

CARNEGIE MELLON UNIVERSITY

Controllable Text Generation

Should machines reflect the way humans interact in society?

Thesis Proposal

by

Shrimai Prabhumoye

Thesis proposal submitted in partial fulfillment
for the degree of Doctor of Philosophy

Thesis committee:

Alan W Black (co-chair)

Ruslan Salakhutdinov (co-chair)

Yulia Tsvetkov

Jason Weston (Facebook AI Research)

March 2020

©Copyright by Shrimai Prabhumoye

Abstract

The 21st century is witnessing a major shift in the way people interact with technology and Natural Language Generation (NLG) is playing a central role. Users of smartphones and smart home devices now *expect* their gadgets to be aware of their situation, and to produce natural language outputs in interactions. This thesis identifies three aspects of human communication to make machines sound human-like - style, content and structure. This thesis provides deep learning solutions to controlling these variables in neural text generation. I first outline the various modules which could be manipulated to perform effective controllable text generation. I provide a novel solution for style transfer using back-translation and introduce two new tasks to leverage information from unstructured documents into the generation process. I also provide a new elegant design for the sentence ordering task to learn effective document structures. At the end, I provide a discussion on the ethical considerations of the applications of controllable text generation. As proposed work, I plan to: (i) provide an empirical understanding of the various techniques of controllable text generation, (ii) provide computational understanding of style and build useful style representations, (iii) design an efficient way of content grounded generation, and (iv) explore the broader impact of controllable text generation.

Contents

Abstract	i
1 Introduction	1
1.1 Overview	4
1.2 Timeline	5
2 Controllable Text Generation Techniques	6
2.1 Generation Process	6
2.2 External Input	7
2.3 Sequential Input	10
2.4 Generator Operations	11
2.5 Output	12
2.6 Training Objective	14
2.7 Proposed Work	16
3 Style	17
3.1 Methodology	18
3.1.1 Meaning-Grounded Representation	20
3.1.2 Style-Specific Generation	20
3.2 Style Transfer Tasks and Datasets	22
3.3 Evaluation	25
3.3.1 Style Transfer Accuracy	25
3.3.2 Preservation of Meaning	26
3.3.3 Fluency	27
3.4 Results	27
3.5 Related Work	29
3.6 Proposed Work	30
3.6.1 Style Representations	30
3.6.2 Understanding Style	32

4	Content	34
4.1	Methodology	36
4.1.1	Generative models	37
4.1.2	Extractive models	38
4.2	Datasets	39
4.2.1	Grounded Wikipedia Edit Generation	39
4.2.2	Grounded Dialog Generation	41
4.3	Results	45
4.3.1	Automated Evaluation	45
4.3.2	Human Evaluations	47
4.4	Related Work	50
4.5	Proposed Work	51
5	Structure	54
5.1	Methodology	55
5.1.1	Topological Sort	55
5.1.2	Constraint Learning	56
5.2	Experiments	57
5.2.1	Datasets	57
5.2.2	Baselines	57
5.2.3	Evaluation Metric	58
5.3	Results	59
5.4	Related Work	61
6	Ethical Considerations	63
6.1	Principles of Ethics	63
6.2	Broader impact of Controllable Text Generation	67
6.3	Proposed Work	67
A	Appendix A	69
B	Appendix B	74
C	Appendix C	85
	Bibliography	88

Chapter 1

Introduction

“The common misconception is that language has to do with words and what they mean. It doesn’t. It has to do with people and what they mean.”

Herb Clark and Michael Schober, 1992

One of the important goals of artificial intelligence (AI) is to model and simulate human intelligence. Modeling human interactions is a sub-goal in the path of answering the larger question on human intelligence. Natural Language Generation (NLG) is an important aspect of modeling human communications. NLG by definition focuses on producing human languages¹ from non-linguistic machine representations of knowledge. The process of generating language poses an interesting question of how information is best communicated between a machine and a human.

This thesis is inspired by the research question *“Should machines reflect the the way humans interact in society?”*. I have identified three aspects of human communication that I am interested in using for generation: (1) Style (2) Content and (3) Structure. Style is used in human communication to convey specific goals effectively and also to define social identities. All human communications carry some degree of information in them, which I call content. Even one way communications or documentations such as blogs, memos, reports etc enclose relevant information. The ordering of information in these communications is structure and each of these communication goals requires different structures to achieve the desired effect.

Most human practices display style (Coupland, 2007). For example, fashionable style is reflected in the choice of clothes and accessories we wear, architectural style is exhibited in the choice of raw materials used, color, design plans etc of the construction, culinary style is demonstrated in the raw materials, size and color of crockery, etc. Similarly, linguistic style is expressed in the choice of words or phrases as well syntactic structures used to convey a piece of information. Note that ‘style’ in computational linguistics is a loaded term and I don’t partake in

¹Although the philosophy and techniques mentioned in this thesis are applicable to any natural language, I focus only on English (#BenderRule).

disambiguating its usage. I define style as a group of natural language sentences that belong to a particular class or label. I focus on controlling the neural generation process to adhere to a specific style. In particular, I propose the novel approach of using neural back-translation for building a hidden representation that has reduced stylistic elements but is grounded in semantic meaning to the input sentence. I finally use an adversarial training objective to ensure that the generation complies with the target style.

Human communication by definition is a process by which individuals exchange information and influence one another through a common system of symbols and signs (Higgins and Semin, 2001). This behavior is however not mirrored in natural language generation systems. Typically, models hallucinate information to be generated as they are not conditioned on any external source of knowledge. Generating natural language from schematized or structured data such as database records, slot-value pair, Wikipedia Infobox etc has been explored in prior work (Mei et al., 2016; Wen et al., 2015; Lebrecht et al., 2016). A lot of information resides in unstructured format in the form of books, Encyclopedias, news articles, Wikipedia articles etc. I focus on leveraging this information to guide the generation process to include relevant pieces in the generated text. I propose various neural models to incorporate both context and an external source of information into the generation step.

Human beings effortlessly produce complicated pieces of text that are well connected and appropriately ordered (Hovy, 1993). Most effective human communication is not in the form of randomly ordered information but it is well planned and structured. In spite of the recent advances in natural language processing (NLP), NLG systems have not gained the ability to plan and organize multiple sentences. I focus on solving the sentence ordering sub-task which involves ordering the information in a document. Sentence ordering is the task of arranging the sentences of a given text in the correct order. In particular, I pose this task as a constraint solving problem and leverage rich sentence representations provided by pre-trained language models to design these constraint.

Reiter and Dale (2000) detail seven sub-tasks which are conceptually distinct to describe the generation process. These sub-tasks can be modelled separately or in some cases they may interleave. In (Reiter and Dale, 2000), these seven sub-tasks are primarily characterized as content or structure tasks. Contrary to this characterization, I connect the style, content and structure aspects of this thesis to the different sub-tasks in (Reiter and Dale, 2000). The seven sub-tasks are: (1) *Content Determination* is the sub-task of deciding what information needs to be communicated in the generated piece of text. (2) *Document Structuring* is the sub-task of grouping similar content together and then deciding the relations between the groups to generate a coherent structured text. (3) *Lexicalization* is the sub-task of choosing specific set of phrases or other linguistic features such as syntactic constructs to express the selected content in the desired manner. (4) *Referring Expression Generation* is involved with selecting the desired expressions to be used to refer to entities. (5) *Aggregation* is concerned with mapping document structures onto linguistic structures such as sentences and paragraphs. This sub-task can also decide the ordering of information that has to be generated. (6) *Linguistic Realisation* is the

sub-task of converting abstract representations of sentences into the real text. (7) *Structure Realisation* is the sub-task of converting abstract structures such as paragraphs and sections into mark-up symbols and segments understood by humans.

Style is related to the *lexicalization* sub-task and I control the generation process by selecting the desired phrases or other linguistic resources. Content is *content determination* sub-task and I guide the generation process with explicit information that is needed in the generated text. I focus on understanding document structures and hence appeal to the *document structuring* sub-task and provide an elegant solution for ordering if sentences for the *aggregation* sub-task in my exploration of Structure. Note that the *linguistic realisation* sub-task is already solved by sequence-to-sequence frameworks which generate sentence from a hidden representation of it. The sub-task of deciding document structure boundaries in *structure realisation* and *referring expression generation* is left for future work.

At a minimum, controlling these three aspects of communication can be used for tasks such as:

- Dialogue systems - controlling the persona of the system, various aspects of the response such formality, authority etc, and grounding conversation on unstructured content.
- Story generation - introducing NLG into audience-appropriate narrative texts, generating stories from given plots or events.
- Report generation - pulling disparate source documents into a coherent unified whole, which can use a shared set of sources to generate a variety of genres:
 - News articles covering current events with historical context.
 - Wiki articles summarizing a topic’s evolution over time.
 - Scientific article summaries highlighting key findings on a topic.

Human communications also sometimes carry debatable features such as usage of swear words and obscenity in language (McEnery, 2005), or using the power of language to target minority groups to project social biases and reinforce stereotypes on people (Fiske, 1993). Providing fine grained control on style, content and structure in generated text runs the risk of generating language which has undesirable consequences such as spewing hate or targeting groups to promote violence or social disorder. In the last part of my thesis, I open the discussion on the ethical considerations of controllable text generation. While mirroring the style, content and structure aspects of human communication, it is also important to think about the scenarios when we don’t want the machines to reflect human interactions. Particularly, I want to focus on how I can use this for social causes such as generating demographically balanced data and generating powerful narratives that change human attitudes towards stereotypes.

1.1 Overview

Thesis Statement *Controlling style, content and structure leads to human-like generations which should be used with ethical considerations.*

This thesis presents a background on controllable text generation techniques, collection of work on each of the above mentioned three aspects of communication and an exploration of the ethical considerations of this technology.

Controllable Text Generation Techniques (Chapter 2): I start by providing the necessary technical background for understanding this thesis. In this chapter, I connect the different works in controllable text generation and collate the knowledge about the similarities of these tasks and techniques. I organize the prior work and propose a new schema which contains five modules that can be changed to control the generation process - external output module, sequential input module, generator module, output module and the training objective module. I lay grounds to the different theories of representing control vectors and incorporating them into the generation process as well as provide a qualitative assessment of these techniques.

Proposed Work: I propose to select three controllable text generation tasks for example style transfer, content grounded dialogue generation, recipe generation, plot driven story generation etc and explore the different techniques of controlling each of the five modules. I want to provide empirical insight into which of the described techniques work better or worse for different tasks. It is also possible that a combination of techniques is suitable for some tasks. With these set of experiments, I wish to provide new directions to explore for controllable text generation.

Style (Chapter 3): This chapter talk about the importance of *style* aspect in human communication. I describe the novel approach of back-translation to perform style transfer in non-parallel data for various tasks such as gender transfer, political slant transfer and sentiment modification. I outline and also provide insights in both automatic and human evaluations for three dimensions of accessing style transfer methods: style transfer accuracy, preservation of meaning and fluency.

Proposed Work: In spite of the large focus on modeling new style transfer techniques, there is lack of understanding a style itself. I propose to design computational methods to learning the different forms in which the style becomes realized. In particular, I want to focus on lexical and structural understanding of style. I also propose to extract a good representation of style given a sample of sentences of that style. This could be useful in cross domain style classification or style classification when number of samples of that style are low.

Content (Chapter 4): This chapter provides an overview on the different tasks that exist for content grounded generation. I propose two new tasks for grounded generation in two different domains. First is Wikipedia edit generation task which is concerned with generating a Wikipedia update given an external news article and the Wikipedia article context. Second is

dialogue response generation which involves generating a response based on the knowledge from an external source and the current dialogue history. I also provide an extensive evaluation for models trained for the Wikipedia edit generation task. I propose two new human evaluations for this task and adopt absolute human evaluation from prior work.

Proposed Work: I propose a new attention based model to perform the task Wikipedia edit generation and dialogue response generation from an external source of document. I also propose a new evaluation metric to determine the fidelity of the generations.

Structure (Chapter 5): In this chapter, I provide an overview of the different techniques used to capture document structures. I particularly focus on the sub-task of sentence ordering and propose a new framing of this problem as a constraint solving task. I also introduce a new model based on the new design of the problem. I suggest a new human evaluation for this task which analyzes the human choices for predicted orders in comparison to the reference orders.

Ethics (Chapter 6): With this chapter I start the discussion on the ethical considerations of controllable text generation. I give an overview of the various ethical issues pertaining to NLP and the need to discuss these issues. I also provide a summary of two principles of ethical science - *generalization principle* and *utilitarian principle*.

Proposed Work: I propose to use controllable text generation for generating balanced dataset for downstream application tasks. I also propose to analyze the broader impact of controllable text generation and how it can be used to change human attitude towards stereotypes.

1.2 Timeline

PhD Timeline for 2020-21

May – Aug 2020	• Internship at Salesforce Empirical analysis of the various techniques used for three controllable text generation tasks (Ch 2) Develop a content grounded generation method (Ch 4)
Sep – Dec 2020	• Work on proposed work in style (Ch 3) Extract useful style representations Understand the linguistic features that define a style
Jan – Feb 2021	• Analysis of ethical concerns of controllable generation (Ch 6)
Mar – Apr 2021	• Apply for job and write thesis

Chapter 2

Controllable Text Generation Techniques

Neural controllable text generation is an important area gaining attention due to its plethora of applications. Currently, the work in this space is accomplished as independent tasks and techniques. In this chapter, I want to organize the prior work and connect it together by proposing a new schema containing five modules. The main advantage of this schema is that it can be used with any algorithmic paradigm like sequence-to-sequence, probabilistic models, adversarial methods, reinforcement learning etc. I present an overview of the various techniques used to modulate each of these five modules to provide with control of attributes in the generation process. I also provide an analysis on the advantages and disadvantages of these techniques and open paths to develop new architectures based on the combination of the modules described here.

2.1 Generation Process

Given a corpus of tokens $\mathbf{U} = (\mathbf{u}_1, \dots, \mathbf{u}_T)$, the task of language modeling is to estimate the joint probability $P(\mathbf{U})$, which is often auto-regressively factorized as $P(\mathbf{U}) = \prod_t^T P(\mathbf{u}_t | \mathbf{u}_{<t})$. For this thesis, I consider conditional language model which has an input or a *source* sequence \mathbf{U} and an output or *target* sequence \mathbf{Y} to be generated. In this case we model the probability of the *target* sequence conditioned on the *source* sequence given by $P(\mathbf{Y} | \mathbf{U}) = \prod_t^T P(y_t | \mathbf{U}, y_{<t})$. Sequence-to-sequence models which refers to the broader class of models that map one sequence to another, are generally used to build conditional language models. The representation of the probability $P(\mathbf{U})$ of the *source* sequence given by a neural model is denoted by \mathbf{h}_e . The initialization of the standard generation process \mathbf{h}_0 is equal to \mathbf{h}_e . The generation of the target tokens of the sequence \mathbf{Y} unfolds as a time series where each token y_t is generated at a time step t .

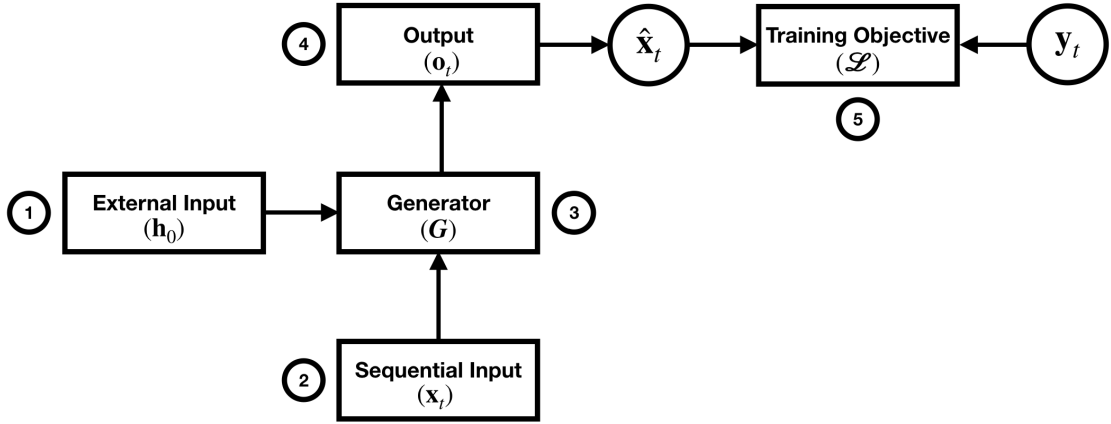


FIGURE 2.1: Modules that control the generation process

At a given time step t , a generative model performs some set of operations G by taking in as input the previous hidden state \mathbf{h}_{t-1} and the input \mathbf{x}_t . Note that the hidden state \mathbf{h}_{t-1} represents the probability of the tokens generated up to time step t as well as the *source* sequence \mathbf{U} . \mathbf{x}_t is the word embedding of the token y_{t-1} . The generator produces an output hidden state \mathbf{h}_t at the current time step. In the standard generation process the output state \mathbf{o}_t is equal to the hidden state \mathbf{h}_t . \mathbf{o}_t is projected to the vocabulary space using a linear transform given by $\mathbf{W}_o \mathbf{o}_t + \mathbf{b}_o$ which is used to predict token $\hat{\mathbf{x}}_t$ using decoding strategies. Typically an argmax function is used as a decoding strategy which means that the token with the highest probability at the current time step is predicted. The ground truth token to be generated is denoted by \mathbf{y}_t and a loss \mathcal{L} is computed by comparing \mathbf{y}_t to $\hat{\mathbf{x}}_t$.

In the remainder of the chapter, I provide an overview of the schema which contains five modules that can be used for controlling the generation process (shown in Figure 2.1):

1. External Input \mathbf{h}_0 , which is the initialization of the generative model
2. Sequential Input \mathbf{x}_t at each time step
3. Generator Operations
4. Output \mathbf{o}_t of the generator
5. Training Objective

2.2 External Input

In this section I discuss the different techniques which can be used to control the generation process by controlling \mathbf{h}_0 . This is marked as position 1 in Figure 2.1.

Arithmetic or Linear Transform: One of the easiest ways to control the generation is to concatenate a control vector \mathbf{s} to output of the encoder. Let the output of the encoder be \mathbf{h}_e (described in §2.1), then the initialization of the decoder \mathbf{h}_0 will be $[\mathbf{h}_e; \mathbf{s}]$, where $[a; b]$ denotes concatenation. Here, the control vector \mathbf{s} would provide the generator with a strong signal to guide the generation process.

Fu et al. (2017) use this technique to control the style representation for their generator. The encoder builds representation that is devoid of the style and only retains content. The control vector for style is then concatenated to the encoder representation to initialize the decoder. This technique is commonly used in (Ghazvininejad et al., 2018; Zhou et al., 2018) to concatenate information from external sources to dialogue context to generate dialogue responses. Chandu et al. (2019b) concatenate personality representation \mathcal{P} derived from a separate corpus to generate visual stories. They also experiment with a simple arithmetic operation on \mathbf{h}_e given by $\mathbf{h}_0 = \mathbf{h}_e - \mathcal{S} + \mathcal{P}$ to get the initialization of the generator (here \mathcal{S} denotes the average representation of the story). They observed that while concatenation technique is better at preserving the meaning of the generated story, the arithmetic operation provides a better signal of the personality for the generation process.

Hoang et al. (2016) uses both the concatenation technique as well as performs a linear transform of \mathbf{s} to obtain \mathbf{h}_0 for language modelling task. The control vectors in this case represents meta data such as key-words, topics etc. In case of the linear transform $\mathbf{h}_0 = \tanh(\mathbf{W}_1 \mathbf{h}_e + \mathbf{W}_2 \mathbf{s} + \mathbf{b})$. The paper also explores adding the control vector to the encoder representation ($\mathbf{h}_0 = \mathbf{h}_e + \mathbf{s}$).

In case of addition, the resulting \mathbf{h}_0 would be averaged representation of the input representation \mathbf{h}_e and \mathbf{s} . Information could be lost in this case as control is not explicit. In case of concatenation, if the size of the control vector \mathbf{s} is too small compared to the context vector \mathbf{h}_e , then \mathbf{s} is over-shadowed by \mathbf{h}_e and the generator will not be able to pay attention to \mathbf{s} . Hence it is important to choose comparable dimensions for these two vectors. But this increases the size of model considerable and could be quite costly. Linear transform avoids these issues and performs better than the other two techniques for Hoang et al. (2016).

Stochastic Changes: Kingma and Welling (2014) introduce variational auto-encoder (VAE), where you can stochastically draw a continuous latent variable \mathbf{z} from a Gaussian distribution. The initialization of the generator \mathbf{h}_0 is based on this latent variable which is drawn. Bowman et al. (2016) use this concept for generating sentences from this continuous latent representation. This process of changing the encoder state \mathbf{h}_e is can only be used with Kullback-Leibler (KL) Divergence training objective described in (§2.6).

In (Wang et al., 2019b), VAE is used to guide the generation process with topics of a document. A gaussian mixture model is used to incorporate topics into latent variables. In (Xu et al., 2019), VAE is used to control for sentiment attribute in style transfer task by constraining the posterior mean to a learned probability simplex.

Such a design of controllable text generation works when the control attributes can be represented as latent variables for example style, topics, strategies etc. This design will not work for content grounded text generation tasks where specific information, keywords or entities have to guide the generation process.

Decompose: You can decompose the encoder representation \mathbf{h}_e into multiple subspaces, each of which signifies a different attribute you would like to control. [Liu and Lapata \(2018\)](#) split the encoder representation \mathbf{h}_e into two components, one which represents the structure in the document and the other represents the semantic information. This formulation was used by [\(Balachandran et al., 2020\)](#) for controlling structure in abstractive summarization. This work performs the split with respect to the dimensions of \mathbf{h}_e . The method forces the first n dimensions of \mathbf{h}_e to capture meaning and the latter to capture structure. [Balachandran et al. \(2020\)](#) also show quantitative and qualitative analysis on the types of structures of documents learnt by this technique.

[Romanov et al. \(2019\)](#) decompose the encoder representation \mathbf{h}_e into a form vector \mathbf{f} and a meaning vector \mathbf{m} . During the training phase, a *discriminator* enforces \mathbf{m} to not carry any information about the form using an adversarial loss and a *motivator* is used for a motivational loss that encourages \mathbf{f} to carry the information about the form. The generation process can then be guided to adhere to the desired target form. As opposed to splitting \mathbf{h}_e with respect to dimensions, this work learns subspaces \mathbf{W}_m and \mathbf{W}_f given by $\mathbf{m} = \tanh(\mathbf{W}_m \mathbf{h}_e + \mathbf{b}_m)$ and $\mathbf{f} = \tanh(\mathbf{W}_f \mathbf{h}_e + \mathbf{b}_f)$ respectively. When \mathbf{h}_e is projected on \mathbf{W}_m , we get the meaning vector \mathbf{m} and similarly when it is projected on \mathbf{W}_f we get the form vector \mathbf{f} . This work shows qualitatively how \mathbf{m} and \mathbf{f} are learnt in the subspaces using t-SNE plots. It also shows quantitatively the use of \mathbf{m} and \mathbf{f} in downstream paraphrase detection tasks. This is an excellent method in building interpretable representations for control attributes. Although, the effectiveness of this technique is not yet proven in the style transfer task or the abstractive summarization task. In both the above mentioned works, the models learn interpretable representations of control attributes but were not able to beat state of the art methods in their respective tasks. It is also worth noting that learning good decomposed vectors is especially hard when no supervision is provided on what the decomposed components are supposed to learn.

This technique works well when the representation space of the input \mathbf{x} can be decomposed into subspaces which represent different control attributes. This means that the input \mathbf{x} needs to contain signal of the control attributes. It will not work when the control attributes need to be externally provided. For example in case of content grounded generation tasks described in [\(Prabhumoye et al., 2019; Dinan et al., 2018; Zhou et al., 2018\)](#), the input may not necessarily contain the content that needs to be generated. A separate input of the content to be generated is provided in these cases.

External Feedback: A regularizer is often used to control the external input \mathbf{h}_0 to the generator. In many cases, an adversarial loss to manipulate the latent space is used as an external

feedback mechanism. This essentially controls the latent space of the encoder which is eventually provided as an initialization to the generator. In (Fu et al., 2018), a multi-layer perceptron (MLP) is used for predicting the style labels from \mathbf{h}_0 . Similarly, the adversarial loss is also used in (Wang et al., 2019a) to control the latent representation \mathbf{h}_0 for style attributes. In (Romanov et al., 2019), an adversarial loss is used to ensure that the meaning representation \mathbf{m} does not carry any style signals. The adversarial loss is obtained by training a discriminator which takes as input a representation \mathbf{m} and tells if it carries the target style signal. Similarly, this work also employs a motivator loss which is the opposite of the adversarial loss to ensure that the style representation \mathbf{f} actually does carry the stylistic information. John et al. (2019) use multiple losses to control the style and content information represented in \mathbf{h}_0 .

