-encoders (Hu et al., 2017) or adversarial training (Shen et al., 2017) have been explored for non-parallel stylistic text generation. Sennrich et al. (2016) propose modifications to neural machine translation to tune the level of politeness in the generated text. Ficler and Goldberg (2017) use a conditional language model to control variations like descriptiveness and sentiment simultaneously during generation. Efforts for constrained text summarization are rather limited with no efforts attempting to incorporate psycho-linguistic preferences. Krishna and Srinivasan (2018) incorporate input topic information in the output summary using a topic-vector along with the input word sequence.

Fan et al. (2017) control length and entity in textual summaries using explicit input indicators or tokens. Paulus et al. (2018) directly control the ROUGE evaluation metric using the Self Critical Sequence Training (SCST) (Rennie et al., 2017) algorithm. We build on these approaches and show through our experiments that our approach is able to generate better formality-tuned summaries in comparison to these methods.

2.2 Incorporating Formality

Formality is an important style or tone dimension in written text. Although there are existing works which model formality in text (Brooke et al., 2010; Lahiri, 2015; Pavlick and Tetreault, 2016; Chhaya et al., 2018), there have been limited attempts to incorporate it in text generation. Sheikha and Inkpen (2011) used a predefined set of rules based on formal-informal parallel lists to generate formal and informal sentences. More recently, a parallel corpus of formality style transfer (Rao and Tetreault, 2018) was released with NMT-based benchmarks. To the best of our knowledge, we are the first to introduce formality in the space of abstractive text summarization.

Generating text with varying levels of formality was studied recently in Machine Translation (Niu et al., 2017, 2018). A re-ranking mechanism on

the decoded hypotheses is used to control the formality of the generated translations. We augment our decoding module with a similar, but simpler re-ranking method to enhance our approach.

3 Pointer-Generator Framework

Our approach uses the pointer generator network (See et al., 2017) as the underlying architecture. This section explains this framework briefly for the sake of completion. The model is based on an encoder-decoder setup. The bi-directional LSTM encoder takes the article x as an input and computes a sequence of encoder hidden states $h_1, h_2, ..h_n$. The last state h_n becomes the initial state of the LSTM decoder which uses an attention mechanism to generate the output summary word by word. Further, at each time step, the decoder network computes p_{gen}, the probability of generating a new word from the vocabulary,

$$p_{gen} = \sigma(w_h^T h_t^* + w_s^T s_t + w_x^T x_t + b_{gen}) \quad (1)$$

where w_h, w_s, w_x, b_{gen} are trainable parameters. h_t^* is the context vector capturing the attention distribution, s_t is the decoder internal state and x_t is the decoder input at t^{th} time step. The total probability of w being the next word generated in the summary, $p(w)$, is given by,

$$p(w) = p_{gen} P_{vocab}(w) + (1 - p_{gen}) \sum_{i:w_i=w} a_i^t \quad (2)$$

where p_{gen} provides a switch between generating a new word from $P_{vocab}(w)$ or copying a word from the input based on the attention distribution. The training loss is set to be the average negative log-likelihood of the ground truth summary:

$$L_{nll} = -\frac{1}{T} \sum_{t=1}^{T} \log \left[p(y_t^* | y_1^*, ..., y_{t-1}^*, x) \right] \quad (3)$$

where y^* is the ground-truth word sequence. We propose to modify the above training objective in order to incorporate psycho-linguistic preferences of the target audience, with a focus on formality.

4 Generating (In)formal Summaries

A major challenge in incorporating formality into the summarization system is the lack of a formality-diverse dataset. To the best of our knowledge, there exists no data which either has both formal and informal ground-truth summaries for the same input article or where the provided ground-truth summaries are diverse on the formality scale. This makes the direct use of explicit indicators ineffective (Section 6.2), which have been shown to capture this diversity in the given dataset well (Fan et al., 2017; Krishna and Srinivasan, 2018). To address this, we work off a dataset mixed from two different domains: news and social, making it more formality-diverse (Section 5).