2.3 Sequential Input

In this section I discuss the different techniques which can be used to control the generation process by controlling the sequential input \mathbf{x}_t to the decoder at each time step. This is marked as position 2 in Figure 2.1.

Arithmetic or Linear Transform: Similar to changing the initialization, we can change the input to the decoder by concatenating the information at each time step with some additional control vector \mathbf{s} . Typically, teacher forcing method (Williams and Zipser, 1989) is used to train the generator. At time step t , the generator takes as input the word embedding \mathbf{x}_t of the word that was predicted at step $t - 1$ and predicts the word to be generated \mathbf{y}_t at the current time step. Note that $\mathbf{x}_t = \mathbf{y}_{t-1}$. The input \mathbf{x}_t can be concatenated with \mathbf{s} at each time step to control the generation process. Hence, $\tilde{\mathbf{x}}_t = [\mathbf{x}_t; \mathbf{s}]$.

Noraset et al. (2017), use this technique in the task of definition modeling. They concatenate word embedding vector \mathbf{s} of the word to be defined at each time step of the definition generation process. Unfortunately, for this task, this technique has not proved to be effective compared to other techniques of controlling the generation. Zhou et al. (2018) concatenate the hidden representation of the external source of information \mathbf{s} to each time step of dialogue response generation. Similarly, Prabhumoye et al. (2019) also concatenate the hidden representation of the external source of information \mathbf{s} to each time step of Wikipedia update generation process. In this work as well, this results of this technique were not as impressive as simple concatenating the control context to the input of the encoder. Harrison et al. (2019) concatenate a side constraint \mathbf{s} which represents style and personality into the generation process. For this task of generating language from meaning representations with stylistic variation, this method performed better than conditioning the encoder with side constraint in terms of BLEU metric. Chandu et al. (2019b) also concatenate the personality representation \mathcal{P} at each time step of the story generation process. This is used to control the personality of the visual stories. In addition to concatenation, this work proposes to modify the sequential input as $\tilde{\mathbf{x}}_t = \mathbf{x}_t - \mathcal{S} + \mathcal{P}$ (here \mathcal{S} denotes the average representation of the story and \mathcal{P} denotes the representation of

the personality). The latter technique is better at generating personality conditioned stories than the concatenation technique. Neither of these techniques prove to be conclusively better than making similar changes to the external input module (§2.2).

2.4 Generator Operations

This module takes in the external input \mathbf{h}_0 , the sequential input \mathbf{x}_t at time step t and performs computation to return an output \mathbf{o}_t . Different set of operations can be performed to compute \mathbf{o}_t which are enlisted below. You can also decide to change the operations based on the control vector \mathbf{s} to compute \mathbf{o}_t . This is shown as position 3 in Figure 2.1.

Recurrent Neural Networks: Recurrent Neural Networks (RNNs) are designed to model sequential information. RNNs perform the same operations for every element of a sequence, with the output depending on previous computations. This recurrence serves as a form of memory. It allows contextual information to flow through the network so that relevant outputs from previous time steps can be applied to network operations at the current time step. Theoretically, RNNs can make use of information in arbitrarily long sequences, but empirically, they are limited to looking back only a few steps.

The Long Short-Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997) units are a type of RNNs that have additional ‘memory cell’ apart from standard units of basic RNNs. The memory cell can maintain information in memory for long periods of time. A set of gates is used to control when information enters the memory, when it’s output, and when it’s forgotten. This architecture lets them learn longer-term dependencies. The vanishing gradient problem of RNNs is resolved here. Gated Recurrent Units (GRUs) (Cho et al., 2014) are similar to LSTMs, but use a simplified structure designed to adaptively capture dependencies of different time scales. They also use a set of gates to control the flow of information, but they don’t use separate memory cells, and they use fewer gates.

RNNs, LSTMs and GRUs are commonly used to model sequence-to-sequence controllable text generation tasks (Prabhumoye et al., 2019; Rao and Tetreault, 2018; See et al., 2017; Zhou et al., 2018; Fu et al., 2017).

Transformer: Transformers are proposed by (Vaswani et al., 2017) and they rely on attention mechanism to draw global dependencies between input and output. The Transformer uses stacked self-attention and point-wise, fully connected layers for both the encoder and decoder. The encoder stacks N identical layers, each of which has two sub-layers. The first sub-layer is a multi-head self-attention mechanism (described in §2.5), and the second sub-layer is a positionwise fully connected feed-forward network. Each sub-layer uses residual connections around each of the sub-layers, followed by layer normalization. The decoder has an additional third sub-layer, which performs multi-head attention over the output of the encoder stack.

Pre-trained models: Recently pre-trained conditional language models are used for text generation like GPT (Radford et al., 2018), GPT2 (Radford et al., 2019), XLNet (Yang et al., 2019), etc. Several works have fine-tuned the pre-trained models for downstream controllable text generation tasks (Sudhakar et al., 2019; Dinan et al., 2018; Urbanek et al., 2019).

Controlled Generator Operations: Gan et al. (2017) propose a variant of the LSTM model, named factored LSTM, which controls style representation in image caption task. The parameters of the LSTM module which are responsible to transform the input \mathbf{x}_t are factored into three components \mathbf{U} , \mathbf{S} and \mathbf{V} . The operations of the input (\mathbf{i}_t), forget (\mathbf{f}_t) and output gate (\mathbf{o}_t) are given by:

$$\begin{aligned}\mathbf{i}_t &= \text{sigmoid}(\mathbf{U}_{ix}\mathbf{S}_{ix}\mathbf{V}_{ix}\mathbf{x}_t + \mathbf{W}_{ih}\mathbf{h}_{t-1}) \\ \mathbf{f}_t &= \text{sigmoid}(\mathbf{U}_{fx}\mathbf{S}_{fx}\mathbf{V}_{fx}\mathbf{x}_t + \mathbf{W}_{fh}\mathbf{h}_{t-1}) \\ \mathbf{o}_t &= \text{sigmoid}(\mathbf{U}_{ox}\mathbf{S}_{ox}\mathbf{V}_{ox}\mathbf{x}_t + \mathbf{W}_{oh}\mathbf{h}_{t-1}) \\ \tilde{\mathbf{c}}_t &= \text{tanh}(\mathbf{U}_{cx}\mathbf{S}_{cx}\mathbf{V}_{cx}\mathbf{x}_t + \mathbf{W}_{ch}\mathbf{h}_{t-1})\end{aligned}$$

Particularly, the matrix set $\{\mathbf{S}\}$ is specific to each style in the task and is responsible to capture the underlying style features in the data.

In (Kiddon et al., 2016), the GRU unit is modified to accommodate extra inputs - goal \mathbf{g} and agenda items E_t^{new} in the recipe generation task. The operation of the new component $\tilde{\mathbf{h}}_t$ is given by:

$$\tilde{\mathbf{h}}_t = \text{tanh}(\mathbf{W}_h\mathbf{x}_t + \mathbf{r}_t \odot \mathbf{U}_h\mathbf{h}_{t-1} + \mathbf{s}_t \odot \mathbf{Y}\mathbf{g} + \mathbf{q}_t \odot (\mathbf{1}_L^T \mathbf{Z}\mathbf{E}_t^{new})^T)$$

where \mathbf{s}_t is a goal select gate and \mathbf{q}_t is a item select gate. With this modification, the generation process is controlled for the items to be generation in the recipe and the goal.

Wen et al. (2015) adapt the LSTM to control the dialogue act information in the generation process. The operation to compute the cell value \mathbf{c}_t is given by:

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \tilde{\mathbf{c}}_t + \text{tanh}(\mathbf{W}_d\mathbf{d}_t)$$

The dialogue act representation \mathbf{d}_t is build using another LSTM cell.

2.5 Output

Here, I discuss the various techniques used to modulate the sequential output \mathbf{o}_t at each time step of the generator. This is marked as position 4 in Figure 2.1.

Attention: Attention is the most popular way of guiding the generation process. It is typically used to guide the generation process to focus on the source sequence (Bahdanau et al., 2015). The attention calculating module takes as input the current hidden state \mathbf{h}_t of the generator at each time step t . The aim of this module is to determine a context vector \mathbf{c}_t that captures relevant source-side information to help predict the current target word \mathbf{y}_t . In case of *global attention*, all the hidden states of the encoder are considered to calculate the context vector \mathbf{c}_t (Luong et al., 2015a). This faces the downside of expensive calculation especially for longer source sequences like documents. To overcome this challenge, *local attention* only chooses to focus only on a small subset of the source positions per target word. In this case, \mathbf{c}_t is calculated over a window of size D of the source hidden states.

Vaswani et al. (2017) view attention as a mapping a query and a set of key-value pairs to an output, where the query, keys, values, and output are all vectors. The output is computed as a weighted sum of the values, where the weight assigned to each value is computed by a compatibility function of the query with the corresponding key. This work proposes the simultaneous use of *scaled dot-product* attention which helps in parallelizing computation and a *multi-headed* attention which allows the model to jointly attend to information from different representation subspaces at different positions.

Sudhakar et al. (2019) use self-attention to control for style by simply adding the a special target style token in the source sequence. Dinan et al. (2018) also use transformers to attend over information from external document for guided dialogue response generation in their Two Stage model. (Zhang et al., 2018) uses the encoded representation of personas to compute the attention weights \mathbf{a}_t at a given time step of the decoder. The attention is reweighted according to the persona of the response to be generated in dialogue. So far, work has not been done to modulate the attention weights to control for attributes like style, topic, content etc.

External Feedback: The output latent space of the generator can be controlled by external feedback. Similar to changing the external input \mathbf{h}_0 , the output latent space can also be changed using adversarial loss. In (Logeswaran et al., 2018a), an adversarial loss is used which encourages the generation realistic and attribute compatible sentences. The adversarial loss tries to match the distribution of sentence and attribute vector pairs (\mathbf{x}, \mathbf{s}) where the sentence can either be a real or generated sentence. Gong et al. (2019) also control the output latent space by providing different types of rewards like style reward, semantic reward and fluency reward in the reinforcement learning setup.

Arithmetic or Linear Transform: Hoang et al. (2016) demonstrate three simple ways of changing the output \mathbf{o}_t of an RNN to control for meta information like topic, keywords etc. They show that you can add the control vector \mathbf{s} to \mathbf{o}_t . Hence the modified output $\tilde{\mathbf{o}}_t$ is $\tilde{\mathbf{o}}_t = \mathbf{o}_t + \mathbf{s}$. Similarly, you can create $\tilde{\mathbf{o}}_t$ by concatenating \mathbf{s} to \mathbf{o}_t ($\tilde{\mathbf{o}}_t = [\mathbf{o}_t; \mathbf{s}]$). We can also build $\tilde{\mathbf{o}}_t$ using a perceptron layer dependent on \mathbf{s} and \mathbf{o}_t . In this case, $\tilde{\mathbf{o}}_t$ is given by $\tilde{\mathbf{o}}_t =$

$\tanh(\mathbf{W}_o \mathbf{o}_t + \mathbf{W}_s \mathbf{s} + \mathbf{b}_o)$. In each of the three cases, the modified output $\tilde{\mathbf{o}}_t$ is then projected to the vocabulary space to predict the token \mathbf{y}_t .

2.6 Training Objective

In this section I describe various methods used to control the generation using objective functions. The output \mathbf{o}_t at each time step t of the generation process is projected to the vocabulary space using a linear transform ($\tilde{\mathbf{o}}_t = \mathbf{W}_o \mathbf{o}_t + \mathbf{b}$). A token $\hat{\mathbf{x}}_t$ is predicted from the vocabulary by passing \mathbf{o}_t through a softmax function and taking the max value. The predicted token $\hat{\mathbf{x}}_t$ is compared with the reference token \mathbf{y}_t using the loss function. This loss function can be tweaked to ensure that the generated text carries the desired control attributes.

General Loss Objectives: Here, I describe the loss objectives commonly used in natural language generation tasks. These loss objectives do not try to control for any attribute. Instead they try to ensure fluent, grammatical and diverse generations.

Cross Entropy Loss is the basic loss used to compare the generated tokens with the reference tokens and is used in all text generation process. At each time step t , the generation has to predict a token from the vocabulary. Hence, it could be seen as a classification problem with number of classes being equal to vocabulary size. The categorical cross entropy loss is given by:

$$-\sum_{c=1}^M \mathbf{y}_{t,c} \log(p_{t,c})$$

where $p_{t,c}$ is the probability of the token c at time step t . Note that $p_t = \text{softmax}(\tilde{\mathbf{o}}_t)$ is the probability distribution over the vocabulary.

Unlikelihood objective maintains a set of negative candidates which is based on repeating tokens or n-grams and frequent tokens. This set is updated at each time step as tokens are generated. This works at both token and sequence level and the objective tries to minimize the repetitions in generations. This is used at train time in augmentation with the maximum likelihood objective and can be used for any task.

Diversity-Promoting objective is used to generate a varied set of sentences given similar inputs. Particularly, Li et al. (2016a) use Maximum Mutual Information (MMI) as an objective function for the dialogue response generation task. Most generation systems use maximum likelihood objective but this objective additionally tries to reduce the proportion of generic responses. It is given by:

$$\hat{\mathbf{T}} = \text{argmax}_T \{ \log p(\mathbf{T}|\mathbf{S}) - \lambda \log p(\mathbf{T}) \}$$

where $\hat{\mathbf{T}}$ is the generated target sequence, \mathbf{T} is the reference target sequence and \mathbf{S} is the source sequence. The second term controls the generation of the high frequency or the generic target sequences. Note that this objective is only used during the inference and the generators are

trained using cross entropy loss. Zhang et al. (2018), also use a diversity encouraging objective for dialogue response generation. They train a discriminator to calculate similarity between the source \mathbf{S} and target \mathbf{T} ($D_\psi(\mathbf{T}, \mathbf{S})$), as well as between the source \mathbf{S} and the generated target $\hat{\mathbf{T}}$ ($D_\psi(\hat{\mathbf{T}}, \mathbf{S})$). They finally try to minimize the difference between $D_\psi(\mathbf{T}, \mathbf{S})$ and $D_\psi(\hat{\mathbf{T}}, \mathbf{S})$.

Apart from these, many other objectives rely on post-hoc decoding strategies such as stochastic decoding which include Top k -sampling (Fan et al., 2018), nucleus sampling (Holtzman et al., 2020), or beam search variants (Paulus et al., 2018; Kulikov et al., 2019; Vijayakumar et al., 2018; Holtzman et al., 2018).

KL Divergence: The Kullback-Leibler (KL) Divergence score, quantifies how much one probability distribution differs from another probability distribution. The KL divergence between two distributions \mathcal{Q} and \mathcal{P} is often stated using the following notation:

$$\text{KL}(\mathcal{P} \parallel \mathcal{Q})$$

where the operator “ \parallel ” indicates *divergence* or \mathcal{P} ’s divergence from \mathcal{Q} . Note that KL Divergence is not symmetric i.e $\text{KL}(\mathcal{P} \parallel \mathcal{Q}) \neq \text{KL}(\mathcal{Q} \parallel \mathcal{P})$. KL divergence can be used to minimize the information loss while approximating a distribution. In text generation, the KL Divergence is combined with the evidence lower bound (ELBO) to approximately maximize the marginal likelihood of data $p(\mathbf{x})$ which helps in better generations. This objective is used in variational autoencoders and its variants in combination with sampling techniques described in §2.2. This objective fits in the controllable text generation paradigm because it allows you to approximate the posterior distribution of the control variables in the latent \mathbf{z} -space.

Classifier Loss: This loss is specifically used to ensure that the generated tokens $\hat{\mathbf{x}}$ comply with the control attributes \mathbf{s} . Note the difference between this loss and the external feedback loss used for the *external input* module and the *output* module is that this loss operates at the token level and the external feedback loss works on the latent hidden representations.

In case of style transfer task, this loss is used to guide the generation process to output the target style tokens. Some works (Prabhumoye et al., 2018; Sudhakar et al., 2019; Hu et al., 2017) use this loss to discriminate between all the styles in their task (one verses all fashion). This type of design will suffer from low accuracy scores when the number of styles increases. To counter this problem, this loss can be setup to calculate if the generated sentence $\hat{\mathbf{x}}$ belongs to style \mathbf{s}_1 or not and similarly to calculate another separate loss term for each style (Chandu et al., 2019b). This type of loss design encounters increasing number of loss terms depending on the number of styles. The third way to motivate this loss term is to discriminating between a sentence \mathbf{x} from data which belongs to style \mathbf{s}_1 and a generated sentence $\hat{\mathbf{x}}$ which belongs to the same style \mathbf{s}_1 (Yang et al., 2018). Again, you would need as many loss terms as the number of styles in this case. All of these works use cross entropy loss function to measure their losses.

Hu et al. (2019) use a classifier based loss in the visual storytelling task. The classifier is a pre-trained language model (Devlin et al., 2019) used to measure the coherence between generated sentences of the story. Particularly, the classifier takes as input two sentences at a time \hat{x}_1 and \hat{x}_2 and outputs a binary label which indicates if \hat{x}_2 follows \hat{x}_1 . In this case, the control variable is coherence in stories which is used to guide the generator to produce consistent sentences.

Task Specific Loss: Depending on the end task and the attribute to be controlled, you can design different loss objectives to ensure that generations abide by the target attributes.

Strategy Loss: Zhou et al. (2020) use a dialogue strategy based objective to generate responses for negotiation tasks. This task has ground truth strategies that lead to better negotiations. This loss captures the probability of a particular strategy occurring for the next utterance given the dialogue history. It guides the generator to align the responses with particular strategies.

Coverage Loss: Generating repeated words or phrases is a common problem for text generation systems, and this becomes especially pronounced for multi-sentence text generation task such as abstractive document summarization. See et al. (2017) introduce a *coverage loss* which penalizes repeatedly attending to the same locations of the source document.

Structure loss: Li et al. (2018b) introduce two new loss objectives *structural compression* and *structural coverage* based on sentence-level attention. These objectives are specially designed for the task of abstractive document summarization. *structural compression* is used to generate a sentence by compressing several specific source sentences and *structural coverage* is used to cover more salient information of the original document. These objectives leverage document structure in document summarization, and explore the effectiveness of capturing structural properties of document summarization by regularization of the generative model to generate more informative and concise summaries.

2.7 Proposed Work

I propose to select three controllable text generation tasks and provide empirical insight into which of the described techniques work better or worse for different tasks. In particular I plan to work on the style transfer tasks of sentiment modification (Shen et al., 2017) and political slant transfer (Prabhumoye et al., 2018), content grounded dialogue generation (Zhou et al., 2018), content grounded Wikipedia edit generation task (Prabhumoye et al., 2019) and the persona based dialogue response generation task (Zhang et al., 2018). I will select one or two techniques from each of the five modules to gain understanding of the contribution of each of the modules in controlled text generation. It is also possible that a combination of techniques is suitable for some tasks. With these set of experiments, I wish to provide new directions to explore for controllable text generation.

Chapter 3

Style

Style is used to communicate in an economical, strategic and believable way. For example simply stating ‘I’m angry’ is less convincing than shouting ‘Damn!’ (Eckert, 2019). Yet, defining a style is a non-trivial task. A descriptive approach to defining a style is of very little use in a theory of language production, since it never makes clear why and how each style is formed out of words; nor does it indicate any systematicity behind the classification of styles. Additionally, classifying all the possible styles of text is an impossible task: One can imagine text characteristics that fit almost any adjective! (Hovy, 1987). Eckert (2019) investigates the social indexicalities that can contribute to the emergence of a particular style. An important point made here is that style develops as a contrast to the existing indexicalities. An example of this in the non-linguistic domain is that if everyone wears black all the time then there is no existence of a style. Style will only exist when at least one person decides to wear a different color or form of clothing.

Kang and Hovy (2019) adopts Hovy (1987)’s functional approach to define style by its pragmatics aspects and rhetorical goals, or personal and group characteristics of participants. This work categorizes style along the two axes of social participation and content coupledness. It further identifies demographic attributes such as gender, age, education etc as personal styles, and formality, politeness as interpersonal. Sentiment, humor, romance on the other hand have been identified as heavily content coupled styles. For the purpose of my experiments, I assume a group of examples of text that belong to same label as one style. For example a set of sentences from a comedy show intended to incite humor are considered to belong to humorous style. Similarly, sentences written by George Orwell would be considered to be written in the Orwellian author style. I acknowledge that a piece of text could be a mix of multiple styles. For example, Orwellian work is both written in the author style as well as satirical.

Style transfer is the task of rephrasing the text to contain specific stylistic properties without changing the intent or affect within the context. There is a constant debate in the community on what is considered as preserving the semantic content in the case of style transfer. In my opinion the evaluation of meaning preservation should be done using the downstream

application for which the style transfer is to be used. For example, when writing a customer review for a product or restaurant, the over all sentiment of the review should remain the same while changing the demographic attributes or politeness of the review. If the review complains or appreciated the food/service then the generated sentence should maintain the same. In a lenient evaluation it might be ok to change the name of the food item in the review. But such a mistake will not be appreciated if the downstream task is ordering food from a restaurant or products from amazon. When generating sentences for orders in different style, the quantity and the food item/product name should remain the same in the output.

The most popular application of style transfer is to generate diverse responses for dialogue systems. You can control politeness, authority, persona etc of the dialogue responses. Style transfer can also be used to control the politeness of email request. I have automatically labelled a huge dataset of 1.39 million sentences from Enron email corpus (Yeh and Harnly, 2006) for politeness (Madaan et al., 2020) and show effective transfer of non-polite email requests to polite. Story generation is another interesting application of style transfer. You can use style transfer to generate the story with different emotional endings (Peng et al., 2018) or as I show in (Chandu et al., 2019b), you can generate stories in different persona types. The use of style transfer in machine translation task has recently caught attention (Niu et al., 2017; Niu and Carpuat, 2019).

Challenges: The main challenge in style transfer task is the lack of parallel data. Very few datasets exist with sentences which are aligned in all styles (Rao and Tetreault, 2018). This also makes it hard to evaluate the generated sentences for the style transfer task. The other challenges include not having good definitions of style, the datasets for style transfer may contain confounding variables on which the sentences might depend on, there are no good evaluation metrics to evaluate both style transfer accuracy and meaning preservation in style, for style transfer techniques it is hard to disentangle the meaning of a sentence from its style.

3.1 Methodology

I introduce a novel approach to transferring style of a sentence while better preserving its meaning. I hypothesize—relying on the study of Rabinovich et al. (2016) who showed that author characteristics are significantly obfuscated by both manual and automatic machine translation—that grounding in back-translation is a plausible approach to rephrase a sentence while reducing its stylistic properties. I thus first use back-translation to rephrase the sentence and reduce the effect of the original style; then, I generate from the latent representation, using separate style-specific generators controlling for style.

Given two datasets $\mathbf{X}_1 = \{\mathbf{x}_1^{(1)}, \dots, \mathbf{x}_1^{(n)}\}$ and $\mathbf{X}_2 = \{\mathbf{x}_2^{(1)}, \dots, \mathbf{x}_2^{(n)}\}$ which represent two different styles s_1 and s_2 , respectively, my task is to generate sentences of the desired style while preserving the meaning of the input sentence. Specifically, I generate samples of dataset

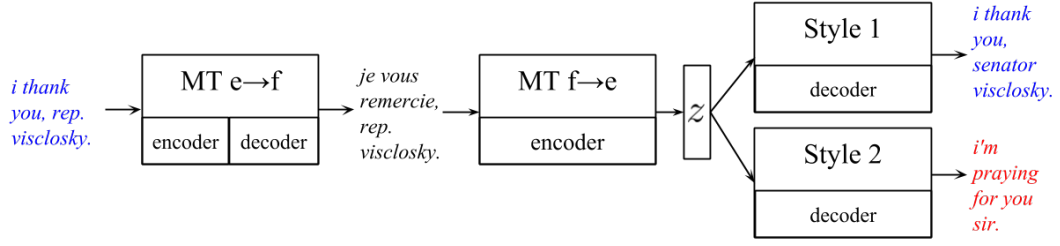


FIGURE 3.1: Style transfer pipeline: to rephrase a sentence and reduce its stylistic characteristics, the sentence is back-translated. Then, separate style-specific generators are used for style transfer.

\mathbf{X}_1 such that they belong to style s_2 and samples of \mathbf{X}_2 such that they belong to style s_1 . I denote the output of dataset \mathbf{X}_1 transferred to style s_2 as $\hat{\mathbf{X}}_1 = \{\hat{x}_2^{(1)}, \dots, \hat{x}_2^{(n)}\}$ and the output of dataset \mathbf{X}_2 transferred to style s_1 as $\hat{\mathbf{X}}_2 = \{\hat{x}_1^{(1)}, \dots, \hat{x}_1^{(n)}\}$.