While diversity in the data helps the model to learn the (in)formal parts in the text, we further employ a modified reinforcement learning approach which teaches the decoder module to write (in)formally through explicit feedback. The model is trained using feedback on the formality of both sampled and ground-truth summaries together. The pointer-generator model is trained using the negative log-likelihood loss L_{nll} as given in Equation 3. We make the use of policy gradients by introducing an additional loss term L_{rl} in the training objective:

$$L_{rl} = -\left[r(y^s) \right] \sum_{t=1}^{T} \log \left[p(y_t^s | y_1^s, .., y_{t-1}^s, x) \right], \quad (4)$$

where r is the formality-based reward function. y^s is a sampled word sequence generated by sampling from the $p(y_t^s | y_1^s, y_2^s, .., y_{t-1}^s, x)$ distribution at each time step. In essence, optimizing L_{rl} improves the expected reward of the generated output. The final loss is a linear combination of negative log-likelihood loss with the RL loss,

$$L = (1 - \alpha) \cdot L_{nll} + \alpha \cdot L_{rl}, \quad (5)$$

where α governs the strength of the RL-based loss term.As we will show in Section 4.2, we define L_{rl} based on the input tokens and thus the same trained model can generate formal and informal summary variants for a given input article.

While decoding, our framework employs a beam search algorithm to explore plausible outputs (or hypotheses) for generating the final summary. It finally outputs the hypothesis with the maximum probability of generation. However, we observe a reasonable difference in the formality scores among hypotheses with similar generation probabilities. As a part of post-processing, we therefore employ hypotheses re-ranking to further strengthen our generation following Niu et al. (2017). We pick the hypothesis with the maximum formality score among the k hypotheses with

highest generation probabilities. We also perform word-replacement using parallel formal-informal word lists curated from (Sheikha and Inkpen, 2011). This helps to tackle unwanted formal or informal words which must have been copied directly from the input article by the pointer generator frameworks (See et al., 2017). As we show later, this helps our model to better capture formality oracle.

4.1 Measuring Formality in Text

The first step towards defining our reward function is to construct an oracle to measure formality. We define formality using lexical scores (Brooke et al., 2010; Brooke and Hirst, 2014) leveraging the implementation by Niu et al. (2017) who incorporate formality into Machine Translation[1]. They report best performance using a combination of a Support Vector Machine (SVM) model with Word2Vec representations on the CTRW (Hayakawa and Ehrlich, 1994) and BEAN (Lahiri, 2015; Pavlick and Tetreault, 2016) datasets, obtaining 84.4% accuracy on the former and a Spearman's ρ of 0.662 on the latter.

First, two sets of seed words are chosen, which represent formal and informal language respectively. We use the lists curated by Sheikha and Inkpen (2011). They contain various abbreviations with their full forms and 'informal:formal' semantically similar word pairs such as 'about:approximately', 'copy:replica', 'risk:jeopardy', and 'tasty:palatable'. We combine these word lists to create a total of 667 formal and informal seeds. Next, an SVM model is trained to find a separating hyperplane between vector space representations (coming from Word2Vec model trained on Google News corpus) of these formal and informal seeds. Once the model is trained, for any given word, Euclidean distance to this hyperplane is used as a measure of word level formality. To compute the formality of a word sequence y, we use the weighted average function from Niu et al. (2017):

$$F(y) = \frac{\sum_{w_i \in y} |L(w_i)|.L(w_i)}{\sum_{w_i \in y} |L(w_i)|}, \qquad (6)$$

where $L(w_i)$ is the lexical formality score from the SVM model described above.

$F(y)$ represents the formality of the word sequence y, where larger positive values correspond to higher levels of formality and negative values represent informality in text. Using this measure for formality as an oracle, our objective now is to teach the model the intricacies of high and low levels of formality and ultimately, taking this into consideration while summary generation. This is achieved through the reward function r, which is described next.

4.2 Defining the reward function r

We propose an indicator-based rewarding setup to simultaneously benefit from the common learnings of the two models (formal and informal summaries) and incorporate feedback on the ground-truth summaries in the dataset, as against using the formality oracle $F(y)$ described in Section 4.1 directly as separate reward function for formal and informal models.

We use explicit indicators along with the input sequence and set the reward function accordingly. For training, first the oracle $F(y)$ from equation 6 is used to classify ground-truth summaries in the dataset into informal, neutral, or formal classes. We then assign two vocabulary ids (called tokens or indicators) to each class. While training, these tokens are added to the beginning and end of the input article, based on the formality class of the corresponding ground-truth summary. For instance, for a given (article, summary) pair (a, s) in the training dataset, if s is classified as formal, we add the corresponding two tokens at the beginning and end of the input a. The usage of tokens in this manner acts as indicators, providing the model with feedback on the ground-truth summary in the training stage. The usage of two tokens keeps the input **symmetric**, making it easier for both the forward and backward LSTM encoder networks to absorb the formality information at the start of generating their half of the encoding states. We then determine our reward function for RL loss L_{rl} based on the formality class of the ground-truth summary:

$$r(y^s) = \begin{cases} F(y^s) & \text{for formal } y^* \\ 0.0 & \text{for neutral } y^* \\ -F(y^s) & \text{for informal } y^* \end{cases} \qquad (7)$$

where y^* is the ground-truth summary sequence and $F(.)$ is the score from Equation 6. If the ground-truth summary is formal (as denoted by the corresponding tokens), our reward works to maximize the expected formality of the output sum-

[1] https://github.com/xingniu/computational-stylistic-variations

mary and minimize it in case of informal ground-truth summaries. Given an unseen input article, the two tokens can be added to the input sequence based on the type of output summary required. An input-dependent reward function allows the same model to generate both the summary variants, unlike the vanilla framework, which loses the opportunity to learn the commonalities in the two spaces.

4.3 Similarity to SCST method

Equation 4 employs REINFORCE algorithm (Williams, 1992) and can be seen as a modification to the Self Critical Sequence Training (SCST) (Rennie et al., 2017) which was applied to successfully optimize on ROUGE in summarization (Paulus et al., 2018). In SCST, the word with the maximum probability is greedily chosen from the output distribution at each time step, forming a greedy sequence y^b. The model uses the reward of y^b as a baseline in the loss function. Instead, our formulation can be seen as using the baseline reward of 0. This compares the formality level of the sampled sequence y^s with the perfectly neutral summary (with formality score 0.0), penalizing any sequences lying on the opposite side of the desired formality levels.

5 Experimental Setup

We evaluate our approach on its ability to generate formal and informal summaries for an input article. We compare it with several baselines using both automated metrics and crowd-sourcing experiments.

Dataset: A combined dataset from 2 domains: news and social media is used. For the former, we use the CNN/DailyMail news dataset (Hermann et al., 2015; Nallapati et al., 2016), widely used for the task of abstractive text summarization. For the latter, we use the Webis-TLDR-17 corpus (Völske et al., 2017), automatically created using $TL;DR$ tags on Reddit[2]. Figure 1 shows the distribution of lexical formality scores over these and the complete dataset (based on Equation 6). As depicted, the combination allows us to ensure that the dataset contains formality-diverse 'article:summary' pairs. For CNN/DM, the average formality is -0.097, with minimum as -2.451 and maximum as 2.62. For Reddit dataset, the average formality is -1.068, minimum -2.653 and maximum is 2.783. While the average values are negative, through a manual analysis, we found the summaries with more than -0.5 to be reasonably formal in general. We refer the readers to Section A in Supplementary material where we show some sample ground-truth summaries in the dataset along with their formality scores.

The news dataset contains $287,226$ training instances, $13,368$ validation and $11,490$ test instances. The articles have an average length of 781 tokens and multi-sentence summaries with average length of 56 tokens. We use these average values to extract a similar-sized subset of 4 million data points in the Reddit dataset and merge them with the news dataset. We filtered out poor summaries in Reddit dataset heuristically. Several summaries which contain edit: actually refer to additional information not in the article. We filter out summaries containing such keywords. Keeping only the most formal and informal pairs, the training dataset reduces to $286,358$ input-output pairs. $10,000$ pairs are held out for validation and testing, each containing data points from both domains.