Hu et al. (2017) and Shen et al. (2017) introduced state-of-the-art style transfer models that use variational auto-encoders (Kingma and Welling, 2014, VAEs) and cross-aligned auto-encoders, respectively, to model a latent content variable z . The latent content variable z is a code which is not observed. The generative model conditions on this code during the generation process. My aim is to design a latent code z which (1) represents the meaning of the input sentence grounded in back-translation and (2) weakens the style attributes of author's traits. To model the former, I use neural machine translation. Prior work has shown that the process of translating a sentence from a source language to a target language retains the meaning of the sentence but does not preserve the stylistic features related to the author's traits (Rabinovich et al., 2016). I hypothesize that a latent code z obtained through back-translation will normalize the sentence and devoid it from style attributes specific to author's traits.

Figure 3.1 shows the overview of the proposed method. In my framework, I first train a machine translation model from source language e to a target language f . I also train a back-translation model from f to e . Let us assume the styles s_1 and s_2 correspond to DEMOCRATIC and REPUBLICAN style, respectively. In Figure 3.1, the input sentence *i thank you, rep. visclosky.* is labeled as DEMOCRATIC. I translate the sentence using the $e \rightarrow f$ machine translation model and generate the parallel sentence in the target language f : *je vous remercie, rep. visclosky.* Using the fixed encoder of the $f \rightarrow e$ machine translation model, I encode this sentence in language f . The hidden representation created by this encoder of the back-translation model is used as z . I condition my generative models on this z . I then train two separate decoders for each style s_1 and s_2 to generate samples in these respective styles in source language e . Hence the sentence could be translated to the REPUBLICAN style using the decoder for s_2 . For example, the sentence *i'm praying for you sir.* is the REPUBLICAN version of the input sentence and *i thank you, senator visclosky.* is the more DEMOCRATIC version of it.

Note that in this setting, the machine translation and the encoder of the back-translation model remain fixed. They are not dependent on the data I use across different tasks. This facilitates re-usability and spares the need of learning separate models to generate z for a new style data.

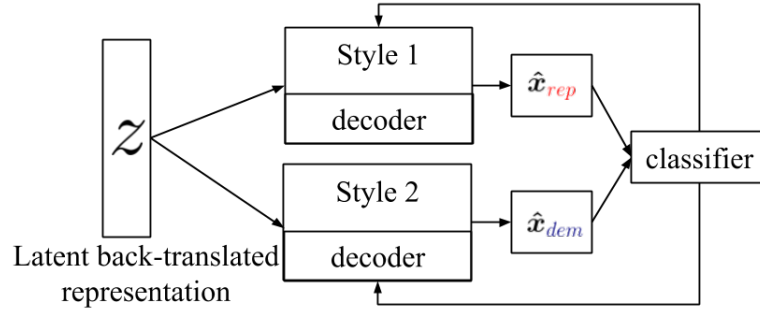


FIGURE 3.2: The latent representation from back-translation and the style classifier feedback are used to guide the style-specific generators.

3.1.1 Meaning-Grounded Representation

In this section I describe how I learn the latent content variable z using back-translation. The $e \rightarrow f$ machine translation and $f \rightarrow e$ back-translation models are trained using a sequence-to-sequence framework (Sutskever et al., 2014; Bahdanau et al., 2015) with style-agnostic corpus. The style-specific sentence *i thank you, rep. visclosky.* in source language e is translated to the target language f to get *je vous remercie, rep. visclosky.* The individual tokens of this sentence are then encoded using the encoder of the $f \rightarrow e$ back-translation model. The learned hidden representation is z .

Formally, let θ_E represent the parameters of the encoder of $f \rightarrow e$ translation system. Then z is given by:

$$z = \text{Encoder}(x_f; \theta_E) \quad (3.1)$$

where, x_f is the sentence x in language f . Specifically, x_f is the output of $e \rightarrow f$ translation system when x_e is given as input. Since z is derived from a non-style specific process, this Encoder is not style specific.

3.1.2 Style-Specific Generation

Figure 3.2 shows the architecture of the generative model for generating different styles. Using the encoder embedding z , I train multiple decoders for each style. The sentence generated by a decoder is passed through the classifier. The loss of the classifier for the generated sentence is used as feedback to guide the decoder for the generation process. The target attribute of the classifier is determined by the decoder from which the output is generated. For example, in the case of DEMOCRATIC decoder, the target attribute is DEMOCRATIC and for the REPUBLICAN decoder the target is REPUBLICAN.

Style Classifiers

I train a convolutional neural network (CNN) classifier to accurately predict the given style. I also use it to evaluate the error in the generated samples for the desired style. I train the classifier in a supervised manner. The classifier accepts either discrete or continuous tokens as inputs. This is done such that the generator output can be used as input to the classifier. I need labeled examples to train the classifier such that each instance in the dataset \mathbf{X} should have a label in the set $\mathbf{s} = \{s_1, s_2\}$. Let θ_C denote the parameters of the classifier. The objective to train the classifier is given by:

$$\mathcal{L}_{class}(\theta_C) = \mathbb{E}_{\mathbf{X}} [\log q_C(\mathbf{s}|\mathbf{x})]. \quad (3.2)$$

To improve the accuracy of the classifier, I augment classifier's inputs with style-specific lexicons. I concatenate binary style indicators to each input word embedding in the classifier. The indicators are set to 1 if the input word is present in a style-specific lexicon; otherwise they are set to 0. Style lexicons are extracted using the log-odds ratio informative Dirichlet prior (Monroe et al., 2008), a method that identifies words that are statistically overrepresented in each of the categories.

Generator Learning

I use a bidirectional LSTM to build the decoders which generate the sequence of tokens $\hat{\mathbf{x}} = \{x_1, \dots, x_T\}$. The sequence $\hat{\mathbf{x}}$ is conditioned on the latent code \mathbf{z} (in my case, on the machine translation model). In this work I use a corpus translated to French by the machine translation system as the input to the encoder of the back-translation model. The same encoder is used to encode sentences of both styles. The representation created by this encoder is given by Eq. 3.1. Samples are generated as follows:

$$\hat{\mathbf{x}} \sim \mathbf{z} = p(\hat{\mathbf{x}}|\mathbf{z}) \quad (3.3)$$

$$= \prod_t p(\hat{x}_t | \hat{\mathbf{x}}^{<t}, \mathbf{z}) \quad (3.4)$$

where, $\hat{\mathbf{x}}^{<t}$ are the tokens generated before \hat{x}_t .

Tokens are discrete and non-differentiable. This makes it difficult to use a classifier, as the generation process samples discrete tokens from the multinomial distribution parametrized using softmax function at each time step t . This non-differentiability, in turn, breaks down gradient propagation from the discriminators to the generator. Instead, following Hu et al. (2017) I use a continuous approximation based on softmax, along with the temperature parameter which anneals the softmax to the discrete case as training proceeds. To create a continuous representation of the output of the generative model which will be given as an input to the classifier, I use:

$$\hat{x}_t \sim \text{softmax}(\mathbf{o}_t/\tau),$$

where, \mathbf{o}_t is the output of the generator and τ is the temperature which decreases as the training proceeds. Let θ_G denote the parameters of the generators. Then the reconstruction loss is calculated using the cross entropy function, given by:

$$\mathcal{L}_{recon}(\theta_G; \mathbf{x}) = \mathbb{E}_{q_E(\mathbf{z}|\mathbf{x})}[\log p_{gen}(\mathbf{x}|\mathbf{z})] \quad (3.5)$$

Here, the back-translation encoder E creates the latent code \mathbf{z} by:

$$\mathbf{z} = E(\mathbf{x}) = q_E(\mathbf{z}|\mathbf{x}) \quad (3.6)$$

The generative loss \mathcal{L}_{gen} is then given by:

$$\min_{\theta_{gen}} \mathcal{L}_{gen} = \mathcal{L}_{recon} + \lambda_c \mathcal{L}_{class} \quad (3.7)$$

where \mathcal{L}_{recon} is given by Eq. 3.5, \mathcal{L}_{class} is given by Eq. 3.2 and λ_c is a balancing parameter.

I also use global attention of (Luong et al., 2015b) to aid my generators. At each time step t of the generation process, I infer a variable length alignment vector \mathbf{a}_t :

$$\mathbf{a}_t = \frac{\exp(\text{score}(\mathbf{h}_t, \bar{\mathbf{h}}_s))}{\sum_{s'} \exp(\text{score}(\mathbf{h}_t, \bar{\mathbf{h}}_{s'}))} \quad (3.8)$$

$$\text{score}(\mathbf{h}_t, \bar{\mathbf{h}}_s) = \text{dot}(\mathbf{h}_t^T, \bar{\mathbf{h}}_s), \quad (3.9)$$

where \mathbf{h}_t is the current target state and $\bar{\mathbf{h}}_s$ are all source states. While generating sentences, I use the attention vector to replace unknown characters (UNK) using the copy mechanism in (See et al., 2017).

3.2 Style Transfer Tasks and Datasets

Much work in computational social science has shown that people’s personal and demographic characteristics—either publicly observable (e.g., age, gender) or private (e.g., religion, political affiliation)—are revealed in their linguistic choices (Nguyen et al., 2016). There are practical scenarios, however, when these attributes need to be modulated or obfuscated. For example, some users may wish to preserve their anonymity online, for personal security concerns (Jardine, 2016), or to reduce stereotype threat (Spencer et al., 1999). Modulating authors’ attributes while preserving meaning of sentences can also help generate demographically-balanced training data for a variety of downstream applications.

Moreover, prior work has shown that the quality of language identification and POS tagging degrades significantly on African American Vernacular English (Blodgett et al., 2016; Jørgensen

et al., 2015); YouTube’s automatic captions have higher error rates for women and speakers from Scotland (Rudinger et al., 2017). Synthesizing balanced training data—using style transfer techniques—is a plausible way to alleviate bias present in existing NLP technologies.

I thus focus on two tasks that have practical and social-good applications, and also accurate style classifiers. To position my method with respect to prior work, I employ a third task of sentiment transfer, which was used in two state-of-the-art approaches to style transfer (Hu et al., 2017; Shen et al., 2017). I describe the three tasks and associated dataset statistics below. The methodology that I advocate is general and can be applied to other styles, for transferring various social categories, types of bias, and in multi-class settings.

Gender: In sociolinguistics, gender is known to be one of the most important social categories driving language choice (Eckert and McConnell-Ginet, 2003; Lakoff and Bucholtz, 2004; Coates, 2015; Tannen, 1991, 1993; Kendall et al., 1997; Eckert and McConnell-Ginet, 2003; Lakoff and Bucholtz, 2004; Coates, 2015). Numerous computational studies have also explored how gender is manifested in language of social media users (Rao et al., 2010; Burger et al., 2011; Peersman et al., 2011; Bergsma and Van Durme, 2013; Flekova and Gurevych, 2013; Bamman et al., 2014; Volkova et al., 2015; Carpenter et al., 2016, *inter alia*). Gender-induced style differences include, for example, that women are more likely to use pronouns, emotion words (like *sad*, *love*, and *glad*), interjections (*ah*, *hmmmm*, *ugh*), emoticons, and abbreviations associated with online discourse (*lol*, *omg*), while men tend to use higher frequency standard dictionary words, proper names (e.g., the names of sports teams), numbers, technology words, and links (Bamman et al., 2014). Reddy and Knight (2016) proposed a heuristic-based method to obfuscate gender of a writer. This method uses statistical association measures to identify gender-salient words and substitute them with synonyms typically of the opposite gender. This simple approach produces highly fluent, meaning-preserving sentences, but does not allow for more general rephrasing of sentence beyond single-word substitutions. In my work, I adopt this task of transferring the author’s gender and adapt it to my experimental settings.

I used Reddy and Knight’s (2016) dataset of reviews from Yelp annotated for two genders corresponding to markers of sex.¹ I split the reviews to sentences, preserving the original gender labels. To keep only sentences that are strongly indicative of a gender, I then filtered out gender-neutral sentences (e.g., *thank you*) and sentences whose likelihood to be written by authors of one gender is lower than 0.7.²

¹I note that gender may be considered along a spectrum (Eckert and McConnell-Ginet, 2003), but use gender as a binary variable due to the absence of corpora with continuous-valued gender annotations.

²I did not experiment with other threshold values.

Style	<i>class</i>	<i>train</i>	<i>dev</i>	<i>test</i>
gender	2.57M	2.67M	4.5K	535K
political	80K	540K	4K	56K
sentiment	2M	444K	63.5K	127K

TABLE 3.1: Sentence count in style-specific corpora.

Political slant: My second dataset is comprised of top-level comments on Facebook posts from all 412 current members of the United States Senate and House who have public Facebook pages (Voigt et al., 2018).³ Only top-level comments that directly respond to the post are included. Every comment to a Congressperson is labeled with the Congressperson’s party affiliation: democratic or republican. Topic and sentiment in these comments reveal commenter’s political slant. For example, *defund them all, especially when it comes to the illegal immigrants* . and *thank u james, praying for all the work u do* . are republican, whereas *on behalf of the hard-working nh public school teachers- thank you !* and *we need more strong voices like yours fighting for gun control* . represent examples of democratic sentences. My task is to preserve intent of the commenter (e.g., to thank their representative), but to modify their observable political affiliation, as in the example in Figure 3.1. I preprocessed and filtered the comments similarly to the gender-annotated corpus above.

Sentiment: To compare my work with the state-of-the-art approaches of style transfer for non-parallel corpus I perform sentiment transfer, replicating the models and experimental setups of Hu et al. (2017) and Shen et al. (2017). Given a positive Yelp review, a style transfer model will generate a similar review but with an opposite sentiment. I used Shen et al.’s (2017) corpus of reviews from Yelp. They have followed the standard practice of labeling the reviews with rating of higher than three as positive and less than three as negative. They have also split the reviews to sentences and assumed that the sentence has the same sentiment as the review.

Dataset statistics: I summarize below corpora statistics for the three tasks: transferring gender, political slant, and sentiment. The dataset for sentiment modification task was used as described in (Shen et al., 2017). I split Yelp and Facebook corpora into four disjoint parts each: (1) a training corpus for training a style classifier (*class*); (2) a training corpus (*train*) used for training the style-specific generative model described in §3.1.2; (3) development and (4) test sets. I have removed from training corpora *class* and *train* all sentences that overlap with development and test corpora. Corpora sizes are shown in Table 4.1.

Table 3.2 shows the approximate vocabulary sizes used for each dataset. The vocabulary is the same for both the styles in each experiment.

³The posts and comments are all public; however, to protect the identity of Facebook users in this dataset Voigt et al. (2018) have removed all identifying user information as well as Facebook-internal information such as User IDs and Post IDs, replacing these with randomized ID numbers.

Style	gender	political	sentiment
Vocabulary	20K	20K	10K

TABLE 3.2: Vocabulary sizes of the datasets.

Table 3.3 summarizes sentence statistics. All the sentences have maximum length of 50 tokens.

Style	Avg. Length	%data
male	18.08	50.00
female	18.21	50.00
republican	16.18	50.00
democratic	16.01	50.00
negative	9.66	39.81
positive	8.45	60.19

TABLE 3.3: Average sentence length and class distribution of style corpora.

3.3 Evaluation

Evaluating style transfer techniques is hard. I have to not only evaluate the generations for the success of style transfer but also if the generated sentence maintains the same meaning as the input sentence. Additionally, I must also evaluate if the generations are syntactically and grammatically sound. Both automatic evaluation and human judgments are used to evaluate style transfer systems along the three dimensions of: (1) Style transfer accuracy, measuring the proportion of my models' outputs that generate sentences of the desired style. (2) Preservation of meaning. (3) Fluency, measuring the readability and the naturalness of the generated sentences.

3.3.1 Style Transfer Accuracy

Automatic Evaluation: I measure the accuracy of style transfer for the generated sentences using a pre-trained style classifier. The classifier is trained on data that is not used for training the style transfer generative models shown in Table 3.1. I transfer the style of test sentences and then test the classification accuracy of the generated sentences for the opposite label. For example, if I want to transfer the style of male Yelp reviews to female, then I use the fixed common encoder of the back-translation model to encode the test male sentences and then I use the female generative model to generate the female-styled reviews. I then test these generated sentences for the *female* label using the gender classifier.

The classifier has an accuracy of 82% for the gender-annotated corpus, 92% accuracy for the political slant dataset and 93.23% accuracy for the sentiment dataset.

Human Evaluation: Li et al. (2018a) introduce human evaluation for assessing the strength of transfer. Human judges are asked to annotate the generated sentence on a scale of 1 to 5 for similarity to target attribute. Although this is a good practice, demographic attributes such as gender, age and personal choices such political slant etc must not be evaluated by human judgements as there is a danger of bias and stereotypes introduced by people during the evaluation process. This work has performed an analysis of the correlation of the human judgements with the automatic evaluation and argues that it depends on the dataset and the task. Hence, the correlation cannot be taken for granted.

3.3.2 Preservation of Meaning

Automatic Evaluation: To measure preservation of meaning in style transfer, some works have borrowed metrics from other generation or translation tasks such as BLEU (Papineni et al., 2002), ROUGE (Lin and Hovy, 2002) or METEOR (Denkowski and Lavie, 2011). Li et al. (2018a) have released a test set of human references primarily for the sentiment modification task. In this case, you can calculate BLEU between the human references and the generated sentences. While popular, the metrics of Transfer Accuracy and BLEU have significant shortcomings making them susceptible to simple adversaries. BLEU relies heavily on n-gram overlap and classifiers can be fooled by certain polarizing keywords. I test this hypothesis on the sentiment transfer task by a *Naive Baseline*. This baseline adds “*but overall it sucked*” at the end of the sentence to transfer it to negative sentiment. Similarly, it appends “*but overall it was perfect*” for transfer into a positive sentiment. This baseline achieves an average accuracy score of 91.3% and a BLEU score of 61.44 on the Yelp dataset. Despite the stellar performance, it does not reflect a high rate of success on the task. In summary, evaluation via automatic metrics might not truly correlate with task success.

Human Evaluation: Meaning preservation in style transfer is not trivial to define as literal meaning is likely to change when style transfer occurs. For example “My girlfriend loved the desserts” vs “My partner liked the desserts”. Thus I must relax the condition of literal meaning to *intent* or *affect* of the utterance within the context of the discourse. Thus if the intent is to criticize a restaurant’s service in a review, changing “salad” to “chicken” could still have the same effect but if the intent is to order food that substitution would not be acceptable. Ideally I wish to evaluate transfer within some downstream task and ensure that the task has the same outcome even after style transfer. This is a hard evaluation and hence I resort to a simpler evaluation of the “meaning” of the sentence.

I set up a manual pairwise comparison following Bennett (2005). The test presents the original sentence and then, in random order, its corresponding sentences produced by the baseline and my models. For the gender style transfer I asked “Which transferred sentence maintains the same sentiment of the source sentence in the same semantic context (i.e. you can ignore if food items are changed)”. For the task of changing the political slant, I asked “Which transferred

sentence maintains the same semantic intent of the source sentence while changing the political position”. For the task of sentiment transfer I have followed the annotation instruction in (Shen et al., 2017) and asked “Which transferred sentence is semantically equivalent to the source sentence with an opposite sentiment”

I then count the preferences of the eleven participants, measuring the relative acceptance of the generated sentences.⁴ A third option “=” was given to participants to mark no preference for either of the generated sentence. The “no preference” option includes choices both are equally bad and both are equally good. I conducted three tests one for each type of experiment - gender, political slant and sentiment. I also divided my annotation set into short ($\#tokens \leq 15$) and long ($15 < \#tokens \leq 30$) sentences for the gender and the political slant experiment. In each set I had 20 random samples for each type of style transfer. In total I had 100 sentences to be annotated. Note that I did not ask about appropriateness of the style transfer in this test, or fluency of outputs, only about meaning preservation.

3.3.3 Fluency

Automatic Evaluation: Yang et al. (2018); He et al. (2020); Lample et al. (2018) use perplexity to measure the fluency of the generated sentences. In most cases perplexity is not correlated with human judgements of fluency.

Human Evaluation: Finally, I evaluate the fluency of the generated sentences. Fluency was rated from 1 (unreadable) to 4 (perfect) as is described in (Shen et al., 2017). I randomly selected 60 sentences each generated by the baseline and the BST model.

3.4 Results

Translation quality: The BLEU scores achieved for English–French MT system is 32.52 and for French–English MT system is 31.11; these are strong translation systems. I deliberately chose a European language close to English for which massive amounts of parallel data are available and translation quality is high, to concentrate on the style generation, rather than improving a translation system.⁵

In Table 3.4, I detail the accuracy of each classifier on generated style-transferred sentences.⁶ I denote the (Shen et al., 2017) Cross-aligned Auto-Encoder model as CAE and my model as Back-translation for Style Transfer (BST).

⁴None of the human judges are authors of this paper

⁵Alternatively, I could use a pivot language that is typologically more distant from English, e.g., Chinese. In this case I hypothesize that stylistic traits would be even less preserved in translation, but the quality of back-translated sentences would be worse. I have not yet investigated how the accuracy of the translation model, nor the language of translation affects my models.

⁶In each experiment, I report aggregated results across directions of style transfer; same results broke-down to style categories are listed in the Supplementary Material.

Experiment	CAE	BST
Gender	60.40	57.04
Political slant	75.82	88.01
Sentiment	80.43	87.22

TABLE 3.4: Accuracy of the style transfer in generated sentences.

Experiment	CAE	No Pref.	BST
Gender	15.23	41.36	43.41
Political slant	14.55	45.90	39.55
Sentiment	35.91	40.91	23.18

TABLE 3.5: Human preference for meaning preservation in percentages.

On two out of three tasks my model substantially outperforms the baseline, by up to 12% in political slant transfer, and by up to 7% in sentiment modification.

The results of human evaluation are presented in Table 3.5. Although a no-preference option was chosen often—showing that state-of-the-art systems are still not on par with human expectations—the BST models outperform the baselines in the gender and the political slant transfer tasks.

Crucially, the BST models significantly outperform the CAE models when transferring style in longer and harder sentences. Annotators preferred the CAE model only for 12.5% of the long sentences, compared to 47.27% preference for the BST model.

The results shown in Table 3.6 are averaged fluency scores for each model.

Experiment	CAE	BST
Gender	2.42	2.81
Political slant	2.79	2.87
Sentiment	3.09	3.18
Overall	2.70	2.91
Overall Short	3.05	3.11
Overall Long	2.18	2.62

TABLE 3.6: Fluency of the generated sentences.

BST outperforms the baseline overall. It is interesting to note that BST generates significantly more fluent longer sentences than the baseline model. Since the average length of sentences was higher for the gender experiment, BST notably outperformed the baseline in this task, relatively to the sentiment task where the sentences are shorter.

Discussion: The loss function of the generators given in Eq. 3.5 includes two competing terms, one to improve meaning preservation and the other to improve the style transfer accuracy. In the task of sentiment modification, the BST model preserved meaning worse than the baseline, on the expense of being better at style transfer. I note, however, that the sentiment modification task is not particularly well-suited for evaluating style transfer: it is particularly hard (if not impossible) to disentangle the sentiment of a sentence from its propositional content, and to modify sentiment while preserving meaning or intent. On the other hand, the style-transfer accuracy for gender is lower for BST model but the preservation of meaning is much better for the BST model, compared to CAE model and to "No preference" option. This means that the BST model does better job at closely representing the input sentence while taking a mild hit in the style transfer accuracy.⁷

3.5 Related Work

Style transfer with non-parallel text corpus has become an active research area due to the recent advances in text generation tasks. [Hu et al. \(2017\)](#) use variational auto-encoders with a discriminator to generate sentences with controllable attributes. The method learns a disentangled latent representation and generates a sentence from it using a code. This paper mainly focuses on sentiment and tense for style transfer attributes. It evaluates the transfer strength of the generated sentences but does not evaluate the extent of preservation of meaning in the generated sentences. In my work, I show a qualitative evaluation of meaning preservation.

[Shen et al. \(2017\)](#) first present a theoretical analysis of style transfer in text using non-parallel corpus. The paper then proposes a novel cross-alignment auto-encoders with discriminators architecture to generate sentences. It mainly focuses on sentiment and word decipherment for style transfer experiments.

[Fu et al. \(2018\)](#) explore two models for style transfer. The first approach uses multiple decoders for each type of style. In the second approach, style embeddings are used to augment the encoded representations, so that only one decoder needs to be learned to generate outputs in different styles. Style transfer is evaluated on scientific paper titles and newspaper tiles, and sentiment in reviews. This method is different from ours in that I use machine translation to create a strong latent state from which multiple decoders can be trained for each style. I also propose a different human evaluation scheme.