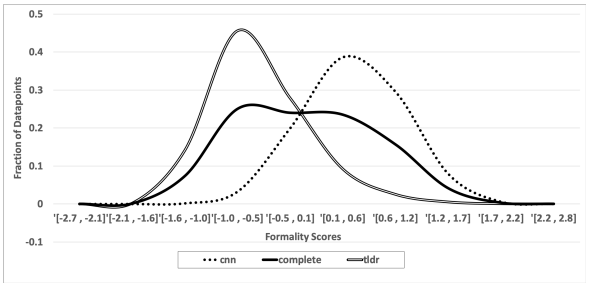


Figure 1: Distribution of lexical formality (Equation 6) in CNN/Daily Mail, Reddit and the complete dataset. Positive values on the X-axis indicate high formality and negative values indicate informality.

Hyperparameters: All methods are implemented using the pointer-generator framework described in Section 3. Following See et al. (2017), the network uses 256 hidden dimensions, embedding size as 128, vocabulary size as $50,000$, 400 maximum encoding steps and 100 maximum decoding steps. We use these hyper-parameters for all the approaches. All our models train for approximately $50,000$ iterations using a batch of size 16.

6 Automated Evaluation

We report the F1 scores for ROUGE-1, ROUGE-2, and ROUGE-L metrics, evaluating how close the generated summaries are to the reference sum-

[2]https://www.reddit.com/

	Informal Summaries					Formal Summaries				
Method	**ROUGE F-score**			**Formality**	**Improvement**	**ROUGE F-score**			**Formality**	**Improvement**
	1	**2**	**L**		(%) vs **PGen**	**1**	**2**	**L**		(%) vs **PGen**
SymVoTo (A)	**26.95**	**8.23**	**23.28**	-0.560 ± 0.902	$+6.66$	**27.00**	**8.15**	**23.23**	-0.432 ± 0.822	$+17.71$
ZeroRL (B)	24.03	6.90	21.08	-0.581 ± 0.925	$+10.66$	24.42	6.93	20.90	-0.358 ± 0.760	$+31.80$
A+B	24.75	7.27	21.49	-0.603 ± 0.923	$+14.85$	25.72	7.51	22.20	-0.399 ± 0.801	$+24.00$
Our Approach	23.02	6.54	20.19	$\mathbf{-0.727 \pm 0.846}$	$\mathbf{+38.47}$	24.8	6.72	21.58	$\mathbf{-0.052 \pm 0.738}$	$\mathbf{+90.09}$

Table 2: Performance of various ablations of the proposed approach on automated evaluation metrics in generating informal and formal summaries. Formality score is computed from the oracle (Eq. 6), averaged over the testset. **SymVoTo** refers to the use of two vocabulary tokens in a symmetrical manner, **ZeroRL** refers to the use of Zero baseline reward instead of Greedy baseline (Section 4). **A+B** combines these two methods and the complete approach further employs post-processing steps.

	Informal Summaries					Formal Summaries				
Method	**ROUGE F-score**			**Formality**	**Improvement**	**ROUGE F-score**			**Formality**	**Improvement**
	1	**2**	**L**		(%) vs **PGen**	**1**	**2**	**L**		(%) vs **PGen**
PGen	**26.96**	**8.18**	**23.18**	-0.525 ± 0.875	-	**26.96**	**8.18**	**23.18**	-0.525 ± 0.875	-
VoTo	26.61	8.17	22.99	-0.556 ± 0.883	$+5.90$	26.76	8.13	23.04	-0.452 ± 0.821	$+13.90$
StyleSum	14.84	2.06	13.03	$\mathbf{-0.762 \pm 0.815}$	$\mathbf{+45.14}$	11.18	0.63	9.93	-0.59 ± 0.800	-12.38
GreedyRL	26.00	7.53	22.39	-0.476 ± 0.827	-9.33	26.47	7.92	22.72	-0.458 ± 0.829	$+12.76$
Our Approach	23.02	6.54	20.19	-0.727 ± 0.846	$+38.47$	24.8	6.72	21.58	$\mathbf{-0.052 \pm 0.738}$	$\mathbf{+90.09}$

Table 3: Performance based on Automated Evaluation Metrics for Generating Informal and Formal Summaries. Formality score is computed from the oracle (Eq. 6), averaged over the testset. All the prior approaches were adapted for formality, as described in Section 6.2.

maries. To evaluate the efficacy of the methods in capturing formality, we report the average formality in the output summaries and corresponding percentage improvements in average formality, relative to **PGen** (See et al., 2017).

6.1 Ablation study

We performed an ablation study over our approach to analyze the effect in performance by the use of two symmetric vocabulary tokens (**SymVoTo**) and a zero-reward baseline (**ZeroRL**) separately.

For training in **SymVoTo**, three levels of formality are defined based on the scores from the formality oracle $F(y)$ (Equation 6): informal (less than -0.2), neutral (between -0.2 and $+0.2$), and formal (greater than 0.2). In our training dataset, $169,628$ summaries were tagged as informal, $31,129$ as neutral and $85,601$ as formal. While training, the two tokens are added to the input article based on the formality level of the ground-truth summary. To generate the formal or informal summaries for an unseen article, we add the corresponding two tokens to the input and pass it through the trained model.

The **ZeroRL** method is trained using the joint objective in Equation 5. However, instead of using the input-dependent reward function in Equation 7, it directly optimizes on the formality oracle. Due to the slow training speeds with policy learning, we first pre-train our network with pointer-generator method (Section 3). We further train the model with policy learning for 3000 iterations. We use a fixed weight of 0.9 for L_{rl} in Equation 5 and 0.1 for negative log likelihood loss L_{nll}. To generate formal and informal summaries in this case, we train two separate models, one where the reward function is $F(y)$ to maximize the expected formality, and second, in which the reward function is $-F(y)$, to minimize the expected formality.

Training for our own approach is similar to the vanilla RL method described above, but with three differences. First, in order to use input-dependent rewards (Equation 7), we first pre-train the model with **SymVoTo** model instead of the pointer-generator method. Secondly, once pre-trained, we optimize on our input-dependent reward function instead of directly using the formality oracle. Finally, once completely trained, the same model can be used to generate formal and informal summaries by supplying the corresponding tokens at the input. While decoding these respective summaries, we use k=4 for hypothesis reranking.

The results of this study are shown in Table 2. The models **SymVoTo** and **ZeroRL** are independently able to beat the baseline from Table 3 in capturing formality. Our approach which combines these two methods using input-dependent rewards and further employs post-processing is able to better capture the formality oracle.

6.2 Evaluation against existing baselines

Multiple style transfer and summarization models are adapted for this task as baselines. Our first baseline is the vanilla, pointer-generator network described in Section 3. It generates a single, generic summary, without using any formality information. We refer to it as **PGen**.

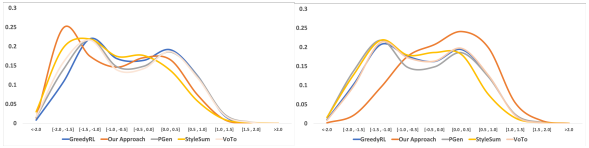
We next implement the use of single vocabulary tokens from Fan et al. (2017) in the same manner as **SymVoTo** except with the usage of one token instead of two. We refer to this method as **VoTo**.

In order to show the benefit of incorporating formality directly into the generation process, we also implement a style-transfer pipeline where we first summarize the input article and then transfer its formality to the desired level. For this purpose, we leverage the sequence to sequence implementation from Jhamtani et al. (2017) and train it on GYAFC parallel formality dataset (Rao and Tetreault, 2018). We build two models, one for formal to informal style transfer and one for informal to formal. Using the output from **PGen** method as an input to these two models, gives us the corresponding informal and formal summaries. We refer to this approach as **StyleSum**.

Next, we compare our method against the vanilla RL baseline, adapted from Paulus et al. (2018). The implementation here is similar to **ZeroRL** (Section 6.1) but with the usage of greedy baseline rewards instead of 0. We refer to this approach as **GreedyRL**.

Table 3 summarizes the results of our experiment for generating informal and formal summaries. First, we observe that as we generate more formal and informal variants, they deviate from the ground-truth summaries at lexical level, as visible in the decreasing ROUGE scores. As we later show (Section 7) through our human evaluation, this lexical difference does not affect the performance of our approach in comparison to other baselines in terms of their correctness, meaning and suitability. Secondly, we observe the desired shift in average formality scores. For both the variants, our approach better captures formality over the baseline methods. While the average formality is still on the negative spectrum for formal summaries, our method is better able to capture the oracle as compared to other baseline approaches.