[Li et al. \(2018a\)](#) first extract words or phrases associated with the original style of the sentence, delete them from the original sentence and then replace them with new phrases associated with the target style. They then use a neural model to fluently combine these into a final output. [Junbo et al. \(2017\)](#) learn a representation which is style-agnostic, using adversarial training of the auto-encoder.

⁷Details about hyper-paramters, generated examples and additional experiments are provided in Appendix A.

My work is also closely-related to a problem of paraphrase generation (Madnani and Dorr, 2010; Dong et al., 2017), including methods relying on (phrase-based) back-translation (Ganitkevitch et al., 2011; Ganitkevitch and Callison-Burch, 2014). More recently, Mallinson et al. (2017) and Wieting et al. (2017) showed how neural back-translation can be used to generate paraphrases. An additional related line of research is machine translation with non-parallel data. Lample et al. (2018) and Artetxe et al. (2018) have proposed sophisticated methods for unsupervised machine translation. These methods could in principle be used for style transfer as well.

3.6 Proposed Work

3.6.1 Style Representations

Pre-trained language models like BERT have shown to contain syntax information (Li and Eisner, 2019; Hewitt and Manning, 2019) and relational knowledge (Petroni et al., 2019). I want to extend these ideas to get a stylistic representation from BERT representation.

I first experiment with simple averaging of BERT representations to get the style representation. I have dataset of sentences $\mathbf{x} = \{x_1, \dots, x_n\}$ each x_i is mapped to one or more styles in the set $\mathbf{y} = \{y_1, \dots, y_k\}$. Suppose the set of sentences which belong to style y_i is \mathbf{x}_{y_i} . To build the representation of a style $y_i \in \mathbf{y}$, I follow:

$$\mathbf{S}_{y_i} = \sum_{x_j \in \mathbf{x}_{y_i}} \text{BERT}(x_j) \quad (3.10)$$

I build representations for each style $y_i \in \mathbf{y}$ using Eq. 3.10. To test the quality of y_i , I design a binary classification task to determine if two sentences belong to the same style or not. Note that this is not a style classification task. I test my style representation by training three different classifiers for this task. All three models are based on the pre-trained base uncased BERT model (Devlin et al., 2019) and I don't fine tune the BERT layers. I get the representations of the two sentences (x_1 and x_2 say) using the BERT model (\mathbf{s}_1 and \mathbf{s}_2 say). For the *BERT-model* classifier, I concatenate $[\mathbf{s}_1; \mathbf{s}_2]$ to get a representation h which then passed through two linear layers to get the final prediction. This model provides a baseline accuracy on how much can you learn about styles from BERT. For the *BERT-style* classifier, I subtract the style representations from the sentence representations. I get \mathbf{h} as follows:

$$\mathbf{h} = [\mathbf{s}_1 - \mathbf{S}_{y_i}; \mathbf{s}_2 - \mathbf{S}_{y_j}] \quad (3.11)$$

where y_i and y_j are the styles of the two sentences x_1 and x_2 respectively. The *BERT-random* model obtains \mathbf{h} by subtracting the same random vector r from \mathbf{s}_1 and \mathbf{s}_2 and then concatenating them together. I have experimented with gender, age and education tasks from PASTEL dataset (Kang and Hovy, 2019) and the results are shown in Table 3.7.

Model	Accuracy		
	Gender	Age	Education
<i>BERT-model</i>	56.10	58.42	58.94
<i>BERT-style</i>	54.49	56.61	50.96
<i>BERT-random</i>	54.97	58.10	58.94

TABLE 3.7: Classifier accuracies

As we can see, after subtracting the style embeddings, I get a drop in classification accuracy suggesting that the style embeddings do capture some style information. This drop is not seen when I subtract a random vector which further provides evidence that the style embeddings capture information related to the style of the sentence.

One question that still remains is the usefulness of $s_1 - \mathbf{S}_{y_i}$ i.e the representation that remains after subtracting the style vector. I propose the following evaluation for assessing the quality of the style representation:

Style Transfer: I propose style transfer using the style representation obtained from Eq. 3.10. This style representation could be concatenated to the input sentence representation to guide the generation process.

Retrieval Techniques: I design the following two retrieval experiments to test the style representation:

1. *Retrieve Style:* In this task, I take a sentence x_1 and find k sentences with similar meaning to x_1 in all the given styles using cosine similarity between their BERT representations. I average the BERT representations of these sentences and consider this as the meaning vector (m_1) of x_1 . I retrieve the style vector \mathbf{S}_y which is closest to $\text{BERT}(x_1) - m_1$ using cosine similarity.
2. *Retrieve Sentence:* This task is performed to understand if the style representation can be used for style transformations. In this task, I take a the BERT representation of a sentence x_1 (say s_1). I get a transformed representation \hat{x}_1 where $\hat{x}_1 = s_1 - \mathbf{S}_{y_i} + \mathbf{S}_{y_j}$, where y_i is the style of x_1 and y_j is the style to which I would like to transform x_1 . My candidate set is made of k sentences from which $k - 1$ belong to style y_i and one sentence belongs to y_j . The task is to retrieve the sentence that belongs to style y_j using cosine similarity between \hat{x}_1 and the BERT representations of the candidate sentences. Note that the sentences selected for the candidate set will also be compared to x_1 and only the sentences which are close in meaning with x_1 would be chosen to belong to the candidate set.

I acknowledge that the simple averaging technique will not be effective for all types of styles, especially it may not be effective for highly content coupled styles. Hence, I plan to extend this work and design better ways of extracting style borrowing ideas from (Kumar et al., 2019; Li et al., 2015). The style representation could be used in cross domain classification tasks. I plan to test this hypothesis by testing the style representation to predict the style of a sentence in a domain other than the one used to build the style representation. For example, I can get my gender style representation from PASTEL gender dataset (Kang and Hovy, 2019) and then test this representation for the Yelp gender dataset (Prabhumoye et al., 2018).

3.6.2 Understanding Style

I would like to propose a computational approach towards understanding the different complexities of style. For example, to determine whether a style is content coupled or not, I can build a classifier which only uses bag of words. I would like to identify other features such as POS tag sequences, sentence structure, usage of function words etc which contribute in defining the style for the classifier. Based on this understanding, I can formulate different types of transformations that are possible for that style. For example, for content decoupled style like politeness, as simple delete, replace or add pipeline would suffice to perform style transfer. Contrastingly, for heavily content coupled styles like sarcasm which depends on the context, this simple transformation may not work. The specific models would be dependent on the type of transformations that need to be performed.

I propose two part analysis to understand the features and complexities of a style.

Lexical Understanding. This analysis will give us an understanding of how content dependent is the style. I propose an ablation of the classifier performance with different lexical features. I plan to analyze the performance of simple SVM classifier for each style with N-gram features to understand their contribution in discriminating between styles. Another analysis that I propose is checking the distribution of function and content words in each style. I can then mask the content words of a sentence and train a classifier based on the function words. This would give us a better understanding on the type of words the style is dependent on.

Structural Understanding. Style may not necessarily be only defined by surface level features. I hypothesize that some styles are dependent heavily on the underlying structure of the sentence. In this case, I propose an ablation of the classifier performance with various structural features of the sentence. I plan to perform experiments for prediction of the style using only the POS tag sequences of the sentence. The POS tag N-grams denote the structure of the sentence. I have shown some preliminary results of classification accuracy in Table 3.8 for the PASTEL dataset (Kang and Hovy, 2019) on the demographic attributes of gender, age and education. The results for the *BERT-model* are taken from (Kang and Hovy, 2019) and the *POS-model* corresponds to a SVM model trained only on POS unigram, bigram and trigram features.

These results suggest that the POS N-grams are highly indicative of the gender styles of Male, Female and Non-binary. I also plan to perform similar experiments using features from parse trees, as well as additional features such as depth and breadth of the trees.

	<i>BERT-model</i>	<i>POS-model</i>
Gender	73.0	71.7
Age	46.3	40.5
Education	42.5	38.0

TABLE 3.8: Comparison of classification accuracy for *BERT-model* and *POS-model*

Chapter 4

Content

Monkey selfie copyright dispute

From Wikipedia, the free encyclopedia

The monkey selfie copyright dispute is a series of disputes about the copyright status of selfies by macaques.

On 4 July 2011 several publications picked up the story and quoted Slater as describing the photographs as self-portraits. Slater said reports that a monkey ran off with his camera and "began taking self-portraits" were incorrect and that the portrait was shot when his camera had been on a tripod, with the primates playing around with a remote cable release as he fended off other monkeys.^[14]



One of the monkey selfies at issue in the dispute

Ape-rture priority photographer plays down monkey reports

amateurphotographer.co.uk
July 5, 2011

A photographer who says he witnessed monkeys taking pictures of themselves, tells Amateur Photographer (AP) that much of the media coverage has been exaggerated.

Speaking to AP, David explained that his camera had been mounted on a tripod when the primates began playing around with a remote 'cable release' as he was trying to fend off other monkeys.

FIGURE 4.1: Example of content transfer: Given existing curated text (yellow) and a document with additional relevant information (green), the task is to update the curated text (orange) to reflect the most salient updates.

Recent work in neural natural language generation (NLG) has witnessed a growing interest in controlling text for various form-related and linguistic properties, such as style (Ficler and Goldberg, 2017), affect (Ghosh et al., 2017), politeness (Sennrich et al., 2016), persona (Li et al., 2016b) voice (Yamagishi et al., 2016), grammatical correctness (Ji et al., 2017), and length (Kikuchi et al., 2016). This trend offers the promise of empowering existing authoring tools such as Grammarly, Google Smart Compose, and Microsoft Word with the ability to control

a much greater variety of textual properties, which are currently mostly limited to grammar, spelling, word choice, and wordiness. What has been relatively less explored in neural NLG research is the ability to control the generation of a current sentence not only in its *form*, but also its *content*. Historically, NLG has focused on generation from structured content such as a database or semantic representation, but I am interested in generation from free-form text or unstructured data. Consider for example Figure 4.1, which illustrates a situation where an author edits a document (here a Wikipedia article), and the goal is to generate or suggest a next sentence (shown in orange) to the author. This type of unconstrained, long-form text generation task (Mostafazadeh et al., 2016; Fan et al., 2018) is of course extremely difficult. Free-form generation can easily go astray due to two opposing factors. On one hand, ensuring that the generated output is of relatively good quality often comes at the cost of making it bland and devoid of factual content (Li et al., 2016a). On the other hand, existing techniques can help steer neural models away from blandness in order to produce more contentful outputs (using temperature sampling (Fan et al., 2018), GAN (Goodfellow et al., 2014), etc.), but often at the cost of “hallucinating” (Wiseman et al., 2017) words or concepts that are totally irrelevant. Neither situation provides a compelling experience to the user.

What is clearly missing from the aforementioned authoring scenario is the notion of *grounding*: there is often a profusion of online resources that bear at least some relevance to any given document currently being written. Much of the general-purpose world knowledge is available in the form of encyclopedias (e.g., Wikipedia), books (e.g., Project Gutenberg, Google Books), and news articles. While the generation of good quality texts without any conditioning on “external” sources (Fan et al., 2018) might be an interesting research endeavor on its own, I argue that grounding can make the generation task much easier, e.g., as shown in Figure 4.1 where a passage of a news article (green) can be reformulated considering the current context of the document (yellow) in order to produce a natural next sentence (orange). In light of this desideratum, this chapter addresses the problem of grounded text generation, where the goal is to infuse the content or knowledge from an external unstructured source (e.g., a news article as in Figure 4.1) in order to generate a follow-up sentence of an existing document. I see this as a form of *Content Transfer*, as other characteristics of the external source—such as style and linguistic form—are not controlled.

Apart from the aforementioned scenario, generation grounded in an external unstructured source of information is very useful in various other scenarios like generating factful dialogue responses from a given document, generating stories based on a plot, generating one coherent report from multiple source documents as in the case of scientific summary etc. Particularly, I am also interested in dialogue response generation. Most of the dialog systems hallucinate a response given the context. I introduce a new dataset with human-human conversations which grounds the dialogue responses in Wikipedia articles about a topic. In this case, the current context is the current dialogue history, and I am interested in generating the appropriate response from the information in the external document (Wikipedia article). Figure 4.2 shows an example of this task from the dataset collected. Although the dataset is based on the topic on

User1: The Notebook is hands-down one of my favorite movies EVER! Have you ever seen The Notebook?

User2: No I have never seen this movie. I am going to try it out now

User1: It was a heartwarming story of young love. The main characters are played by Ryan Gosling and Rachel McAdams.

User2: Ok this sounds nice. I think Ryan is a good actor.

User1: For all the praise it received, I was surprised to see that it only got a 5.7/10 on Rotten Tomatoes.

User2: That is interesting. They never get the rating correct.

User1: Ryan is a great actor, as well as Rachel McAdams. The story goes back and forth between present day and the past. Older Ryan is played by James Garner and older Rachel is played by Gena Rowlands. Yeah, Rotten Tomatoes never gets the right ratings..LOL. I always like to see the ratings but if I want to see a movie, I will watch it even if it has a bad rating.

The Notebook

From Wikipedia, the free encyclopedia

For other uses, see [Notebook \(disambiguation\)](#).

The Notebook is a 2004 romantic drama film directed by Nick Cassavetes and written by Jeremy Leven from Jan Sardi's adaptation of the 1996 novel by Nicholas Sparks. The film stars Ryan Gosling and Rachel McAdams as a young couple who fall in love in the 1940s. Their story is narrated from the present day by an elderly man (played by James Garner), telling the tale to a fellow nursing home resident (played by Gena Rowlands, who is Cassavetes's mother).

The Notebook received generally mixed reviews, but performed well at the box office and received a number of award nominations, winning eight Teen Choice Awards, a Satellite Award, and an MTV Movie Award. The film became a sleeper hit^{[3][4]} and has gained a cult following.^{[5][6]} On November 11, 2012, ABC Family premiered an extended version with deleted scenes added back into the original storyline.^[7]

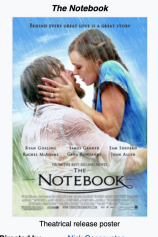
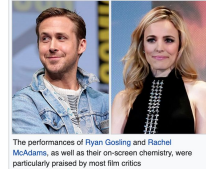
Reception [[edit](#)]

Box office [[edit](#)]

The film premiered June 25, 2004, in the United States and Canada and grossed \$13,464,745 in 2,303 theaters its opening weekend, ranking number 4 at the box office.^[8] The film grossed a total of \$115,603,229 worldwide, \$81,001,787 in Canada and the United States and \$34,601,442 in other countries.^[9] It is the 15th highest-grossing romantic drama film of all time.^[10]

Critical reception [[edit](#)]

The Notebook received a mixed reaction from film critics. The 178 reviews on review aggregator *Rotten Tomatoes* show that 53% of critics gave the film a positive review, with an average rating of 5.6/10 and the website's consensus stating "It's hard not to admire its unabashed sentimentality, but *The Notebook* is too clumsily manipulative to rise above its melodramatic clichés."^[11] At *Metacritic*, which assigns an average rating out of 100 to reviews from mainstream critics, the film currently holds an average score of 53, based on 34 reviews, which indicates "mixed or average reviews."^[12]

Theatrical release poster

Directed by [Nick Cassavetes](#)

The performances of Ryan Gosling and Rachel McAdams, as well as their on-screen chemistry, were particularly praised by most film critics

FIGURE 4.2: Example of human-human dialogue where *User1* has access to the Wikipedia document and *User2* does not. The information underlined in red is taken from the Wikipedia article by *User1*.

movies, I see the same techniques being valid for other external documents such as manuals, instruction booklets, and other informational documents.

Formally I define my task as follows: given an existing curated text s and a document d describing novel information relevant to that text, the system must produce a revised text s' that incorporates the most salient information from d . I restrict my focus to the cases where the revised text s' can be obtained by appending the new information from d to the original curated text s ¹. In particular, I assume I can transform the old curated text s into the new text s' by appending one additional update sentence x to s . This makes the same techniques applicable to the dialogue response generation task shown in Figure 4.2.

4.1 Methodology

For training data, I rely on a large dataset of existing curated text $S = \{s_1, \dots, s_n\}$, corresponding documents with novel information $D = \{d_1, \dots, d_n\}$, and the update sentences $X = \{x_1, \dots, x_n\}$. I have designed the task to generate the update sentence x_i that could be appended to the curated text s_i in order to incorporate the additional information from document d_i . The goal would be to identify new information (in particular, $d_i \setminus s_i$) that is most salient to the topic or focus of the text, then generate a single sentence that represents this information.

¹In the case of generating Wikipedia edits and similar tasks, updated information from d might demand substantial changes to s : perhaps core assumptions of s were contradicted, necessitating many removed and rewritten sentences. I postpone this complex setting to future work.

4.1.1 Generative models

A natural though difficult means of generating this additional update sentence x is to use a generative model conditioned on the information in the curated text s and the new document d . Recent methods inspired by successful neural machine translation systems have produced impressive results in abstractive summarization (Nallapati et al., 2016). Hence, my first step is to use the sequence-to-sequence encoder-decoder model (Bahdanau et al., 2015) with attention (Luong et al., 2015b) for my task. This kind of model assumes that the output sentence can be generated word-by-word. Each output word x_i^t generated is conditioned on all prior words $x_i^{<t}$ and an encoded representation of the context z :

$$\prod_t p(\hat{x}_i^t | \hat{x}_i^{<t}, z) \quad (4.1)$$

Context Agnostic Generative (CAG) Model: One simple baseline is to train a sequence-to-sequence model for the document d alone that does not directly incorporate information from the curated text s . Here, the algorithm is trained to generate the most likely update sentence $\hat{x} = \arg \max p(x|d)$. In this setting, I consider the reference document d_i as the source and the update sentence to be generated x_i as the target.

$$z = \text{Encoder}(d_i, \theta) \quad (4.2)$$

The encoder and decoder do not directly see the information from the curated text s , but the update x inherently carries some information about it. The parameters of the model are learned from updates that were authored given the knowledge of the curated text. Hence, the model may capture some generalizations about the kinds of information and locations in d that are most likely to contribute novel information to s .

Context Only Generative (COG) Model: This algorithm is trained to generate the most likely update sentence $\hat{x} = \arg \max p(x|s)$. This model is similar to CAG except that I consider the curated s_i as the source. In this setting, there is no grounding of the content to be generated.

Context Informed Generative (CIG) Model: An obvious next step is to incorporate information from the curated text s as well. I can concatenate the document and the curated text, and produce an encoded representation of this sequence.

$$z = \text{Encoder}([d_i; s_i], \theta) \quad (4.3)$$

This approach incorporates information from both sources, though it does not differentiate them clearly. Thus, the model may struggle to identify which pieces of information are novel with respect to the curated text.

To clearly identify the information that is already present in the curated text s , a model could encode s and d separately, then incorporate both signals into the generative procedure.

Context Receptive Generative (CRG) Model: My next step was to condition the generative process more concretely on the curated text s . I condition the generative process on the representation of s at each time step. Formally:

$$z_d = \text{Encoder}_d(d_i, \theta_d) \quad (4.4)$$

$$z_s = \text{Encoder}_s(s_i, \theta_s) \quad (4.5)$$

$$\hat{x}_i \sim \prod_t p(\hat{x}_i^t | [\hat{x}_i^{<t}; z_s], z_d) \quad (4.6)$$

where, θ_d and θ_s are the parameters of the encoder for the document d and encoder for the curated text s respectively, z_d and z_s are the encoded representations of the document d_i and curated text s_i respectively. At each time step of generation, the output is conditioned on the tokens generated up to the time step t concatenated with z_s . Hence, the generative process is receptive of the context at each time step.

4.1.2 Extractive models

Generative models that construct new sentences conditioned on the relevant context are compelling but have a number of modeling challenges. Such a model must both select the most relevant content *and* generate a fluent linguistic realization of this information.

I also consider extractive models: approaches that select the most relevant sentence from the document d to append to the curated text s . These approaches can focus solely on the content selection problem and ignore the difficulties of generation. This simplification does come at a cost: the most effective sentence to add might require only a subset of information from some sentence in the document, or incorporate information from more than one sentence.

Sum-Basic (SB): One common baseline is Sum-Basic, an extractive summarization technique that relies on word frequency statistics to select salient sentences (Nenkova and Vanderwende, 2005). As an initial step, unigram probabilities are computed from the set of input documents using relative frequency estimation. Then, sentences are selected one-by-one in greedy rounds until the summary budget is saturated. At each round, this model selects the most likely sentence according to the current unigram distribution. The selected sentence is added to the summary and removed from the pool of available sentences. The unigram probabilities of all words in the selected sentence are heuristically discounted (replaced by square root). Select-then-discount operations continue until the summary is written. Discounting

is crucial to prevent repetition: once a word (or ideally a concept) has been selected for the summary, it is much less likely to be picked in a subsequent round.

I use Sum-Basic as a Context Agnostic extractive model: I provide the document d as an input to the model and run Sum-Basic for exactly one round. The selected sentence is considered to be the update sentence x .

Context Informed Sum-Basic (CISB): I developed a simple modification of the Sum-basic technique to incorporate information from the curated text s as context. Initial unigram probabilities are computed using word counts from *both* the curated text *and* the document. Next, for each sentence in the curated text, I apply just the discount procedure, updating the probability distribution as if those sentences were selected. Finally, I select the single sentence from the document that is most likely according to the resulting discounted unigram probabilities. This simple modification of Sum-Basic helps select a sentence that is novel with respect to the curated text by lowering the probability of all words already present.

Extractive CAG, CIG, CRG Models: Any generative model of x can also be used as an extractive model: I simply estimate the likelihood of each sentence in the document according to the model, and select the most likely one. Generative models may fail because either they are unable to select the most relevant information, or because the resulting sentence is ill-formed. Extractive ranking circumvents all errors due to generation and can help isolate model issues.

Hybrid CAG, CIG, CRG Models: Since the document d can be quite large, a generative model may struggle to pick the most salient information based on the context. To simplify the generative modeling task, I can pre-filter the document toward only the most salient parts. I use the Context Informed Sum-Basic technique to first select the top five sentences from the document. I supply only these five sentences in place of the source document d , then apply the CAG, CIG, and CRG techniques described above.

4.2 Datasets

4.2.1 Grounded Wikipedia Edit Generation

Wikipedia can provide a naturally-occurring body of text with references to primary sources. A substantial fraction of Wikipedia sentences include citations to supporting documentation, a ripe source of data for content transfer. That said, some of the citations are quite difficult to follow or trust: broken URLs might lead to lost information; citations to books are difficult to consume given the large scope of information; etc. Therefore, cases where the reference links to some well-known news sources are considered.

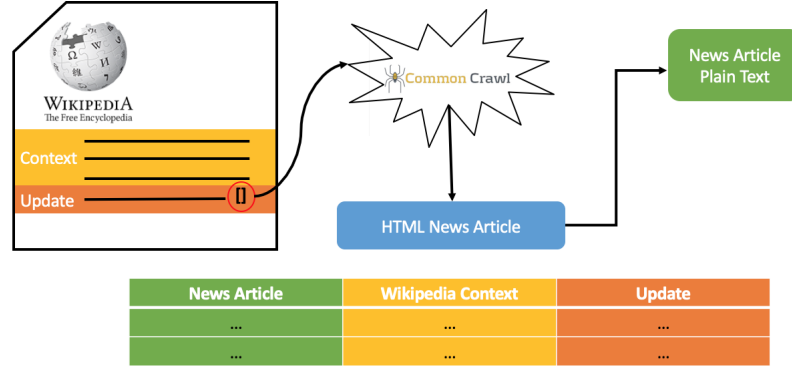


FIGURE 4.3: Dataset creation process for Wikipedia Edit Generation

Based on citation frequency, I selected a list of 86 domains,² primarily news outlets. During the data creation process I only considered citations belonging to one of these eighty six domains. This simplifying assumption is made for several reasons. First, my English Wikipedia dump contained approximately 23.7 million citation URLs belonging to 1.6 million domains; fine-grained filtering would be a daunting task. The hand-vetted list of domains is a high-precision (albeit low-recall) means of selecting clean data. Second, I wanted to ground the generated text on credible, consistent, and well-written sources of information. Furthermore, well-known domains are readily available on Common Crawl,³ leading to an easily-reproducible dataset.

Figure 4.3 illustrates the procedure used to create a dataset for the Wikipedia Edit generation task shown in Figure 4.1. For each Wikipedia article, I extracted the plain text without markdown. When encountering a citation belonging to a selected domain, I considered the sentence just before the citation to be generated based on the content of the citation. This sentence became my reference update sentence: the additional update sentence x added to the curated text s to produce the new text s' . The k sentences prior to the target sentence in the Wikipedia article were considered to be the curated text s . In this case, I used a window of $k = 3$ sentences to select the context. The cited article acted as the document d , from which the appropriate update x can be generated.

The HTML source of the citation was downloaded from Common Crawl for reproducibility and consistency. The HTML derived from Common Crawl is then processed to get the plain text of the news article. The resulting dataset C consists of aligned tuples $C = (d_i, s_i, x_i)_{i \in [1, n]}$, where n is the total number of samples in the dataset.

Alternatively, one might rely on Wikipedia edit history to create a dataset. In this setting, edits which include a new citation would act as the update x . Although this has the upside of identifying potentially complex, multi-sentence updates, preliminary analysis suggested that these edits are noisy. Editors may first generate the content in one edit, then add the citation in a subsequent edit, they may only rephrase a part of the text while adding the citation, or they

²This list is provided in the data release of this paper.

³<http://commoncrawl.org/>

Corpus	Input	Output	#Examples	Rouge-1 R
Gigaword (Graff and Cieri, 2003)	10^1	10^1	10^6	78.7
CNN/DailyMail (Nallapati et al., 2016)	10^2-10^3	10^1	10^5	76.1
WikiSum (Liu et al., 2018)	10^2-10^6	10^1-10^3	10^6	59.2
Content Transfer (this paper)	10^1-10^3	10^1-10^2	10^5	66.9

TABLE 4.1: Key characteristics of the dataset: approximate size of input and output instances, approximate dataset size, and recall of reference output against the source material, as a measure of dataset difficulty.

may check in a range of changes across the document in a single edit. My simpler sentence-based approach leads to an interesting dataset with fewer complications.

Dataset Statistics and Analysis: Table 4.1 describes some key statistics of this dataset and how it compares with other datasets used for similar tasks. The ROUGE-1 recall scores of reference output x against document d suggest this task will be difficult for conventional extractive summarization techniques.⁴ I hypothesize that during content transfer, the language in document d often undergoes substantial transformations to fit the curated text s . The average unigram overlap (after stopword removal) between the document d and the reference update sentence x is 55.79%; overlap of the curated text s and the reference update sentence x is 30.12%. This suggests the reference update sentence x can be derived from the document d , though not extracted directly. Furthermore, the content of x is very different from the content of s but appears topically related.

My dataset consists of approximately 290k unique Wikipedia articles. Some heavily-cited articles include ‘Timeline of investigations into Trump and Russia (2017)’, ‘List of England Test cricketers’, and ‘2013 in science’. I randomly split the dataset into 580k training instances, 6049 validation instances, and 50k test instances, ensuring that any Wikipedia article appearing in the train set must not appear in validation or test.

4.2.2 Grounded Dialog Generation

This task is described in Figure 4.2. In this case, the curated text s is the history of dialogue utterances up to the current state, document d is an external document based on Wikipedia articles and the task is to generate the current dialogue turn x .

To create a dataset for this task, the following were required: (1) A set of documents (2) Two humans chatting about the content of the document for more than 12 turns. I collected conversations about the documents through Amazon Mechanical Turk (AMT) and restricted the topic of the documents to be movie-related articles to facilitate the conversations. Initially, I experimented with different potential domains. Since movies are engaging and widely known,

⁴ROUGE-1 recall was computed on a sample of 50k instances from the entire dataset.

User 2:	Hey have you seen the inception?
User 1:	No, I have not but have heard of it. What is it about
User 2:	It's about extractors that perform experiments using military technology on people to retrieve info about their targets.

TABLE 4.2: An example conversation for *scenario 1*. User 1 does not have access to the document, while User 2 does.

people actually stay on task when discussing them. In fact in order to make the task interesting, I offered a choice of movies to the participants so that they are invested in the task.

Document Set Creation: I chose Wikipedia⁵ articles to create a set of documents $D = \{d_1, \dots, d_{30}\}$ for grounding of conversations. I randomly selected 30 movies, covering various genres like thriller, super-hero, animation, romantic, biopic etc. I extracted the key information provided in the Wiki article and divide it into four separate sections. This was done to reduce the load of the users to read, absorb and discuss the information in the document. Hence, each movie document d_i consists of four sections $\{e_1, e_2, e_3, e_4\}$ corresponding to basic information and three key scenes of the movie. The basic information section e_1 contains data from the Wikipedia article in a standard form such as year, genre, director. It also includes a short introduction about the movie, ratings from major review websites, and some critical responses. Each of the key scene sections $\{e_2, e_3, e_4\}$ contains one short paragraph from the plot of the movie. Each paragraph contains on an average 7 sentences and 143 words. These paragraphs were extracted automatically from the original articles, and were then lightly edited by hand to make them of consistent size and detail.

Dataset Creation

To create a dataset of conversations which uses the information from the document, involves the participation of two workers. Hence, I explore two scenarios: (1) Only one worker has access to the document and the other worker does not and (2) Both the workers have access to the document. In both settings, they are given the common instructions of chatting for at least 12 turns.

Scenario 1: One worker has document. In this scenario, only one worker has access to the document. The other worker cannot see the document. The instruction to the worker with the document is: *Tell the other user what the movie is, and try to persuade the other user to watch/not to watch the movie using the information in the document*; and the instruction to the worker without the document is: *After you are told the name of the movie, pretend you are interested in watching the movie, and try to gather all the information you need to make a decision whether to watch the movie in the end*. An example of a dialogue for this scenario is shown in Table 4.2.

⁵<http://en.wikipedia.org>

User 1:	Hi
User 2:	Hi
User 2:	I thought The Shape of Water was one of Del Toro's best works. What about you?
User 1:	Did you like the movie?
User 1:	Yes, his style really extended the story.
User 2:	I agree. He has a way with fantasy elements that really helped this story be truly beautiful.
User 2:	It has a very high rating on rotten tomatoes, too. I don't always expect that with movies in this genre.

TABLE 4.3: An example conversation for *scenario 2*. Both User 1 and User 2 have access to the Wiki document.

Scenario 2: Both workers have document. In this scenario, both the workers have access to the same Wiki document. The instruction given to the workers are: *Discuss the content in the document with the other user, and show whether you like/dislike the movie*. An example of the dialogue for this scenario is shown in Table 4.3.

Workflow: When two workers enter the chat-room, they are given only the first section on basic information e_1 of the document d_i . After they complete 3 turns (for the first section 6 turns is needed due to initial greetings), the users will be shown the next section. The workers are encouraged to discuss information in the new section, but are not constrained to do so.

Dataset	# Utterances	Avg. # of Turns
CMU-DoG	130,000	31.00
Persona-chat (Zhang et al., 2018)	164,356	14.00
Cornell Movie (Danescu-Niculescu-Mizil and Lee, 2011)	304,713	1.38
Frames dataset (El Asri et al., 2017)	19,986	15.00

TABLE 4.4: Comparison with other datasets. The average number of turns are calculated as the number of utterances divided by the number of conversations for each of the datasets.

Dataset Statistics

The dataset consists of total 4112 conversations with an average of 21.43 turns. The number of conversations for *scenario 1* is 2128 and for *scenario 2* it is 1984. I consider a turn to be an exchange between two workers (say w_1 and w_2). Hence an exchange of w_1, w_2, w_1 has 2 turns (w_1, w_2) and (w_2, w_1). I show the comparison of my dataset as **CMU Document Grounded Conversations** (CMU-DoG) with other datasets in Table 4.4. One of the salient features of CMU-DoG dataset is that it has mapping of the conversation turns to each section of the document, which can then be used to model conversation responses. Another useful aspect is that I report the quality of the conversations in terms of how much the conversation adheres to the information in the document.

Percentile	20	40	60	80	99
BLEU	0.09	0.20	0.34	0.53	0.82

TABLE 4.5: The distribution of BLEU score for conversations with more than 10 turns.

Split Criteria: I measure the quality of the conversations using BLEU (Papineni et al., 2002) score because I wanted to measure the overlap of the turns of the conversation with the sections of the document. Hence, a good quality conversation should use more information from the document than a low quality conversation. I then divide the dataset into three ratings based on this measure. The BLEU score is calculated between all the utterances $\{x_1, \dots, x_n\}$ of a conversation C_i and the document d_i corresponding to C_i . Incomplete conversations that have less than 10 turns are eliminated. The percentiles for the remaining conversations are shown in Table 4.5. I split the dataset into three ratings based on BLEU score.

Rating 1: Conversations are given a rating of 1 if their BLEU score is less than or equal to 0.1. I consider these conversations to be of low-quality.

Rating 2: All the conversations that do not fit in rating 1 and 3 are marked with a rating of 2.

Rating 3: Conversations are labeled with a rating of 3, only if the conversation has more than 12 turns and has a BLEU score larger than 0.587. This threshold was calculated by summing the mean (0.385) and the standard deviation (0.202) of BLEU scores of the conversations that do not belong rating 1.

The average BLEU score for workers who have access to the document is 0.22 whereas the average BLEU score for the workers without access to the document is 0.03. This suggests that even if the workers had external knowledge about the movie, they have not extensively used it in the conversation. It also suggests that the workers with the document have not used the information from the document verbatim in the conversation. Table 4.6 shows the statistics on the total number of conversations, utterances, and average number of utterances per conversation and average length of utterances for all the three ratings.

Dataset analysis: I perform two kinds of automated evaluation to investigate the usefulness of the document in the conversation. The first one is to investigate if the workers use the information from the document d_i in the conversation. The second analysis is to show that the document adds value to the conversation. Let the set of tokens in the current utterance x_i be \mathbf{N} , the set of tokens in the current section e_i be \mathbf{M} , the set of tokens in the previous three utterances be \mathbf{H} , and the set of stop words be \mathbf{S} . In *scenario 1*, I calculate the set operation *new tokens* as I find that on average 0.78 new tokens (excluding stop words) are introduced in the current utterance x_i that are present in the current section e_i but are not present in the prior

	Rating 1	Rating 2	Rating 3	Rating 2 & 3
Total # of conversations	1443	2142	527	2669
Total # of utts	28536	80104	21360	101464
Avg. # utts/conversation	19.77(13.68)	35.39(8.48)	40.53(12.92)	38.01(9.607)
Avg. length of utterance	7.51(50.19)	10.56(8.51)	16.57(15.23)	11.83(10.58)

TABLE 4.6: The statistics of the dataset. Standard deviation in parenthesis.

three utterances. The average length of x_i is 12.85 tokens. Let the tokens that appear in all the utterances (x_i, \dots, x_{i+k}) corresponding to the current section e_i be \mathbf{K} and the tokens that appear in all the utterances (x_i, \dots, x_{i+p}) corresponding to the previous section e_{i-1} be \mathbf{P} . In *scenario 2*, I calculate the set operation *new tokens* as I find that on average there are 5.84 common tokens in the utterances that are mapped to the current section e_i and in e_i but are not present in the utterances of the previous section e_{i-1} . The average length of the utterances in a section e_i is 117.12 tokens. These results show that people use the information in the new sections and are not fixated on old sections. It also shows that they use the information to construct the responses.

4.3 Results

I evaluate the models using both automated metrics and, for a subset of promising systems, human assessment. One key evaluation is the similarity between the model generated sentence and reference sentence. Human judges are also asked to assess grammaticality and coherence.

6

4.3.1 Automated Evaluation

Grounded Wikipedia Edit Generation: The primary automated evaluation metric for system-generated update sentences is ROUGE-L F1 against reference update sentence,⁷ though I also include BLEU (Papineni et al., 2002) and METEOR (Denkowski and Lavie, 2011) as additional indicators. ROUGE is a standard family of metrics for summarization tasks; ROUGE-L measures the longest common subsequence between the system and the reference, capturing both lexical selection and word order.

Table 4.7 illustrates that this task is quite difficult for extractive techniques. Furthermore, the results emphasize the importance of having curated text as context when generating the update. In all experimental conditions, models aware of context perform much better than models agnostic of it. In contrast to Liu et al. (2018), generative approaches outperformed hybrid, likely because I only had a single input document. Extractive CAG, CIG, and CRG all outperformed

⁶Details about hyper-parameters, generated examples and examples of human dialogues are provided in Appendix B.

⁷I use the pyrouge toolkit along with ROUGE-1.5.5: <https://github.com/bheinzerling/pyrouge>

Model	ROUGE-L	BLEU	METEOR
Sum-Basic	5.6 (5.6–5.7)	0.6	2.0
Context Informed Sum-Basic (CISB)	7.0 (7.0–7.1)	1.0	2.8
Context Agnostic Generative (CAG)	9.1 (9.0–9.2)	1.2	4.6
Context Only Generative (COG)	13.5 (13.4–13.6)	1.7	3.5
Context Informed Generative (CIG)	16.0 (15.9–16.1)	3.5	5.3
Context Receptive Generative (CRG)	14.7 (14.6–14.8)	2.6	4.5
Hybrid CAG	8.0 (7.9–8.0)	1.0	3.8
Hybrid CIG	15.0 (14.9–15.1)	2.7	4.7
Hybrid CRG	13.5 (13.4–13.6)	2.3	4.1
Extractive CAG	9.3 (9.2–9.3)	1.1	3.2
Extractive CIG	9.3 (9.2–9.3)	1.1	3.2
Extractive CRG	9.2 (9.1–9.3)	1.1	3.2
<i>Oracle</i>	<i>28.8</i> (28.7–29.0)	<i>11.0</i>	<i>10.9</i>

TABLE 4.7: Automated metrics; 95% confidence interval in parentheses.

both Sum-Basic and the context informed variant. Extractive CAG was on-par with generative CAG, suggesting the generated sentences were of reasonable quality. However, generative CIG and CRG were substantially better: rewriting to match context was beneficial.

The *Oracle* system of Table 4.7 aims to establish an upper limit attainable by extractive methods, using the following oracle experiment: For each test instance (d_i, s_i, x_i) , I enumerate each extracted sentence e of document d_i and select the one with highest ROUGE-L score as *Oracle*’s update sentence \hat{x}_i (i.e., $\hat{x}_i = \arg \max_{e \in d_i} \text{ROUGE-L}(x_i, e)$).

Note this yields a very optimistic upper bound, as the same ground truth x_i is used both to select an extractive sentence from a large pool of candidates and for final automatic metric scoring.⁸ Nevertheless, these oracle results let me draw two conclusions: (1) They give a better perspective to assess the non-oracle systems, and I believe that their seemingly low automatic evaluation scores are quite reasonable relative to the optimistic upper bound (e.g., CIG’s ROUGE-L’s score is 55% of the oracle). (2) The oracle results suggest that humans are substantially changing the surface realization as they summarize for Wikipedia, as otherwise the oracle results would be much closer to maximum metric scores (i.e., 100%). This shows that extractive methods are not enough for this task, justifying my use of generation techniques.

Grounded Dialog Generation: To automatically evaluate the fluency of the models, I use perplexity measure. I build a language model on the train set of responses using ngrams up to an order of 3⁹. The Context Only Generative (COG) model which generates the dialogue

⁸Previous work has shown that this type of oracle can yield upper bounds that are unrealistically high, and they tend to be above human performance (Och et al., 2004, Table 1). One remedy suggested by Och et al. is a round-robin oracle ensuring that the reference (ground truth) used by the argmax is distinct from that of the final automatic evaluation, but that scheme is only possible with a multi-reference test set.

⁹I use the SRILM toolkit (Stolcke, 2002)

response based on the previous dialogue turn only, achieves a perplexity of 21.8. The Context Receptive Generative (CRG) model on the other hand, which provides the section information as an additional input to each time step of the decoder, achieves a perplexity of **10.11**. This indicates that including the sections of document helps in the generation process.

4.3.2 Human Evaluations

Grounded Wikipedia Edit Generation

For careful evaluation of the performance of the most promising configurations (CAG and CIG models) I also asked human judges for quality assessments. I solicited several types of evaluation, including two relative comparisons between pairs of system outputs and an absolute quality evaluation of individual system outputs.

Close to reference (Relative): The first relative comparison measured how accurately the generated update reflected information in the reference update. Here, the annotators saw only the reference update sentence and the outputs of two systems labeled *A* and *B* in a randomized order. I asked the annotators “Which system output is closest in meaning to the reference update?” The annotators could pick system *A*, system *B*, or indicate that neither was preferred. This is a simple evaluation task though potentially biased toward the sole reference update.

Coherent to context (Relative): The second relative comparison measured whether the generated output contained salient information from the document written in a manner appropriate to the curated text. The annotators saw the document *d*, the curated text *s*, and the outputs of the two systems *A* and *B*, again in a random order. They were asked, “Which system output is more accurate relative to the background information given in the snippet of the article?” Each judge had to consider whether the information fits with the curated text and also whether system-generated content could be supported by the document.

Four human judges each annotated 30 unique output pairs for these two relative comparison settings, a total of 240 relative judgments. Table 4.8 shows the results: the context-aware CIG system was substantially better in both settings.

DUC Guidelines (Absolute): In addition, I performed an absolute quality evaluation following the guidelines from DUC 2007.¹⁰ Each judge was presented with a single system output, then they were asked to evaluate five aspects of system output: grammaticality, non-redundancy, referential clarity, focus, and structure/coherence. For each aspect, the judge provided an assessment on a five-point scale: (1) Very Poor, (2) Poor, (3) Barely Acceptable, (4)

¹⁰<http://duc.nist.gov/duc2007/quality-questions.txt>

Evaluation task	prefer		
	CAG	neither	CIG
Close to reference	15.8%	53.3%	30.8%
Coherent to context	7.5%	53.3%	39.2%

TABLE 4.8: Human preferences of CAG vs. CIG.

Model	Grammaticality	Non-redundancy	Referential Clarity	Focus	Structure
CAG	2.6	1.8	2.7	2.6	2.4
CIG	4.3	3.9	3.6	3.5	3.2

TABLE 4.9: Human absolute quality assessments.

Document (News Article)
sequels are fairly new to bollywood, but director sanjay gadhvi realised there was cash to be made from resurrecting his hit action thriller dhoom, by casting sexy young stars like hrithik rosha, aishwarya rai and abhishek bachchan in an even bigger game of cops and robbers...that the twist in dhoom 2's tail is not explained is yet another shortcoming. it's only roshan's charismatic performance as the criminal mastermind, and the sizzling chemistry he shares with rai's sassy cohort, that rescues this adventure from becoming an elongated tourism commercial.
Curated Text (Wikipedia Context)
it makes no lasting contributions to world cinema, but if two-and-a-half hours of disposable entertainment are all you're after, you could do far worse. "l.a. weekly's david chute stated the film was, "a movie meal as satisfying as this one can make you feel that nothing else matters." jaspreet pandohar of the bbc gave it a two-star rating, writing "by roping in acclaimed action director alan amin to take care of the thrills and spills, you'd expect gadhvi to have spent time crafting out a sophisticated storyline instead of simply sending his cast on a cat-and-mouse chase around the globe.
Reference Update
it's only roshan's charismatic performance as the criminal mastermind, and the sizzling chemistry he shares with rai's sassy cohort, that rescues this adventure from becoming an elongated tourism commercial."
Generated Update
it's only roshan's finest performance as the criminal terrorist, and the sizzling chemistry he shares with rai's sassy anatomy, that attues this adventure from becoming an elongated tourism commercial."

FIGURE 4.4: Example of good quality generation, where the system-generated update is close to the reference.

Good, (5) Very Good. I gathered 120 additional judgments in this setting (4 judges, 30 outputs). Again, context-aware CIG substantially outperforms CAG across the board, as seen in Table 4.9.

Observations: Systems unaware of the curated text s tend to generate long updates with repeated frequent words or phrases. Consider the ratio of unique tokens over the total number of tokens in the generated output, which is denoted by R . A small R indicates many repeated tokens. I find that 88% of the time this ratio R falls below 0.5 for the CAG model, i.e. for 88% instances, more than 50% of the words in the generated output are repeats. This number is relatively small – 14% for CIG and 20% for CRG – in context aware models. In the reference updates only 0.21% instances repeat more than 50% of words.

Document (News Article)

anne kirkbride, who portrayed bespectacled, gravelly-voiced deirdre barlow in coronation street for more that four decades, **has died. the 60-year-old**, whose first appearance in the soap opera was in 1972, died in **a manchester hospital** after a short illness.... kirkbride had left the soap opera after **she was diagnosed with non-hodgkin's lymphoma** in 1993 but returned some months later after treatment and spoke candidly about how she had struggled with depression following the diagnosis...

Curated Text (Wikipedia Context)

in 1993, kirkbride was diagnosis with non-hodgkin's lymphoma. she spoke to the british press about her bout of depression following the diagnosis. she was cured within a year of being diagnosed.

Reference Update

anne kirkbride died of breast cancer in a manchester hospital on 19 january 2015, aged 60.

Generated Update

she was diagnosed with non-hodgkin's lymphoma.

FIGURE 4.5: Example of lower-quality output: the generated update unnecessarily restates information yet misses the most salient detail from the document.

Reference Update	Generated Update
1. rob brydon, the comedian was born in baglan.	he was born in baglan.
2. in may 2014 he was diagnosed with prostate cancer.	st. clair was diagnosed with prostate cancer.
3. he was the first black player to have played a game in the national basketball association.	he was the first african-american to play in the national basketball association.
3. on april 3, 2014, manning signed a one-year deal with the cincinnati bengals.	on march 9, 2014, manning signed a one-year contract with the cincinnati bengals.
4. on oct 10, 2013, barrett signed with the memphis grizzlies.	on feb 9, 2013, barrett signed with the memphis grizzlies.
6. anne kirkbride died of breast cancer in a manchester hospital on 19 january 2015, aged 60.	she was diagnosed with non-hodgkin's lymphoma.
5. some people think elvis is still alive, but most of us think he's dead and gone."	some people think elvis, but most of us think he's dead and gone."
6. it's always the goal of the foreign-language film award executive committee to be as inclusive as possible."	it's always the goal of the foreign- entry film award executive to be as possible."

TABLE 4.10: Example generations from the CIG system, paired with the human generated updates.

Figures 4.4 and 4.5 show good and bad examples generated by the CIG model along with the document, curated text and the reference update. Table 4.10 has a set of updates generated by the CIG model as well as the reference update. As can be seen in examples 3 and 4, the CIG model misplaces the date but correctly generates the remaining content. In examples 1 and 2, the CIG model appears to successfully select the correct pronouns for co-reference resolution, though it gets confused as to when to use the pronoun or the named entity. Examples 5 and 6 represent failure cases due to missing words.

Grounded Dialog Generation

I perform two kinds of human evaluations to evaluate the quality of predicted utterances – engagement and fluency. These experiments are performed on Amazon Mechanical Turk.

Engagement: I set up a pairwise comparison following [Bennett \(2005\)](#) to evaluate the engagement of the generated responses. The test presents the chat history (1 utterance) and then, in random order, its corresponding response produced by the COG and CRG models. A third option “No Preference” was given to participants to mark no preference for either of the generated responses. The instruction given to the participants is “Given the above chat history as context, you have to pick the one which can be best used as the response based on the engagingness.” I randomly sample 90 responses from each of the COG and CRG models. Each response was annotated by 3 unique workers and I take majority vote as the final label. The result of the test is that COG generated responses were chosen only 36.4% times as opposed to CRG generated responses which were chosen **43.9%** and the “No Preference” option was chosen 19.6% of times. This result shows the information from the sections improves the engagement of the generated responses.

Fluency: The workers were asked to evaluate the fluency of the generated response on a scale of 1 to 4, where 1 is unreadable and 4 is perfectly readable. I randomly select 120 generated responses from each model and each response was annotated by 3 unique workers. The COG model got a low score of 2.88, contrast to the CRG score of **3.84**. This outcome demonstrates that the information in the section also helps in guiding the generator to produce fluent responses.

4.4 Related Work

The proposed content transfer task is clearly related to a long series of papers in summarization, including recent work with neural techniques ([Rush et al., 2015](#); [Nallapati et al., 2016](#)). In particular, one recent paper casts the the task of generating an entire Wikipedia article as a multi-document summarization problem ([Liu et al., 2018](#)). Their best-performing configuration was a two-stage extractive-abstractive framework; a multi-stage approach helped circumvent the difficulties of purely abstractive methods given quite large input token sequences.

Looking beyond the clear task similarity of authoring Wikipedia style content, there are several crucial differences in my approach. First, the goal of that paper is to author the whole page, starting from nothing more than a set of primary sources, such as news articles. In practice, however, Wikipedia articles often contain information outside these primary sources, including common sense knowledge, framing statements to set the article in context, and inferences made from those primary sources. My task restricts the focus to content where a human editor

explicitly decided to cite some external source. Hence, it is much more likely that the resulting summary can be derived from the external source content. Furthermore, I focus on the act of adding information to existing articles, rather than writing a complete article without any context. These two scenarios are clearly useful yet complementary: sometimes people want to produce a new reference text where nothing existed before; in other cases the goal is to maintain and update an existing reference.