The **StyleSum** method, although produces formality-diverse summaries, it fails to preserve the content of the input article, as visible by huge decline in ROUGE. This behaviour can be attributed to only a 40% overlap between the vocabulary on which the summarization and style transfer modules were trained. One of the main disadvantages for such an approach is the lack of availability of parallel corpora with the same vocabulary, for both summarization and style transfer-indicating the challenges in cascading such models and curating such corpora for these tasks.



(a) Informal summaries (b) Formal summaries

Figure 2: Distribution of formality scores of the generated summaries. Y-Axis: Fraction of datapoints, X-Axis: Intervals of formality scores.

We compare the distributions of above approaches in Figure 2. For visualization, we divide the formality score from -2.0 (more informal) to 2.0 (more formal) into 10 buckets and plot the fraction of test data points falling into these bins. The desired shifts in the distributions are visible for generating both formal and informal variants, being more profound for our approach.

Metric	Description
Formality	How formal is the given summary
Meaning Similarity	How close or similar is the meaning of the given summary with respect to the reference (ground-truth) summary
Semantic Correctness	How correct is the information present in the summary with respect to the given input article
Suitability	How well the summary suits the input article, how well it captures the key idea behind it

Table 4: Metrics considered for the qualitative analysis of the summaries generated by our approach.

7 Qualitative Evaluation

The automated evaluation is limited to comparing the summaries to a single ground-truth summary based on ROUGE metric. Hence, we further performed a crowd-sourced experiment to evaluate the quality of the generated summaries while also evaluating their formality. We compare the summaries generated by our model with those generated by **VoTo** and **GreedyRL** baseline methods. We did not consider the **Pgen** and **StyleSum** for this comparison since the former only generates a single, generic summary and the latter deviates too

Method	Informal Summaries (in %)				Formal Summaries (in %)			
	Formality	Meaning Similarity	Semantic Correctness	Suitability	Formality	Meaning Similarity	Semantic Correctness	Suitability
All instances								
VoTo	40	62	54	52	52	56	66	54
GreedyRL	52	54	66	62	62	66	66	54
CNN/DM instances								
VoTo	37.5	70.8	45.8	54.2	54.1	58.3	66.7	50
GreedyRL	54.2	54.2	79.2	58.3	62.5	66.7	58.3	54.2
TL;DR instances								
VoTo	42.3	53.8	61.5	50	50	53.8	65.4	57.7
GreedyRL	50	53.8	53.8	65.4	61.5	65.4	73.1	53.8

Table 5: Percentage improvement of the proposed approach w.r.t. baseline methods for formal and informal summary generation. Each value indicates the %age of cases where the summary by our approach was rated equal to or higher in comparison to the baseline summary. For example, in 62% of the cases (All instances), the informal summary generated by our method was rated as being closer to the reference summary in meaning (**Meaning Similarity**) with respect to the summary generated by **VoTo** method. For Informal summaries, lower score on formality is desirable and for all other comparisons, a higher score is more desirable. For all the metrics, our method either performs at par or outperforms the two baselines.

much from the content in the article, as depicted by the ROUGE scores. Table 4 describes the metrics on which we perform the comparison.

The crowd-sourced experiment is conducted via Amazon Mechanical Turk[3]. 50 samples were randomly chosen from our test data, with 24 coming from CNN/DM dataset and 26 coming from Reddit dataset. The annotators were asked to rate the summaries on a discrete scale of 1 to 5 for all our requested metrics. To avoid any inter-annotator bias, we get every annotator to rate all variants of the summary generated for each test case. In total, the summaries for each sample were rated by 5 annotators. Our comparisons between any two summary variants for the same article are based on the majority opinion of these 5 annotators. To ensure the quality of the annotations, we also ask the annotators to mark all the summaries which they saw during the survey. We reject all those assignments where this question was answered incorrectly and those with less than 150 seconds work time.