Another closely related task is update summarization (Dang and Owczarzak, 2008), where systems attempt to provide a brief summary of the novel information in a new article assuming the user has read a known set of prior documents. My focus on curating an authoritative resource is a substantial difference. Also my datasets are substantially larger, enabling generative models to be used in this space, where prior update summarization techniques have been primarily extractive (Fisher and Roark, 2008; Li et al., 2015).

For any generation task, it is important to address both the content (‘what’ is being said) as well its style (‘how’ it is being said). Recently, a great deal of research has focused on the ‘how’ (Li et al., 2018a; Shen et al., 2017), including efforts to collect a parallel dataset that differs in formality (Rao and Tetreault, 2018), to control author characteristics in the generated sentences (Prabhumoye et al., 2018), to control the perceived personality traits of dialog responses (Zhang et al., 2018). I believe this research thread is complementary to my efforts on generating the ‘what’.

Another form of content transfer bridges across modalities: text generation given schematized or semi-structured information. Recent research has addressed neural natural language generation techniques given a range of structured sources: selecting relevant database records and generating natural language descriptions of them (Mei et al., 2016), selecting and describing slot-value pairs for task-specific dialog response generation (Wen et al., 2015), and even generating Wikipedia biography abstracts given Infobox information (Lebret et al., 2016). My task, while grounded in external content, is different in that it leverages *linguistic* grounding as well as prior text context when generating text. This challenging setting enables a huge range of grounded generation tasks: there are vast amounts of unstructured textual data.

4.5 Proposed Work

In spite of the advances in building good representations of language and knowledge, content grounded generation task is not solved yet. Most of the work in this domain has been constrained to defining new tasks and datasets to observe content grounded generation. In this thesis, I propose two new dimensions to improve this task: a new model for grounded generation and a new metric to evaluate neural generative models for grounded generation task.

Methodology: Since most of the current works (Prabhumoye et al., 2019; Dinan et al., 2018; Ghazvininejad et al., 2018; Zhou et al., 2018) focused on defining the task and supplying the dataset, the models proposed are extensions of existing models which are adopted for this task. Most of these models use two separate encoders to encode the context and the external source of knowledge. These two representations are combined by either concatenation, summation or passing them through another neural network (typically a transformer or a linear transformation). I propose to define a model which attends over the two encodings in an efficient manner in the generation process.

Formally if \mathbf{o}_t is the output representation of the generator at time step t and I have attention vectors \mathbf{a}_t^c of the input and \mathbf{a}_t^s of the external knowledge. One solution could be to decide whether to attend over the input representations or the external source p_t as in Eq. 4.7. You can then calculate the attention weights as the weighted average over all the attention vectors (Eq. 4.8) and concatenate to \mathbf{o}_t to get the final output representation $\tilde{\mathbf{o}}_t$ (Eq. 4.9). This $\tilde{\mathbf{o}}_t$ will be then projected to vocabulary size to get the probability distribution over the words post softmax.

$$p_t = \sigma(\mathbf{W}_o \mathbf{o}_t) \quad (4.7)$$

$$\tilde{p}_t = p_t \mathbf{W}_1 \mathbf{a}_t^c + (1 - p_t) \mathbf{W}_2 \mathbf{a}_t^s \quad (4.8)$$

$$\tilde{\mathbf{o}}_t = \tanh(\mathbf{W}_p [\tilde{p}_t; \mathbf{o}_t]) \quad (4.9)$$

Here, \mathbf{W}_o , \mathbf{W}_1 , \mathbf{W}_2 and \mathbf{W}_p are model parameters.

Alternatively, I can also borrow ideas on local attention from (Luong et al., 2015a). If the sequence length of input is N and that of the external document is M , then I calculate p_t as follows:

$$p_t = (N + M) \cdot \sigma(\mathbf{v}_p^T \tanh(\mathbf{W}_o \mathbf{o}_t)) \quad (4.10)$$

Here, \mathbf{v}_p^T and \mathbf{W}_o are model parameters. In this case, p_t predicts the position of the word where the model should pay attention. As a result, $p_t \in [0, N + M]$. I then calculate \tilde{p}_t by averaging the attention vectors within a window k of position p_t .

Although this alternative seems appealing in terms of reducing the computational and memory cost, it might be hard to learn to predict the position of focus with the sigmoid function. I want to explore other options along similar lines which could both decrease the memory usage as well as learn good position predictions. In particular, I would like to explore convolutional neural networks (CNNs) which could provide me with regions of the external document to pay attention to. The outputs of the convolutions could themselves serve as attention weights for the decoding process. I can also explore similar approaches using the multi-headed attention mechanism in (Vaswani et al., 2017) but this might not reduce the memory cost.

Evaluation: The current works do not evaluate the generative models for how much the content is transferred from the external source to the generated output. This is a crucial dimension

along which the content grounded generation models should be evaluated. In (Prabhumoye et al., 2019), I propose two dimensions along which the grounded generation is evaluated: *close to reference* and *coherent to context*. Close to reference measures how close the generated output is to the reference text. Coherent to context measures whether the generated text contains key information from the external document and fits the input context. This has been measured by human judges using A/B testing (Bennett, 2005) comparing two models. I propose an automatic metric for these two evaluations.

I propose to use an information extraction system like OpenIE (Angeli et al., 2015), frames based on intent, effect and reaction (Bosselut et al., 2019), or keyword extraction system (RAKE) (Rose et al., 2010) to extract information. Let the information extracted from the reference be \mathbf{i}_r , the input text be \mathbf{i}_s , the external document be \mathbf{i}_d and the generated text be \mathbf{i}_g . For the *close to reference* metric, I can compare \mathbf{i}_r and \mathbf{i}_g using different metrics like Jaccard similarity score or BLEU. I can use similar metric to compare \mathbf{i}_g with \mathbf{i}_d and \mathbf{i}_s for the *coherent to context* metric. Alternatively, I can calculate cosine similarity between the BERT representations of \mathbf{i}_r and \mathbf{i}_g , and \mathbf{i}_g with \mathbf{i}_d and \mathbf{i}_s .

Chapter 5

Structure

In the Mesopotamian era writing began as a consequence of political expansion, which needed reliable means for transmitting information, maintaining financial accounts, keeping historical records, and similar activities. In the current millennia, writing serves multiple functions which include - improvised additional capacity for the limitations of human memory (e.g. recipes, reminders, logbooks, the proper sequence for a complicated task or important ritual), dissemination of ideas (as in an essay, manifesto etc), imaginative narratives and other forms of storytelling, personal or business correspondence, and lifewriting (e.g., a diary or journal). Note that all of these functions of writing ranging from the ancient cultures to the current day, need the ideas or narratives in the written text to be organized in a logically coherent structure. The arrangement of the words and sentences in a document come together to convey the purpose of the text. My goal is to model this arrangement of text.

Language in the real world has structure. Written texts have constrained structures that vary by genre. Newspaper articles, for instance, typically have an “inverted pyramid” structure: a recent event, followed by the most relevant details of that event, followed by secondary background information at the end of the article. Wikipedia articles, by contrast, are often chronological, beginning with the earliest major historical event on a topic and proceeding sequentially. Various frameworks have been designed to understand document structures. Document structures have been modeled as trees based on the relations between the sentences in Rhetorical Structure Theory (RST; [Mann and Thompson \(1988\)](#)), as graphs ([Wolf and Gibson, 2006](#)), and as entity grid model based on transitions ([Barzilay and Lapata, 2008](#)). Each of these frameworks capture different aspects of structure - structure between two consecutive sentences is captured by local coherence, the relation of a sentence with other sentences of the document captures global structure as in the case of RST, structure can be captured between the ordering of the paragraphs in the document. Structure could also mean the ordering of events in a document, the transition of topics in a document or dialogue, the transition of scenes or a plot in a story, the chronology of events in a Wikipedia article etc.

Document structures are useful in modeling and interpreting various Natural Language Processing tasks. They are important for summarization (Barzilay and McKeown, 2005), automated essay scoring (Burstein et al., 2010; Miltsakaki and Kukich, 2004), question-answering (Verberne et al., 2007), text planning (Hovy, 1988; Marcu, 1997) and document classification (Liu and Lapata, 2018). I primarily care about the sentence ordering sub-task which is important to understand document structures. Sentence ordering is the task of arranging sentences into an order which maximizes the coherence of the text (Barzilay and Lapata, 2008). This subtask provides us insights into modeling the ordering of events in a document. Recent work has modeled this task as a sequence generation task using hierarchical neural models. I have framed the task as a constraint learning problem. I train a model which learns to predict the correct constraint given a pair of sentences. The constraint learnt by my model is the relative ordering between the two sentences. Given a set of constraints between the sentences of a document, I find the right order of the sentences by using sorting techniques. Simple sorting techniques can outperform the previous approaches by a large margin given that it has good sentence representations. The bottleneck for most of the hierarchical models is memory required by the representations of all the sentences and the representation of the paragraph. The new framing also obviates these memory issues.

5.1 Methodology

For this task I have a set of N documents $\mathcal{D} = \{d_1, \dots, d_N\}$. Let the number of sentences in each document d_i be denoted by v_i , where $\forall i, v_i \geq 1$. The task can be formulated as - If you have a set $\{s_{o_1}, \dots, s_{o_{v_i}}\}$ of v_i sentences in a random order where the random order is $\mathbf{o} = [o_1, \dots, o_{v_i}]$, then the task is to find the right order of the sentences $\mathbf{o}^* = [o_1^*, \dots, o_{v_i}^*]$. Prior work (Logeswaran et al., 2018b; Cui et al., 2018) learns to predict the sequence of the correct order \mathbf{o}^* . In this formulation of the task, I have \mathcal{C}_i set of constraints for document d_i . These constraints \mathcal{C}_i represent the relative ordering between every pair of sentences in d_i . Hence, I have $|\mathcal{C}_i| = \binom{v_i}{2}$. For example, if a document has four sentences in the correct order $s_1 < s_2 < s_3 < s_4$, then I have six set of constraints $\{s_1 < s_2, s_1 < s_3, s_1 < s_4, s_2 < s_3, s_2 < s_4, s_3 < s_4\}$. Constraints \mathcal{C}_i are learnt using a classifier neural network described in (§5.1.2). I finally find the right order \mathbf{o}^* using topological sort on the relative ordering between all the \mathcal{C}_i pairs of sentences.

5.1.1 Topological Sort

Topological sort (Tarjan, 1976) is a standard algorithm for linear ordering of the vertices of a directed graph. The sort produces an ordering $\hat{\mathbf{o}}$ of the vertices such that for every directed edge $u \rightarrow v$ from vertex u to vertex v , u comes before v in the ordering $\hat{\mathbf{o}}$. I use the depth-first search based algorithm which loops through each node of the graph, in an arbitrary order. The algorithm visits each node n and prepends it to the output ordering $\hat{\mathbf{o}}$ only after recursively

calling the topological sort on all descendants of n in the graph. The algorithm terminates when it hits a node that has been visited or has no outgoing edges (i.e. a leaf node). Hence, I am guaranteed that all nodes which depend on n are already in the output ordering \hat{o} when the algorithm adds node n to \hat{o} .

I use topological sort to find the correct ordering \mathbf{o}^* of the sentences in a document. The sentences can represent the nodes of a directed graph and the directed edges are represented by the ordering between the two sentences. The direction of the edges are the constraints predicted by the classifier. For example, if the classifier predicts the constraint that sentence s_1 precedes s_2 , then the edge $s_1 \rightarrow s_2$ would be from node of s_1 to s_2 .

This algorithm has time complexity of $O(v_i + |\mathcal{C}_i|)$ for a document d_i . In my current formulation, all the constraints are predicted before applying the sort. Hence, I have to consider all the $|\mathcal{C}_i| = \binom{v_i}{2}$ edges in the graph. The time complexity of my current formulation is $O(v_i^2)$. But the same technique could be adopted using a Merge Sort (Knuth, 1998) algorithm in which case the time complexity would be $O(v_i \log v_i)$. In this case, the sort algorithm is applied first and the constraint is predicted only for the two sentences for which the relative ordering is required during the sort time.

5.1.2 Constraint Learning

I build a classifier to predict a constraint between two sentences s_1 and s_2 (say). The constraint learnt by the classifier is the relative ordering between the two sentences. Specifically, the classifier is trained to predict whether s_2 follows s_1 or not i.e the classifier predicts the constraint $s_1 < s_2$.

BERT based Representation (B-TSort): I use the Bidirectional Encoder Representations from Transformers (BERT) pre-trained uncased language model (Devlin et al., 2019) and fine-tune it on each dataset using a fully connected perceptron layer. Specifically, I leverage the Next Sentence Prediction objective of BERT and get a single representation for both sentences s_1 and s_2 . The input to the BERT model is the sequence of tokens of sentence s_1 , followed by the separator token '[SEP]', followed by the sequence of tokens for sentence s_2 . I use the pooled representation for all the time steps.

LSTM based Representation (L-TSort): In this model I get two separate representations \mathbf{h}_1 and \mathbf{h}_2 for s_1 and s_2 from a bi-directional LSTM encoder, respectively. I pass the concatenation of \mathbf{h}_1 and \mathbf{h}_2 as input to a two layers of perceptron for constraint prediction. This model is trained to gain insight on the contribution of pre-trained sentence representations for the constraint prediction formulation of the task.

Dataset	Length Statistics			Data split			Vocabulary
	min	mean	max	train	valid	test	
NIPS	2	6.0	15	2248	409	402	16721
AAN	1	5.0	20	8569	962	2626	34485
NSF	2	8.9	40	96070	10185	21580	334090
SIND	5	5.0	5	40155	4990	5055	30861

TABLE 5.1: Dataset Statistics

5.2 Experiments

This section describes the datasets, the evaluation metric and the results of my experiments.

5.2.1 Datasets

The dataset statistics for all the datasets are shown in Table 5.1.

NSF, NIPS, AAN abstracts: These three datasets contain abstracts from NIPS papers, ACL papers, and the NSF Research Award Abstracts dataset respectively and are introduced in (Logeswaran et al., 2018b). The paper also provides details about the statistics and processing steps for curating these three datasets.

SIND caption: I also consider the SIND (Sequential Image Narrative Dataset) caption dataset (Huang et al., 2016) used in the sentence ordering task by (Gong et al., 2016). All the stories in this dataset contain five sentences each and I only consider textual stories for this task.

5.2.2 Baselines

Attention Order Network (AON): This is the current state-of-the-art model (Cui et al., 2018) which formulates the sentence ordering task as a order prediction task. It uses a LSTM based encoder to learn the representation of a sentence. It then uses a transformer network based paragraph encoder to learn a representation of the entire document. It then decodes the sequence of the order by using a LSTM based decoder.

BERT Attention Order Network (B-AON). To have a fair comparison between my model and the AON model, I replace the LSTM based sentence representation with the pre-trained uncased BERT model. This model plays a pivotal role of giving us an insight into how much improvement in performance I get only due to BERT.

5.2.3 Evaluation Metric

Perfect Match (PMR): It is the strictest metric and calculates the percentage of samples for which the entire sequence was correctly predicted (Chen et al., 2016). $PMR = \frac{1}{N} \sum_{i=1}^N 1\{\hat{\mathbf{o}}^i = \mathbf{o}^{*i}\}$, where N is the number of samples in the dataset. This is the strictest metric and gives us an absolute accuracy of the whole order being predicted correctly.

Sentence Accuracy (Acc): It is a stringent metric and measures the percentage of sentences for which their absolute position was correctly predicted (Logeswaran et al., 2018b). $Acc = \frac{1}{N} \sum_{i=1}^N \frac{1}{v_i} \sum_{j=1}^{v_i} 1\{\hat{\mathbf{o}}_j^i = \mathbf{o}_j^{*i}\}$, where v_i is the number of sentences in the i^{th} document. This is a stringent metric and tells us the percent of sentences within a document that we can predict the right order for.

Kendall Tau (Tau): This metric quantifies the distance between the predicted order and the correct order in terms of the number of inversions (Lapata, 2006). It calculates the number of inversions required by the predicted order to reach the correct order. $\tau = 1 - 2I/\binom{v_i}{2}$, where I is the number of pairs in the predicted order with incorrect relative order and $\tau \in [-1, 1]$.

Rouge-S (R-S): It calculates the percentage of skip-bigrams for which the relative order is predicted correctly (Chen et al., 2016). Skip-bigrams are the total number of pairs $\binom{v_i}{2}$ in a document. Note that it does not penalize any arbitrary gaps between two sentences as long as their relative order is correct. $Rouge-S = \frac{1}{\binom{v_i}{2}} \text{Skip}(\hat{\mathbf{o}}) \cap \text{Skip}(\mathbf{o}^*)$, where the $\text{Skip}(\cdot)$ function returns the set of skip-bigrams of the given order.

Longest Common Subsequence (LCS): It calculates the ratio of longest correct sub-sequence (Gong et al., 2016) (consecutiveness is not necessary, and higher is better).

Human Evaluation I introduce a human evaluation experiment to assess the orders predicted by the models. I set up a manual pairwise comparison following (Bennett, 2005) and present the human judges with two orders of the same piece of text. The judges are asked “Pick the option which is in the right order according to you.” They can also pick a third option ‘No Preference’ which corresponds to both the options being equally good or bad. In total I had 100 stories from the SIND dataset¹ annotated by 10 judges. I setup three pairwise studies to compare the B-TSort vs AON order, B-TSort vs Gold order and AON vs Gold order (Gold order is the actual order of the text).

¹I choose SIND because all the stories contain 5 sentences and hence it is easy to read for the judges. The orders of the stories are easier to judge as compared to the orders of scientific abstracts like NSF, NIPS and AAN as they require the judges to have an informed background.

Model	PMR	Acc	Tau	Rouge-S	LCS	PMR	Acc	Tau	Rouge-S	LCS
NIPS abstracts					SIND captions					
AON	16.25	50.50	0.67	80.97	74.38	13.04	45.35	0.48	73.76	72.15
B-AON	19.90	55.23	0.73	83.65	76.29	14.30	47.73	0.52	75.77	73.48
L-TSort	12.19	43.08	0.64	80.08	71.11	10.15	42.83	0.47	73.59	71.19
B-TSort	32.59	61.48	0.81	87.97	83.45	20.32	52.23	0.60	78.44	77.21

TABLE 5.2: Results on five automatic evaluation metrics for NIPS and SIND datasets.

Model	PMR	Acc	Tau	Rouge-S	LCS	PMR	Acc	Tau	Rouge-S	LCS
NSF abstracts					AAN abstracts					
AON	13.18	38.28	0.53	69.24	61.37	36.62	56.22	0.70	81.52	79.06
B-TSort	10.44	35.21	0.66	69.61	68.50	50.76	69.22	0.83	87.76	85.92

TABLE 5.3: Results on five evaluation metrics for NSF and AAN datasets.

B-TSort vs B-AON			B-TSort vs Gold			B-AON vs Gold		
B-TSort	No Pref	B-AON	B-TSort	No Pref	Gold	B-AON	No Pref	Gold
41.00%	28.00%	31.00%	26.00%	20.00%	54.00%	24.00%	22.00%	54.00%

TABLE 5.4: Human Evaluation Results on B-TSort vs AON (top), B-TSort vs Gold (middle) and AON vs Gold (bottom).

5.3 Results

Table 5.2 shows the results of the automated metrics for the NIPS and SIND datasets². It shows that AON³ model gains on all metrics when the sentence embeddings are switched to BERT. The L-TSort model which does not utilize BERT embeddings comes close to AON performance on Rouge-S and Tau metrics. This demonstrates that the simple L-TSort method is as accurate as AON in predicting relative positions but not the absolute positions (PMS and Acc metric). Table 5.2 shows that my method B-TSort does not perform better only due to BERT embeddings but also due to the experiment design. Note that BERT has been trained with the Next Sentence Prediction objective and not the sentence ordering objective like ALBERT (Lan et al., 2019). I believe that framing this task as a constraint solving task will further benefit from pre-trained language model like ALBERT. Table 5.3 shows results for the NSF and AAN datasets and the B-TSort model performs better than the AON model on all metrics.

Table 5.4 shows results for the three human evaluation studies on the SIND dataset. It shows that human judges prefer B-TSort orders 10% more number of times than the AON orders. The reference order may not be the only correct ordering of the story. The variability in the orders

²I fine-tune BERT which is memory intensive. Hence, I show the results of B-AON only on these two datasets as they need 2 transformer layers for paragraph encoder (Cui et al., 2018)

³I use the code provided by the authors to train the AON and B-AON model. The numbers reported in Table 5.2 and 5.3 are my runs of the model. Hence, they differ from the numbers reported in the paper (Cui et al., 2018).

Model	Win=1	Win=2	Win=3	% Miss	Win=1	Win=2	Win=3	% Miss
NIPS					SIND			
B-AON	81.81	92.44	96.50	3.48	78.39	92.79	98.43	0.00
B-TSort	87.59	95.59	98.11	0.00	82.67	95.01	99.09	0.00
NSF					AAN			
AON	50.58	63.87	72.96	5.85	82.65	92.25	96.73	0.84
B-TSort	61.41	75.52	83.87	0.00	90.56	96.78	98.71	0.00

TABLE 5.5: Displacement Analysis for all the datasets.

produced by B-TSort and AON is not very high and hence in comparison with Gold orders, we don't see much difference in human preferences.

The low scores of AON could be due to the fact that it has to decode the entire sequence of the order. The search space for decoding is very high (in the order of $v_i!$). Since my framework, breaks the problem to a pairwise constraint problem, the search space for my model is in the order of v_i^2 .

Discussion: I perform a few additional experiments to determine the displacement of sentences in the predicted orders by B-TSort model due to lack of direct global structure, scalability of the model for documents containing more than ten sentences, and an understanding of quality of the human judgements.

To understand the displacement of sentences in the predicted orders, I calculate the percentage of sentences whose predicted location is within 1, 2 or 3 positions (in either direction) from its original location. I observed that B-TSort consistently performs better on all datasets for all three window sizes as shown in Table 5.5. Observe that as window size reduces, the difference between B-TSort and B-AON percentages increases. This implies that displacement of sentences is higher in B-AON despite taking the whole document into account.

I additionally perform a comparison of models on documents containing more than 10 sentences and the results are shown in Table 5.6. B-TSort consistently performs better on all the metrics. SIND dataset is omitted in these experiments as the maximum number of sentences in the story is five for all the stories in the dataset. Note that the AON model generates the order and hence need not generate positions for all the sentences in the input. I calculate the percentage of mismatches between the length of the input document and the generated order. For NSF dataset, the overall mismatch is 3.48%, while the mismatch for documents with more than 10 sentences is 11.60% as shown in Table 5.6. This problem does not arise in my design of the task as it does not have to stochastically generate orders.

To better understand the choices of human judges, I observe the average length of stories calculated in number of tokens. I discover that the average length of the stories is 86 for B-TSort which is much higher as compared to B-AON with average length of 65. The average length of stories is 47 when 'No Preference' option is chosen for B-TSort vs B-AON. This means

Model	PMR	Acc	Tau	Rouge-S	LCS	%Mismatch
NIPS abstracts						
B-AON	0.0	29.18	0.51	74.64	63.81	33.33
B-TSort	0.0	39.43	0.74	83.26	71.68	0.00
NSF abstracts						
AON	2.12	21.42	0.41	67.45	55.47	11.60
B-TSort	0.67	28.57	0.64	68.46	64.86	0.00
AAN abstracts						
AON	0.0	22.70	0.40	68.90	56.19	5.17
B-TSort	0.0	36.86	0.69	78.52	72.01	0.00

TABLE 5.6: Analysis on NIPS, NSF and AAN datasets on documents longer than 10 sentences.

that B-TSort is better according to human judges for longer stories. Similarly for B-TSort vs Gold experiment, the human judges were confused with longer stories, reiterating that B-TSort performs well with long stories. ⁴

5.4 Related Work

Sentence ordering is the task of arranging sentences into an order which maximizes the coherence of the text (Barzilay and Lapata, 2008). This is important in applications where we have to determine the sequence of pre-selected set of information to be presented. This task has been well-studied in the community due to its significance in down stream applications such as ordering of: concepts in concept-to-text generation (Konstas and Lapata, 2012), information from each document in multi-document summarization (Barzilay and Elhadad, 2002; Nallapati et al., 2017), events in storytelling (Fan et al., 2019; Hu et al., 2019), cooking steps in recipe generation (Chandu et al., 2019a), and positioning of new information in existing summaries for update summarization (Prabhumoye et al., 2019). Student essays are evaluated based on how coherent and well structured they are. Hence, automated essay scoring (Burstein et al., 2010; Miltasakaki and Kukich, 2004) can use this task to improve the efficiency of their systems.