Intra-Model Comparison: The annotators rate the formality of a given summary, with 1 being the least and 5, most formal. We perform a comparative analysis between the formal and informal summaries generated by the same model. For our approach, the formal summary was rated as more formal in comparison to it's informal counterpart in 76% of the cases. For **VoTo** and **GreedyRL** method, this number drops down to 66%. Being consistent with the average formality scores in Table 3, this shows that our approach is able to produce better formality-diverse summaries.

Inter-Model Comparison: In order to measure the quality of our generated summaries, we also compare them with baseline outputs on Meaning Similarity, Semantic Correctness, and Suitability (Table 4), all being key requirements in any summarization system. We compare the (in)formal summaries generated by our system with the corresponding (in)formal summaries generated by the two baseline systems. For all these metrics, higher values are more desirable for both formal and informal variants. However, for comparison on Formality, when comparing informal summaries, lesser values are desirable and while comparing formal summaries, higher are more desirable. The results of our comparative study for these metrics are presented in Table 5. All the values represent the percentage of cases where our summary is rated to be better than the corresponding baseline summary by the majority of annotators. We also report the values for each dataset separately. While our method does show a decline in ROUGE scores in comparison to these methods (Table 3), probably due to diversion from the ground-truth summaries, this decline does not translate to the quality metrics in our human evaluation. We observe that our summaries either perform at par or outperform the baseline summaries on all four metrics. We conclude that our approach produces better formality-diverse summaries, while still surpassing other methods on summarization quality.

8 Conclusion

We presented a framework to generate formality-tailored abstractive summaries for a given input article. Our approach employs reinforcement learning to train the model with formality feedback on both ground-truth and sampled summaries to-

[3]https://www.mturk.com/

gether. Automatic and human evaluations show that although we observe some deviation from the ground-truth summaries with respect to baseline methods, the approach is effective in generating formality-diverse summaries while still preserving the meaning, semantic correctness and suitability. Given a suitable oracle, the proposed methodology can be easily extended to other psycho-linguistic preferences such as politeness. We plan to perform this incorporation of other such preferences that can arise in textual content.

References

Mikel Artetxe, Gorka Labaka, Eneko Agirre, and Kyunghyun Cho. 2017. Unsupervised neural machine translation. *arXiv preprint arXiv:1710.11041*.

Siddhartha Banerjee, Prasenjit Mitra, and Kazunari Sugiyama. 2015. Multi-document abstractive summarization using ilp based multi-sentence compression. In *IJCAI*, pages 1208–1214.

Taylor Berg-Kirkpatrick, Dan Gillick, and Dan Klein. 2011. Jointly learning to extract and compress. In *49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*.

Julian Brooke and Graeme Hirst. 2014. Supervised ranking of co-occurrence profiles for acquisition of continuous lexical attributes. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 2172–2183.

Julian Brooke, Tong Wang, and Graeme Hirst. 2010. Automatic acquisition of lexical formality. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 90–98. Association for Computational Linguistics.

Niyati Chhaya, Kushal Chawla, Tanya Goyal, Projjal Chanda, and Jaya Singh. 2018. Frustrated, polite or formal: Quantifying feelings and tone in emails. In *Second Workshop on Computational Modeling of Peoples Opinions, Personality, and Emotions in Social Media, NAACL HLT*.

Sumit Chopra, Michael Auli, Alexander M Rush, and SEAS Harvard. 2016. Abstractive sentence summarization with attentive recurrent neural networks. In *HLT-NAACL*, pages 93–98.

Cristian Danescu-Niculescu-Mizil, Moritz Sudhof, Dan Jurafsky, Jure Leskovec, and Christopher Potts. 2013. A computational approach to politeness with application to social factors. *arXiv preprint arXiv:1306.6078*.

Angela Fan, David Grangier, and Michael Auli. 2017. Controllable abstractive summarization. *arXiv preprint arXiv:1711.05217*.

Jessica Ficler and Yoav Goldberg. 2017. Controlling linguistic style aspects in neural language generation. In *Proceedings of the Workshop on Stylistic Variation*, pages 94–104.

Katja Filippova. 2010. Multi-sentence compression: Finding shortest paths in word graphs. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 322–330.