Early work on coherence modeling and sentence ordering task uses probabilistic transition model based on vectors of linguistic features (Lapata, 2003), content model which represents topics as states in an HMM (Barzilay and Lee, 2004), and entity based approach (Barzilay and Lapata, 2008). Recent work uses neural approaches to model coherence and to solve sentence ordering task. Li and Hovy (2014) introduced a neural model based on distributional sentence representations using recurrent or recursive neural networks and avoided the need of feature engineering for this task. In (Li and Jurafsky, 2017), they extend it to domain independent neural models for coherence and they introduce new latent variable Markovian generative models to capture sentence dependencies. These models used windows of sentences as context

⁴Appendix C details the hyper-parameters used for both the models and presents examples of the orders predicted for SIND and NIPS datasets by the two models.

to predict sentence pair orderings. [Gong et al. \(2016\)](#) proposed end-to-end neural architecture for sentence ordering task which uses pointer networks to utilize the contextual information in the entire piece of text.

Recently hierarchical architectures have been proposed for this task. In ([Logeswaran et al., 2018b](#)), the model uses two levels of LSTMs to first get the encoding of the sentence and then get the encoding of the entire paragraph. [Cui et al. \(2018\)](#) use a transformer network for the paragraph encoder to allow for reliable paragraph encoding. Prior work ([Logeswaran et al., 2018b](#); [Cui et al., 2018](#)) has treated this task as a sequence prediction task where the order of the sentences is predicted as a sequence. The decoder is initialized by the document representation and it outputs the index of sentences in sequential order. Only in ([Chen et al., 2016](#)), this task is framed as a ranking problem. In this work, a pairwise score is calculated between two sentences and then the final score for an order is obtained by summing over all the scores between pairs of sentences. The order which has the maximum score is given as output. Instead of considering all possible permutations of a given order, it uses beam-search strategy to find a sub-optimal order.

Most of the recent work ([Gong et al., 2016](#); [Logeswaran et al., 2018b](#); [Cui et al., 2018](#)) tries to leverage the contextual information but has the limitation of predicting the entire sequence of the order. This has the limitation that the prediction at the current time step is dependent on the prediction of the previous time step. Another limitation of the prior work is the availability of good sentence representations that can help in determining the relative order between two sentences.

Chapter 6

Ethical Considerations

Due to the sheer global reach of machine learning and NLP applications, they are empowered to impact societies (Hovy and Spruit, 2016) - potentially for the worse. Potential harms include exclusion of communities due to demographic bias, overgeneralization of model predictions to amplify bias or prejudice, and overstepping privacy concerns in the pursuit of data and quantification (Mieskes, 2017).

Many researchers are trying to make sense of these topics. Crawford (2017) give us theory to work from, presenting *allocational harm* and *representational harm*; Lewis et al. (2017) examines the role of government regulation on accountability in ethics; Smiley et al. (2017) opens a discussion on ethics checklists for acceptance testing and deployment of trained models. All of these works identify potential ethical issues for NLP, and all propose best practices for data collection and research conduct. But presently there is no external accountability for which approach to ethical NLP is correct - as machine learning researchers, we are evaluating ourselves, on metrics of our own choosing. Much of the existing work on ethics in NLP is *normative* - evaluation of whether a system “does the right thing.” I argue that before the field can establish normative goals, we need to reason about *meta-normative* decisions: specifically, how do we even decide what it means to be “right”?

6.1 Principles of Ethics

I believe that it is time for a more impartial arbitration of ethics in our field, emphasizing the need for a grounding in frameworks that long predate the questions we’re faced with today. By reaching out to other fields, we keep the question of ethics at arms length from our own work, giving us a neutral playing field on which to judge ethical performance of machine learning systems. Philosophy has much to offer us; in this section I describe two competing frameworks: the *generalization principle* and the *utilitarian principle*.

Ethics under the generalization principle

*[An ethical decision-maker] must be rational in believing that the reasons for action are consistent with the assumption that **everyone** with the same reasons will take the same action.*¹

This approach is founded on the work of (Kant, 1785), which fundamentally prioritizes *intent* as the source of ethical action. To analyze this in machine learning, a trained agent \mathcal{A} is expected to take an action \mathbf{d}_i based on a given set of evidence \mathbf{E}_i , from a finite closed set of options \mathcal{D} . This simple notation can be extended to classification, regression, or reinforcement learning tasks. The generalization principle states that agent \mathcal{A} is ethical if and only if, when given two identical sets of evidence \mathbf{E}_1 and \mathbf{E}_2 with the *same* inputs, agent \mathcal{A} chooses to make same decision \mathbf{d}_1 every time. Furthermore, the principle assumes that all *other* such trained agents will *also* make those same predictions.

Here, I presume that the input representation is sufficient to make a prediction, without including any extraneous information. The reasons for an act define the *scope* of the act, or the set of necessary and sufficient conditions under which that act is generalizably moral (Hooker, 2018a). Evidence must be relevant to the decision making process, and moreso must exclude task-irrelevant evidence that might be a source of bias. By excluding such evidence, the agent is invariant to *who* is being evaluated, and instead focuses its decision solely on task-relevant evidence.

This goal cannot be met without transparent and sparsely weighted inputs that do not use more information than is necessary and task-relevant for making predictions. Practically, this definition would privilege research on interpretable, generalizable, and understandable machine learning classifiers. The burden of proof of ethics in such a framework would lie on transparency and expressiveness of inputs, and well-defined, expected behavior from architectures for processing those features. Some work on this - like that from (Hooker and Kim, 2018) - has already begun. If deontological ethics were prioritized, we would expect to see rapid improvement in parity of F_1 scores across subgroups present in our training data - an outcome targeted by practitioners like (Chouldechova, 2017) and (Corbett-Davies et al., 2017).

Ethics under the utilitarian principle:

*An action is ethical only if it is not irrational for the agent to believe that no other action results in greater expected utility.*²

In this formulation, which can be traced back to (Bentham, 1789), an algorithmic system is expected to understand the consequences of its actions. Systems are measured by whether they

¹From (Hooker, 2018b).

²From (Hooker, 2018a).

maximize total overall welfare in their *results*. Once again an agent \mathcal{A} can be trained, which will make a decision \mathbf{d}_i for each evidence set \mathbf{E}_i . But here, you also assign a utility penalty or gain \mathbf{u}_i for each of those decisions. Rather than judge the algorithm based on whether it followed consistent rules, I instead seek to maximize *overall* gain for all N decisions that would be made by agent \mathcal{A} - morality of an agent is equal to $\sum_i^N \mathbf{u}_i$.

This is a very different worldview! Here, the burden of provable ethical decision-making no longer lands on transparency in the algorithm or consistency of a classifier over time. Instead, proof of ethical behavior rests on our ability to observe the consequences of the actions the agent takes. One could argue that consequences are hard to estimate and hence we can pick a random action. But that would be irrational. Hence, the principle judges an action by whether the agent acts according to its rational belief of maximizing the expected utility, rather than by the actual consequences. If the agent is wrong then the action turns out to be a poor choice, but nonetheless ethical because it was a rational choice.

In (Crawford, 2017), the author appeals to researchers to actively consider the subgroups that will be harmed or benefited by the automated systems. This plotting of expected consequences and their exhaustive measurement takes precedence in utilitarian ethics, de-prioritizing the interpretability or transparency of the learned model or features that govern our agent. For machine learning researchers, this would mean shifting the focus toward building rich and exhaustive test datasets, cross-validation protocols, and evaluation suites that mirror real-world applications to get a better measurement of impact.

From this work, we might see an initial drop in reported accuracy of our systems as we develop broader test sets that measure the utility of our systems; however, we would then expect overall accuracy on those broad test sets to be the primary measure of ethical fitness of the classifiers themselves. Subgroup-based parity metrics would fall by the wayside in favor of overall accuracy on data that mirrors the real world.

Real World Scenarios

These philosophical frameworks do not always diverge in their evaluation of models. Sometimes, models have unambiguously unethical gaps in performance. The exploration from (Tatman, 2017), for instance, shows the difference in accuracy of YouTube’s automatic captioning system across both gender and dialect with lower accuracy for women and speakers from Scotland (shown in Figure 6.3, reproduced from the original work). This study shows how this system violates the utilitarian principle by negatively impacting the utility of automatic speech recognition for women and speakers from Scotland. YouTube’s model also violates the generalization principle, by incorporating superfluous information about speakers in the representation space of the models. The authors suggest paths forward for improving those models and show that there is room to improve.

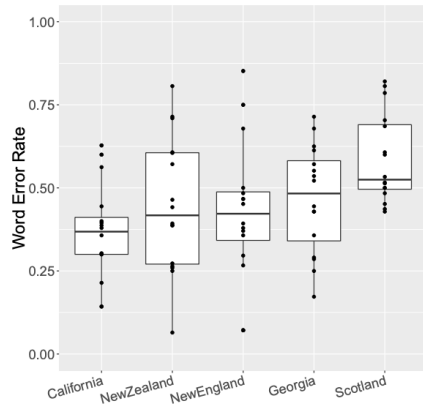


FIGURE 6.1: YouTube automatic caption word error rate by speaker's dialect region. Points indicate individual speakers.

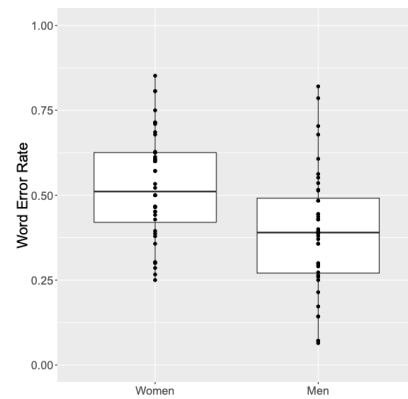


FIGURE 6.2: YouTube automatic caption word error rate by speaker's gender. Points indicate individual speakers.

FIGURE 6.3: Word Error rate plots for gender and dialect (Tatman, 2017)

But sometimes, solutions highlight differences across ethical frameworks. In (Hovy, 2015), for instance, the author shows that text classification tasks, both sentiment and topic classification, benefit from embeddings that include demographic information (age and gender). Here, the two ethical frameworks I have discussed diverge in their analysis. The generalization principle would reject this approach: age and gender shouldn't intrinsically be used as part of a demographic-agnostic topic classification task, if the number of sources of information is to be minimized. Similarly, changing the feature space depending on the author, rather than the content of the author's text, does not result in models that will make the same decision about a text independent of the identity of the author. The utilitarian principle, in contrast, aligns with the Hovy approach. A more accurate system benefits more people; incorporating information about authors improves accuracy, and so including that information at training and prediction time increases the expected utility of the model, even if different authors may receive different predictions when submitting identical texts.

For an alternate example in which the generalization principle was prioritized over utility, consider the widely-cited coreference resolution system of (Bolukbasi et al., 2016). This paper found that word embeddings used for coreference resolution were incorporating extraneous information about gender - for instance, that doctors were more likely to be men, while nurses were more likely to be women. This and similar work in "debiasing" word embeddings follows the generalization principle, arguing that removing information from the embedding space is ethically the correct action, even at the expense of model accuracy. This work finds that it can minimize the drop in expected utility, reducing F1 scores by less than 2.0 while removing stereotypes from their model. However, in a fully utilitarian ethical framework, even this drop would be unjustifiable if the model simply reflected the state of the world, and removing information led to reduced performance.

6.2 Broader impact of Controllable Text Generation

I propose to analyze the broader impact of controllable text generation on the way humans interact with technology and how it has the capacity to change human attitudes and beliefs. [Kaufman and Libby \(2012\)](#) explore the concept of experience-taking in changing human behavior and beliefs. Experience-taking is defined as the imaginative process of spontaneously assuming the identity of a character in a narrative and simulating the character's thoughts, emotions, behaviors, goals and traits as if they were one's own. When experience-taking occurs, readers adopt the character's mindset and perspective as the story progresses rather than orienting themselves as an observer or evaluator of the character. Six studies in this work ([Kaufman and Libby, 2012](#)) investigated the features of narratives that cause individuals to engage in experience-taking without instruction. Additionally, they investigated how the merger between self and other during experience-taking produces changes in self-judgments, attitudes, and behavior that align with the character's. These studies find that greater ability of a narrative to evoke experience-taking increases the ability of a reader to simulate the subjective experience of a character which in turn increases the potential that story has to change the reader's self-concept, attitudes, and behavior. The study found that a first-person narrative depicting an ingroup character elicited the highest levels of experience-taking and produced the greatest change in participants' behavior, compared with versions of the narrative written in 3rd-person voice and/or depicting an outgroup protagonist. The studies demonstrated that whereas revealing a character's outgroup membership as a homosexual or African American early in a narrative inhibited experience-taking, delaying the revelation of the character's outgroup identity until later in the story produced higher levels of experience-taking, lower levels of stereotype application in participants' evaluation of the character, and more favorable attitudes toward the character's group. I propose to open new directions for controllable text generation using these studies as firm basis to change human attitudes towards stereotypes on minority groups. For example, controllable text generation could be used to generate powerful narratives in first person which encourage experience-taking and reveal the identity of the group much later in the interaction. Similarly, ([Seering et al., 2019](#)) explores a chatbot's social role and how it can be used to maintain moderate growing online communities. This work identifies seven categories of chatbots for this role. I propose to extend this work to more personas of chatbots which can further be used to human perspectives on minority groups.

6.3 Proposed Work

In this thesis, I propose to analyze two different dimensions of using controllable text generation for ethical tasks.

Generating balanced datasets: We know that downstream tasks are influenced by the demographic skew of training sets like the sentiment analysis task is affected by the gender

confound (Hovy et al., 2015) and the part of speech (POS) tagging task is affected by the age confound (Hovy and Søgaard, 2015). By building a generation engine that can preserve content while controlling for style, we can now produce demographically balanced datasets for these NLP tasks. I first plan to analyze the generated outputs of current style transfer models for fidelity of transfer attributes. In particular I will analyze the sentiment transfer generations for the back-translation (**BST**) model proposed in (Prabhumoye et al., 2018) and the cross-aligned auto-encoder (**CAE**) model proposed in (Shen et al., 2017). I want to check if the distribution of the gender classes remains consistent when sentiment modification task is performed. I then propose to generate demographically balanced dataset for the sentiment classification task and analyze the usefulness of the generated data. Such an analysis ties back to the *generalizability principle* described in §6.1 according to which a style transfer system should be generalizable and not perform better for certain classes.

Dinan et al. (2019) also face the problem of gender imbalanced data for the dialogue generation task. Various automatic methods and data augmentation as well as data collection methods are explored in this work to resolve this issue. I would like to use style transfer and analyze the practicality of this automatic technique in this scenario. I would like to see how close the automatic method gets to the techniques proposed in this work.

Appendix A

Appendix A

This appendix details the hyper-parameters of the models described in Chapter 3 and presents examples of the generated sentences for each of the three tasks for the BST and CAE models. It also presents additional experiments with auto-encoders.

Hyperparameter settings: In all the experiments, the generator and the encoders are a two-layer bidirectional LSTM with an input size of 300 and the hidden dimension of 500. The generator samples a sentence of maximum length 50. All the generators use global attention vectors of size 500. These are especially used during the test time to replace the ‘unk’ token. The CNN classifier is trained with 100 filters of size 5, with max-pooling. The input to CNN is of size 302: the 300-dimensional word embedding plus two bits for membership of the word in our style lexicons. Balancing parameter λ_c is set to 15. For sentiment task, we have used settings provided in (Shen et al., 2017).

In Tables A.1, A.2, and A.3 we present the style transfer accuracy results broken-down to style categories. We denote the Cross-aligned Auto-Encoder model as CAE and our model as Back-translation for Style Transfer (BST).

Model	Style transfer	Acc	Style transfer	Acc
CAE	male \rightarrow female	64.75	female \rightarrow male	56.05
BST	male \rightarrow female	54.59	female \rightarrow male	59.49

TABLE A.1: Gender transfer accuracy.

Model	Style transfer	Acc	Style transfer	Acc
CAE	republican \rightarrow democratic	65.44	democratic \rightarrow republican	86.20
BST	republican \rightarrow democratic	80.55	democratic \rightarrow republican	95.47

TABLE A.2: Political slant transfer accuracy.

Model	Style transfer	Acc	Style transfer	Acc
CAE	negative \rightarrow positive	81.63	positive \rightarrow negative	79.65
BST	negative \rightarrow positive	95.68	positive \rightarrow negative	81.65

TABLE A.3: Sentiment modification accuracy.

In Table A.4, we detail the accuracy of the gender classifier on generated style-transferred sentences by an auto-encoder; Table A.5 shows the accuracy of transfer of political slant. The experiments are setup as described in Section 5.1. We denote the Auto-Encoder as (AE) and our model as Back-translation for Style Transfer (BST).

Model	Style transfer	Acc	Style transfer	Acc
AE	male \rightarrow female	41.48	female \rightarrow male	41.88
BST	male \rightarrow female	54.59	female \rightarrow male	59.49

TABLE A.4: Gender transfer accuracy for Auto-encoder.

Model	Style transfer	Acc	Style transfer	Acc
AE	republican \rightarrow democratic	60.76	democratic \rightarrow republican	64.05
BST	republican \rightarrow democratic	80.55	democratic \rightarrow republican	95.47

TABLE A.5: Political slant transfer accuracy for Auto-encoder.

To evaluate the preservation of meaning by the Auto-Encoder, the experiments were setup as described in Section 5.2. We conducted four tests, each of 20 random samples for each type of style transfer. Note that we did not ask about appropriateness of the style transfer in this test, or fluency of outputs, only about meaning preservation. We show the results of human evaluation in Table A.6

Style transfer	=	AE	BST
male \rightarrow female	43.3	13.4	43.3
female \rightarrow male	45.0	10.0	45.0
republican \rightarrow democratic	43.3	3.4	53.3
democratic \rightarrow republican	55.00	11.7	33.3

TABLE A.6: Human preference for meaning preservation in percentages.

Examples of the original and style-transferred sentences generated by the baseline and our model are shown in Tables A.7, tab:polit-style-samples and A.9.

Input Sentence	CAE	BST
male → female		
my wife ordered country fried steak and eggs.	i got ta get the chicken breast.	my husband ordered the chicken salad and the fries.
great place to visit and maybe find that one rare item you just have never seen or can not find anywhere else.	we couldn't go back and i would be able to get me to get me.	great place to go back and try a lot of which you've never had to try or could not have been able to get some of the best.
the place is small but cosy and very clean	the staff and the place is very nice.	the place is great but very clean and very friendly.
female → male		
save yourself the huge headaches.	the sauces are excellent.	you are going to be disappointed.
would i discourage someone else from going?	i believe i would be back?	i wouldn't go back!
my husband ordered the salad and the dressing - lrb - blue cheese - rrb - was watered down.	the sauces - lrb - - rrb - - rrb - and - rrb -.	my wife ordered the mac-n- cheese and the salad - lrb - \$ 00 minutes - rrb - was cooked.

TABLE A.7: Gender style transfer examples. In addition to better preserving meaning, sentences generated by the BST model are generally grammatically better structured.

Input Sentence	CAE	BST
republican → democratic		
i will continue praying for you and the decisions made by our government!	i am proud of you and your vote for us!	i will continue to fight for you and the rest of our democracy!
tom, i wish u would bring change.	i agree, senator warren and could be.	brian, i am proud to have you representing me.
all talk and no action-why dont you have some guts like breitbart	and then we will be praying for them and i am proud of this position and i am proud of	keep up and don't know, you have a lot of respect as breitbart
democratic → republican		
as a hoosier, i thank you, rep. visclosky.	a lot , i am proud of you <unk>.	as a hoosier, i'm praying for you sir.
thank you for standing up for justice and against bigotry-racism, homophobia, sexism , misogyny, religious and xenophobia.	do you for standing up for highly and in bigotry-racism, programming, cut, granddaughters, unprecedented and excludes.	thanks for standing up for the constitution and get rid of obamacare, homophobie, cut, and actuality.
thank you for all you are doing for us, attorney general harris!	thank you for standing up for us and i am proud of us!	thanks lawmaker for all you do for us, senator scott!

TABLE A.8: Political slant style transfer examples. In addition to better preserving meaning, sentences generated by the BST model are generally grammatically better structured.

Input Sentence	CAE	BST
negative → positive		
crap fries, hard hamburger buns, burger tasted like crap!	good selection, fresh food, like like like!	empathy, the best food, but it was very nice!
the people behind the counter were not friendly whatsoever.	the people who the staff were friendly.	the people here are really good.
this place is bad news!	this place is great!	this place is amazing!
positive → negative		
the food is excellent and the service is exceptional!	the food is the food and the service is terrible.	the food is horrible and the service is terrible.
great as always, i love there food.	horrible as, i really don't eat here.	really disappointed, i couldn't be back.
i would recommend a visit here.	i would not recommend a dinner here.	i will not recommend this place.

TABLE A.9: Sentiment style transfer examples. In addition to better preserving meaning, sentences generated by the BST model are generally grammatically better structured.

Appendix B

Appendix B

This appendix details the hyper-parameters of the models described in Chapter 4 for both the tasks. It presents examples of the generated sentences for the Wikipedia edit generation task and examples of human dialogues collected for the grounded dialogue response generation task.

Wikipedia Edit Generation

Hyper-parameter settings: For all our experiments with generative models, I have used bidirectional encoder, 2 layers in encoder and decoder, RNN size of 128, word vector size of 100. I have used sentencepiece toolkit¹ to use byte-pair-encoding (BPE) with a vocabulary size of 32k. I used stochastic gradient descent optimizer and the stopping criterion was perplexity on the validation set. I filtered our dataset to contain instances which have length of the document between 50 and 2000 tokens, length of the curated text between 20 and 500 tokens and the length of the update sentence between 5 and 200 tokens.

Dialogue Response Generation

Experimental Setup: For both COG and CRG model, I use a two-layer bidirectional LSTM as the encoder and a LSTM as the decoder. The dropout rate of the LSTM output is set to be 0.3. The size of hidden units for both LSTMs is 300. I set the word embedding size to be 100, since the size of vocabulary is relatively small². The models are trained with adam (Kingma and Ba, 2014) optimizer with learning rate 0.001 until they converge on the validation set for the perplexity criteria. I use beam search with size 5 for response generation. I use all the data (i.e all the conversations regardless of the rating and scenario) for training and testing. The proportion of train/validation/test split is 0.8/0.05/0.15.

¹<https://github.com/google/sentencepiece>

²The total number of tokens is 46000, and we limit the vocabulary to be 10000 tokens.

Movie lists: Here is a list of movie that were selected as topics of conversation.

- Batman Begins
- Bruce Almighty
- Batman v Superman: Dawn of Justice
- Catch me if you can
- Despicable me (2010)
- Dunkirk
- Frozen (2013)
- Home Alone
- How to Train Your Dragon (2010)
- The Imitation Game
- Iron Man (2008)
- Jaws
- John Wick (2014)
- La La Land
- Maleficent
- Mean Girls
- Monsters University
- Real Steel
- The Avengers (2012)
- The Blind Side
- The Great Gatsby (2013)
- The Inception
- The Notebook
- The Post
- The Shape of Water
- The Social Network
- The Wolf of Wall Street
- Toy Story
- Wonder Woman
- Zootopia

Instructions given to the workers

Scenario 1: users with document

- The user you are pairing does not have the document you hold. Please read the document first.
- Tell the other user what the movie is, and try to persuade the other user to watch/not to watch the movie using the information in the document.
- You should try to discuss the new paragraph when the document has changed.
- You will have 3 turns of conversation with your partner on each of the documents.
- You will be given 4 documents each containing a short paragraph. The new paragraph might show just beneath the previous document.

- The next document will be loaded automatically after you finish 3 turns discussing the current document.
- You cannot use information you personally know that is not included there. You can use any information given in the document in the conversation.

Scenario 1: users without document

- The other user will read a document about a movie.
- If you are not told the name of the movie, try to ask the movie name.
- After you are told the name of the movie, pretend you are interested in watching the movie, and try to gather all the information you need to make a decision whether to watch the movie in the end.
- You don't have to tell the other user your decision in the end, but please share your mind at the feedback page.

Scenario 2: both users with document

- The user you pair with has the same set of documents as yours. Please read the document first
- Imagine you just watched this movie. Discuss the content in the document with the other user, and show whether you like/dislike the movie.
- You should try to discuss the new paragraph when the document has changed.
- You will have 3 turns of conversation with your partner on each of the documents.
- You will be given 4 documents each containing a short paragraph. The new paragraph might show just beneath the previous document
- The next document will be loaded automatically after you finish 3 turns discussing the current document.
- You cannot use information you personally know that is not included there. You can use any information given in the document in the conversation.