Pierre-Etienne Genest and Guy Lapalme. 2011. Framework for abstractive summarization using text-to-text generation. In *Proceedings of the Workshop on Monolingual Text-To-Text Generation*.

Caglar Gulcehre, Sungjin Ahn, Ramesh Nallapati, Bowen Zhou, and Yoshua Bengio. 2016. Pointing the unknown words. In *54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

Mengqiao Han, Ou Wu, and Zhendong Niu. 2017. Unsupervised automatic text style transfer using lstm. In *National CCF Conference on Natural Language Processing and Chinese Computing*, pages 281–292. Springer.

Samuel I Hayakawa and Eugene Ehrlich. 1994. *Choose the right word*. Harper Perennial.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems*, pages 1693–1701.

Zhiting Hu, Zichao Yang, Xiaodan Liang, Ruslan Salakhutdinov, and Eric P Xing. 2017. Toward controlled generation of text. In *International Conference on Machine Learning*, pages 1587–1596.

Harsh Jhamtani, Varun Gangal, Eduard Hovy, and Eric Nyberg. 2017. Shakespearizing modern language using copy-enriched sequence to sequence models. In *Proceedings of the Workshop on Stylistic Variation*, pages 10–19.

Kundan Krishna and Balaji Vasan Srinivasan. 2018. Generating topic-oriented summaries using neural attention. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, volume 1, pages 1697–1705.

Shibamouli Lahiri. 2015. SQUINKY! A Corpus of Sentence-level Formality, Informativeness, and Implicature. *CoRR*, abs/1506.02306.

Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. 2017. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. In *AAAI*, pages 3075–3081.

Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Caglar Gulcehre, and Bing Xiang. 2016. Abstractive text summarization using sequence-to-sequence rnns and beyond. In *The 20th SIGNLL Conference on Computational Natural Language Learning*.

Xing Niu, Marianna Martindale, and Marine Carpuat. 2017. A study of style in machine translation: Controlling the formality of machine translation output. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2814–2819.

Xing Niu, Sudha Rao, and Marine Carpuat. 2018. Multi-task neural models for translating between styles within and across languages. *arXiv preprint arXiv:1806.04357*.

Romain Paulus, Caiming Xiong, and Richard Socher. 2018. A deep reinforced model for abstractive summarization.

Ellie Pavlick and Joel Tetreault. 2016. An empirical analysis of formality in online communication. *Transactions of the Association for Computational Linguistics*, 4:61–74.

Sudha Rao and Joel Tetreault. 2018. Dear sir or madam, may i introduce the gyafc dataset: Corpus, benchmarks and metrics for formality style transfer. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, volume 1, pages 129–140.

Steven J Rennie, Etienne Marcheret, Youssef Mroueh, Jarret Ross, and Vaibhava Goel. 2017. Self-critical sequence training for image captioning. In *CVPR*, volume 1, page 3.

Alexander M Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *Conference on Empirical Methods in Natural Language Processing*.

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)*, volume 1, pages 1073–1083.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Controlling politeness in neural machine translation via side constraints. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 35–40.

Fadi Abu Sheikha and Diana Inkpen. 2011. Generation of formal and informal sentences. In *Proceedings of the 13th European Workshop on Natural Language Generation*.

Tianxiao Shen, Tao Lei, Regina Barzilay, and Tommi Jaakkola. 2017. Style transfer from non-parallel text by cross-alignment. In *Advances in Neural Information Processing Systems*, pages 6833–6844.

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.

Alexey Tikhonov and Ivan P Yamshchikov. 2018. What is wrong with style transfer for texts? *arXiv preprint arXiv:1808.04365*.

Zhaopeng Tu, Zhengdong Lu, Yang Liu, Xiaohua Liu, and Hang Li. 2016. Modeling coverage for neural machine translation. In *54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

Michael Völske, Martin Potthast, Shahbaz Syed, and Benno Stein. 2017. Tl; dr: Mining reddit to learn automatic summarization. In *Proceedings of the Workshop on New Frontiers in Summarization*, pages 59–63.

Lu Wang and Claire Cardie. 2013. Domain-independent abstract generation for focused meeting summarization. In *ACL (1)*, pages 1395–1405.

Ronald J Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256.