Post conversation survey questions

For users with document

Choose any:

- The document is understandable.
- The other user is actively responding to me.
- The conversation goes smoothly.

Choose one of the following:

- I have watched the movie before.
- I have not watched the movie before.

For users without document

Choose any:

- The document is understandable.
- The other user is actively responding to me.
- The conversation goes smoothly.

Choose one of the following:

- I will watch the movie after the other user's introduction.
- I will not watch the movie after the other user's introduction.

Conversation Example 1

This is an example of conversation which follows *Scenario 1* where *user2* has access to sections. Tables [B.1](#), [B.2](#), [B.3](#) and [B.4](#) shows the conversation corresponding to each of the four sections of the document.

Conversation Example 2

This is an example of conversation which follows *Scenario 2* where both users have access to sections. Tables [B.5](#), [B.6](#), [B.7](#) and [B.8](#) shows the conversation corresponding to each of the four sections of the document.

Section 1	
Name	The inception
Year	2009
Director	Christopher Nolan
Genre	scientific
Cast	Leonardo DiCaprio as Dom Cobb, a professional thief who specializes in conning secrets from his victims by infiltrating their dreams. Joseph Gordon-Levitt as Arthur, Cobb's partner who manages and researches the missions. Ellen Page as Ariadne, a graduate student of architecture who is recruited to construct the various dreamscapes, which are described as mazes. Tom Hardy as Eames, a sharp-tongued associate of Cobb.
Critical Response	wildly ingenious chess game, the result is a knockout. DiCaprio, who has never been better as the tortured hero, draws you in with a love story that will appeal even to non-sci-fi fans. I found myself wishing Inception were weirder, further out the film is Nolan's labyrinth all the way, and it's gratifying to experience a summer movie with large visual ambitions and with nothing more or less on its mind than (as Shakespeare said) a dream that hath no bottom. Have no idea what so many people are raving about. It's as if someone went into their heads while they were sleeping and planted the idea that Inception is a visionary masterpiece and hold on Whoa! I think I get it. The movie is a metaphor for the power of delusional hype a metaphor for itself.
Introduction	Dominick Cobb and Arthur are extractors, who perform corporate espionage using an experimental military technology to infiltrate the subconscious of their targets and extract valuable information through a shared dream world. Their latest target, Japanese businessman Saito, reveals that he arranged their mission himself to test Cobb for a seemingly impossible job: planting an idea in a person's subconscious, or inception. To break up the energy conglomerate of ailing competitor Maurice Fischer, Saito wants Cobb to convince Fischer's son and heir, Robert, to dissolve his father's company.
Rating	Rotten Tomatoes: 86% and average: 8.1/10; IMDB: 8.8/10
Conversation	
user2:	Hey have you seen the inception?
user1:	No, I have not but have heard of it. What is it about
user2:	It's about extractors that perform experiments using military technology on people to retrieve info about their targets.
user1:	Sounds interesting do you know which actors are in it?
user2:	I haven't watched it either or seen a preview. Bu5 it's scifi so it might be good. Ugh Leonardo DiCaprio is the main character
user2:	He plays as Don Cobb
user1:	Oh okay, yeah I'm not a big scifi fan but there are a few movies I still enjoy in that genre.
user1:	Is it a long movie?
user2:	Doesn't say how long it is.
user2:	The Rotten Tomatoes score is 86%

TABLE B.1: Utterances that corresponds to section 1 of the document in the example conversation 1.

Section 2	
Scene 1	When the elder Fischer dies in Sydney, Robert Fischer accompanies the body on a ten-hour flight back to Los Angeles, which the team (including Saito, who wants to verify their success) uses as an opportunity to sedate and take Fischer into a shared dream. At each dream level, the person generating the dream stays behind to set up a 'kick' that will be used to awaken the other sleeping team members from the deeper dream level; to be successful, these kicks must occur simultaneously at each dream level, a fact complicated due to the nature of time which flows much faster in each successive level. The first level is Yusuf's dream of a rainy Los Angeles. The team abducts Fischer, but they are attacked by armed projections from Fischer's subconscious, which has been specifically trained to defend him against such intruders. The team takes Fischer and a wounded Saito to a warehouse, where Cobb reveals that while dying in the dream would normally wake Saito up, the powerful sedatives needed to stabilize the multi-level dream will instead send a dying dreamer into 'limbo', a world of infinite subconscious from which escape is extremely difficult, if not almost impossible, and a dreamer risks forgetting they are in a dream. Despite these setbacks, the team continues with the mission.
Conversation	
user1:	Wow, that's impressive. I like to look at Rotten Tomatoes when debating whether or not to see a movie. Do you know the director?
user2:	Something about Dom Cobb infiltrates peoples dreams in a dream world.
user2:	The director is Christopher nolan
user2:	Heard of him?
user2:	Wow I thought this was recent but it came out in 2009.
user1:	He directed The Dark Knight which I enjoy. Yeah, I know it's been out awhile but 2009 does seem to be a while back now. Time flies.
user1:	Do you know if it won any awards?
user1:	or how much it made at the box office?
user2:	Oh wow I loved the dark night movies. And it doesn't say if it's won awards or how much at box office.
user2:	A critic did say it could be "weirder"

TABLE B.2: Utterances that corresponds to section 2 of the document in the example conversation 1.

Section 3	
Scene 2	Cobb reveals to Ariadne that he and Mal went to Limbo while experimenting with the dream-sharing technology. Sedated for a few hours of real time, they spent fifty years in a dream constructing a world from their shared memories. When Mal refused to return to reality, Cobb used a rudimentary form of inception by reactivating her totem (an object dreamers use to distinguish dreams from reality) and reminding her subconscious that their world was not real. However, when she woke up, Mal still believed that she was dreaming. In an attempt to 'wake up' for real, Mal committed suicide and framed Cobb for her death to force him to do the same. Facing a murder charge, Cobb fled the U.S., leaving his children in the care of Professor Miles.
Conversation	
user1:	The concept seems interesting and it has a good lead actor as well as director and reviews. I think it must be good. The plot does seem weird, that's for sure.
user2:	Tom Hardy is in the movie as the character Earnes. And yeah the plot is a bit strange.
user2:	I might watch this movie now.
user1:	I think I may as well. I can't say I've heard of Tom Hardy however. Is there any other supporting actors?
user2:	Oh Earnes is a sharp tongue associate of Cobb.
user2:	Ellen Page
user1:	Oh, cool. I am familiar with her. She's in a number of good movies and is great.
user2:	She plays Ariadne , she is a graduate student that constructs the dreamscapes, they're like mazes.

TABLE B.3: Utterances that corresponds to section 3 of the document in the example conversation 1.

Section 4	
Scene 3	Through his confession, Cobb makes peace with his guilt over Mal's death. Ariadne kills Mal's projection and wakes Fischer up with a kick. Revived at the mountain hospital, Fischer enters a safe room to discover and accept the planted idea: a projection of his dying father telling him to be his own man. While Cobb remains in Limbo to search for Saito, the other team members ride the synchronized kicks back to reality. Cobb eventually finds an aged Saito in Limbo and reminds him of their agreement. The dreamers all awake on the plane and Saito makes a phone call. Upon arrival at Los Angeles Airport, Cobb passes the U.S. immigration checkpoint and Professor Miles accompanies him to his home. Using his totem a spinning top that spins indefinitely in a dream world but falls over in reality Cobb conducts a test to prove that he is indeed in the real world, but he ignores its result and instead joins his children in the garden.
Conversation	
user1:	Hmm interesting. Do you know if it's an action movie or mostly just scifi?
user2:	Says scientific
user1:	Certainly seems unique. Do you know if it is based off a book or a previous work?
user2:	Something about at the end he has trouble determining which is reality and which is a dream. It doesn't say it's based off anything.
user1:	Sounds like it might be suspense/thriller as well as scifi which is cool. It seems pretty confusing but enticing. Makes me want to see it to try and figure it all out.
user2:	Yeah its like its got a bit of mystery too. Trying to figure out what's real and what's not.
user1:	I can't think of any other movie or even book that has a related story either which makes it very interesting. A very original concept.
user2:	Yeah well have great day. :)

TABLE B.4: Utterances that corresponds to section 4 of the document in the example conversation 1.

Section 1	
Name	The Shape of Water
Year	2017
Director	Guillermo del Toro
Genre	Fantasy, Drama
Cast	Sally Hawkins as Elisa Esposito, a mute cleaner who works at a secret government laboratory. Michael Shannon as Colonel Richard Strickland, a corrupt military official, Richard Jenkins as Giles, Elisa's closeted neighbor and close friend who is a struggling advertising illustrator. Octavia Spencer as Zelda Delilah Fuller, Elisa's co-worker and friend who serves as her interpreter. Michael Stuhlbarg as Dimitri Mosenkov, a Soviet spy working as a scientist studying the creature, under the alias Dr. Robert Hoffstetler.
Critical Response	one of del Toro's most stunningly successful works, also a powerful vision of a creative master feeling totally, joyously free. Even as the film plunges into torment and tragedy, the core relationship between these two unlikely lovers holds us in thrall. Del Toro is a world-class film artist. There's no sense trying to analyze how he does it. The Shape of Water has tenderness uncommon to del Toro films. While The Shape of Water isn't groundbreaking, it is elegant and mesmerizing. refer Sally Hawkins' mute character as 'mentally handicapped' and for erroneously crediting actor Benicio del Toro as director.
Introduction	The Shape of Water is a 2017 American fantasy drama film directed by Guillermo del Toro and written by del Toro and Vanessa Taylor. It stars Sally Hawkins, Michael Shannon, Richard Jenkins, Doug Jones, Michael Stuhlbarg, and Octavia Spencer. Set in Baltimore in 1962, the story follows a mute custodian at a high-security government laboratory who falls in love with a captured humanoid amphibian creature.
Rating	Rotten Tomatoes: 92% and average: 8.4/10 Metacritic Score: 87/100 CinemaScore: A
Conversation	
user1:	Hi
user2:	Hi
user2:	I thought The Shape of Water was one of Del Toro's best works. What about you?
user1:	Did you like the movie?
user1:	Yes, his style really extended the story.
user2:	I agree. He has a way with fantasy elements that really helped this story be truly beautiful.
user2:	It has a very high rating on rotten tomatoes, too. I don't always expect that with movies in this genre.
user1:	Sally Hawkins acting was phenomenally expressive. Didn't feel her character was mentally handicapped.
user2:	The characterization of her as such was definitely off the mark.

TABLE B.5: Utterances that corresponds to section 1 of the document in the example conversation 2.

Section 2	
Scene 1	Elisa Esposito, who as an orphaned child, was found in a river with wounds on her neck, is mute, and communicates through sign language. She lives alone in an apartment above a cinema, and works as a cleaning-woman at a secret government laboratory in Baltimore at the height of the Cold War. Her friends are her closeted next-door neighbor Giles, a struggling advertising illustrator who shares a strong bond with her, and her co-worker Zelda, a woman who also serves as her interpreter at work. The facility receives a mysterious creature captured from a South American river by Colonel Richard Strickland, who is in charge of the project to study it. Curious about the creature, Elisa discovers it is a humanoid amphibian. She begins visiting him in secret, and the two form a close bond.
Conversation	
user1:	Might as well label Giles too.
user2:	haha. because he is closeted?
user2:	Whoever made that comment was certainly not well informed and not politically correct by any stretch.
user1:	I think Octavia Spencer should look for more roles set in the early 60s.
user2:	Do you think that the creature they find in the movie is supposed to be somehow connected to the cold war?

TABLE B.6: Utterances that corresponds to section 2 of the document in the example conversation 2.

Section 3	
Scene 2	Elisa keeps the creature in her bathtub, adding salt to the water to keep him alive. She plans to release the creature into a nearby canal when it will be opened to the ocean in several days' time. As part of his efforts to recover the creature, Strickland interrogates Elisa and Zelda, but the failure of his advances toward Elisa hampers his judgment, and he dismisses them. Back at the apartment, Giles discovers the creature devouring one of his cats, Pandora. Startled, the creature slashes Giles's arm and rushes out of the apartment. The creature gets as far as the cinema downstairs before Elisa finds him and returns him to her apartment. The creature touches Giles on his balding head and his wounded arm; the next morning, Giles discovers his hair has begun growing back and the wounds on his arm have healed. Elisa and the creature soon become romantically involved, having sex in her bathroom, which she at one point fills completely with water.
Conversation	
user1:	Actually Del Toro does an incredible job showing working people.
user2:	That's an excellent point.
user1:	Yes, the Cold War invented the Russians, I kind of thought it also represented technology in general.
user2:	That makes perfect sense.
user2:	I really like that Eliza chose to keep the creature in her bathtub.
user1:	It was interesting that neither power treated the monster well.
user1:	Yes the magical realism was truly magical ... easy to suspend disbelief.

TABLE B.7: Utterances that corresponds to section 3 of the document in the example conversation 2.

Section 4	
Scene 3	Hoyt gives Strickland an ultimatum, asking him to recover the creature within 36 hours. Meanwhile, Mosenkov is told by his handlers that he will be extracted in two days. As the planned release date approaches, the creature's health starts deteriorating. Mosenkov leaves to rendezvous with his handlers, with Strickland tailing him. At the rendezvous, Mosenkov is shot by one of his handlers, but Strickland shoots the handlers dead and then tortures Mosenkov for information. Mosenkov implicates Elisa and Zelda before dying from his wounds. Strickland then threatens Zelda in her home, causing her terrified husband to reveal that Elisa had been keeping the creature. Strickland searches Elisa's apartment and finds a calendar note revealing when and where she plans to release him. At the canal, Elisa and Giles bid farewell to the creature, but Strickland arrives and attacks them all. Strickland knocks Giles down and shoots the creature and Elisa, who both appear to die. However, the creature heals himself and slashes Strickland's throat, killing him. As police arrive on the scene with Zelda, the creature takes Elisa and jumps into the canal, where, deep under water, he heals her. When he applies his healing touch to the scars on her neck, she starts to breathe through gills. In a closing voiceover narration, Giles conveys his belief that Elisa lived 'happily ever after' with the creature.
Conversation	
user2:	Yes. I think it was beautiful that the creature essentially had healing power.
user1:	Del Toro does well with violence.
user1:	The ending was suspenseful, without being over the top.
user2:	What a powerful ending. Even though it was obviously a pure fantasy scenario, there was so much real emotion.
user2:	He does do well with violence. I've noticed that in all of his movies.
user2:	Del Toro is one of my favorite directors.
user1:	Yes, happy endings usually feel fake. This one felt great.
user2:	Totally. It felt like what should have happened, rather than just a sappy pretend ending that was forced on the viewer.
user1:	Mine too. Evidently Hollywood is starting to agree.
user2:	It took a while, but yes, finally.
user1:	It really appeared to be filmed in Baltimore. Installation looked so authentic.
user2:	Do you know where it was actually filmed?
user1:	No. Can you imagine soaking in that pool?
user2:	:)
user1:	Would make a great tourist draw.
user2:	That would be amazing! What a great idea!
user2:	Haven't we completed the amount of discussion needed yet?
user1:	Place looked like a cross between a nuclear power plant and an aquarium. I think we hit all the points mentioned.

TABLE B.8: Utterances that corresponds to section 4 of the document in the example conversation 2.

Appendix C

Appendix C

This appendix details the hyper-parameters of the models described in Chapter 5 and presents examples of the orders predicted for SIND and NIPS datasets by the B-TSort and the B-AON models.

Hyper-parameters. For AON model we use the code base provided by the authors in (Cui et al., 2018) and we maintain the hyper-parameters described in the paper. For the paragraph encoder of the B-AON models, we follow the same scheme of the AON model but for its sentence encoder we use hyper-parameters of the BERT setting. We use the pretrained BERT uncased base model with 12 layers for the B-AON and B-TSORT models. We fine-tune the BERT model in both cases. Hence, we replace the Adadelta optimizer with the BertAdam (Wolf et al., 2019) optimizer for the B-AON model. The LSTMs in the L-TSort model uses an RNN size of 512 and it uses the same vocabularies as the AON model. L-TSort is trained using stochastic gradient descent with dropout of 0.2, learning rate of 1.0 and learning decay rate of 0.5. For B-TSort and L-TSort we use accuracy on the validation set to stop training. For B-TSort and B-AON we use learning rate of $5e-5$ with adam epsilon value of $1e-8$.

Gold Order	B-TSort Order	B-AON Order
the family sits together for dinner on the first night of the annual reunion. the restaurant we chose had amazing food and everyone loved the presentation. gemma really adored the restaurants decorations and was always gazing at them. aunt harriot had a little trouble deciding what kind of wine she wanted tonight. bob had the whole family cracking up with his jokes.	the family sits together for dinner on the first night of the annual reunion. the restaurant we chose had amazing food and everyone loved the presentation. aunt harriot had a little trouble deciding what kind of wine she wanted tonight. gemma really adored the restaurants decorations and was always gazing at them. bob had the whole family cracking up with his jokes.	the family sits together for dinner on the first night of the annual reunion. aunt harriot had a little trouble deciding what kind of wine she wanted tonight. bob had the whole family cracking up with his jokes. gemma really adored the restaurants decorations and was always gazing at them. the restaurant we chose had amazing food and everyone loved the presentation.
he wanted to take a ride on his new bike. we went on a nice ride out to the lake. we really enjoyed the beautiful view from the dock. it was very peaceful watching the boats. we had such a busy day he needed a nap.	we went on a nice ride out to the lake. he wanted to take a ride on his new bike. we really enjoyed the beautiful view from the dock. it was very peaceful watching the boats. we had such a busy day he needed a nap.	we went on a nice ride out to the lake. he wanted to take a ride on his new bike. it was very peaceful watching the boats. we really enjoyed the beautiful view from the dock. we had such a busy day he needed a nap.
when we finally brought our son home from the hospital so many people were at home with us to see him. everyone wanted a chance to hold him! we were all so happy to have a new addition to the family. my parents were so proud to be grand parents! i am so happy and i love my son very much!	when we finally brought our son home from the hospital so many people were at home with us to see him. we were all so happy to have a new addition to the family. everyone wanted a chance to hold him! my parents were so proud to be grand parents! i am so happy and i love my son very much!	my parents were so proud to be grand parents! when we finally brought our son home from the hospital so many people were at home with us to see him. we were all so happy to have a new addition to the family. everyone wanted a chance to hold him! i am so happy and i love my son very much!

TABLE C.1: Examples of predicted sentence orders for B-TSort and B-AON model for SIND dataset.

Gold Order	B-TSort Order	B-AON Order
<p>we study how well one can recover sparse principal components of a data matrix using a sketch formed from a few of its elements. we show that for a wide class of optimization problems, if the sketch is close (in the spectral norm) to the original data matrix, then one can recover a near optimal solution to the optimization problem by using the sketch. in particular, we use this approach to obtain sparse principal components and show that for m data points in n dimensions, $\mathcal{O}(-2k \max m, n)$ elements gives an ϵ-additive approximation to the sparse pca problem (k is the stable rank of the data matrix). we demonstrate our algorithms extensively on image, text, biological and financial data. the results show that not only are we able to recover the sparse pcas from the incomplete data, but by using our sparse sketch, the running time drops by a factor of five or more.</p> <p>we develop a latent variable model and an efficient spectral algorithm motivated by the recent emergence of very large data sets of chromatin marks from multiple human cell types . a natural model for chromatin data in one cell type is a hidden markov model (hmm) ; we model the relationship between multiple cell types by connecting their hidden states by a fixed tree of known structure . the main challenge with learning parameters of such models is that iterative methods such as em are very slow , while naive spectral methods result in time and space complexity exponential in the number of cell types . we exploit properties of the tree structure of the hidden states to provide spectral algorithms that are more computationally efficient for current biological datasets . we provide sample complexity bounds for our algorithm and evaluate it experimentally on biological data from nine human cell types . finally , we show that beyond our specific model , some of our algorithmic ideas can be applied to other graphical models .</p>	<p>we study how well one can recover sparse principal components of a data matrix using a sketch formed from a few of its elements. we show that for a wide class of optimization problems, if the sketch is close (in the spectral norm) to the original data matrix, then one can recover a near optimal solution to the optimization problem by using the sketch. in particular, we use this approach to obtain sparse principal components and show that for m data points in n dimensions, $\mathcal{O}(-2k \max m, n)$ elements gives an ϵ-additive approximation to the sparse pca problem (k is the stable rank of the data matrix). the results show that not only are we able to recover the sparse pcas from the incomplete data, but by using our sparse sketch, the running time drops by a factor of five or more. we demonstrate our algorithms extensively on image, text, biological and financial data.</p> <p>a natural model for chromatin data in one cell type is a hidden markov model (hmm) ; we model the relationship between multiple cell types by connecting their hidden states by a fixed tree of known structure . the main challenge with learning parameters of such models is that iterative methods such as em are very slow , while naive spectral methods result in time and space complexity exponential in the number of cell types . we develop a latent variable model and an efficient spectral algorithm motivated by the recent emergence of very large data sets of chromatin marks from multiple human cell types . we exploit properties of the tree structure of the hidden states to provide spectral algorithms that are more computationally efficient for current biological datasets . we provide sample complexity bounds for our algorithm and evaluate it experimentally on biological data from nine human cell types . finally , we show that beyond our specific model , some of our algorithmic ideas can be applied to other graphical models .</p>	<p>we study how well one can recover sparse principal components of a data matrix using a sketch formed from a few of its elements. in particular, we use this approach to obtain sparse principal components and show that for m data points in n dimensions, $\mathcal{O}(-2k \max m, n)$ elements gives an ϵ-additive approximation to the sparse pca problem (k is the stable rank of the data matrix). we show that for a wide class of optimization problems, if the sketch is close (in the spectral norm) to the original data matrix, then one can recover a near optimal solution to the optimization problem by using the sketch. the results show that not only are we able to recover the sparse pcas from the incomplete data, but by using our sparse sketch, the running time drops by a factor of five or more. we demonstrate our algorithms extensively on image, text, biological and financial data.</p> <p>the main challenge with learning parameters of such models is that iterative methods such as em are very slow , while naive spectral methods result in time and space complexity exponential in the number of cell types . a natural model for chromatin data in one cell type is a hidden markov model (hmm) ; we model the relationship between multiple cell types by connecting their hidden states by a fixed tree of known structure . we develop a latent variable model and an efficient spectral algorithm motivated by the recent emergence of very large data sets of chromatin marks from multiple human cell types . we exploit properties of the tree structure of the hidden states to provide spectral algorithms that are more computationally efficient for current biological datasets . we provide sample complexity bounds for our algorithm and evaluate it experimentally on biological data from nine human cell types . finally , we show that beyond our specific model , some of our algorithmic ideas can be applied to other graphical models .</p>

TABLE C.2: Examples of predicted sentence orders for B-TSort and B-AON model for NIPS dataset.

Bibliography

- Gabor Angeli, Melvin Jose Johnson Premkumar, and Christopher D Manning. 2015. Leveraging linguistic structure for open domain information extraction. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 344–354.
- Mikel Artetxe, Gorka Labaka, Eneko Agirre, and Kyunghyun Cho. 2018. Unsupervised neural machine translation. In *6th International Conference on Learning Representations, ICLR 2018*.
- Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, ICLR 2015*.
- Vidhisha Balachandran, Artidoro Pagnoni, Jay Yoon Lee, Dheeraj Rajagopal, Jaime Carbonell, and Yulia Tsvetkov. 2020. [Structsum: Incorporating latent and explicit sentence dependencies for single document summarization](#). *ArXiv e-prints*.
- David Bamman, Jacob Eisenstein, and Tyler Schnoebelen. 2014. Gender identity and lexical variation in social media. *Journal of Sociolinguistics*, 18(2):135–160.
- Regina Barzilay and Noemie Elhadad. 2002. Inferring strategies for sentence ordering in multidocument news summarization. *Journal of Artificial Intelligence Research*.
- Regina Barzilay and Mirella Lapata. 2008. Modeling local coherence: An entity-based approach. *Computational Linguistics*, 34(1):1–34.
- Regina Barzilay and Lillian Lee. 2004. [Catching the drift: Probabilistic content models, with applications to generation and summarization](#). In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*, pages 113–120, Boston, Massachusetts, USA. Association for Computational Linguistics.
- Regina Barzilay and Kathleen R. McKeown. 2005. [Sentence fusion for multidocument news summarization](#). *Computational Linguistics*, 31(3):297–328.
- Christina L Bennett. 2005. Large scale evaluation of corpus-based synthesizers: Results and lessons from the blizzard challenge 2005. In *Ninth European Conference on Speech Communication and Technology*.

- Jeremy Bentham. 1789. *An introduction to the principles of morals and legislation*. Clarendon Press.
- Shane Bergsma and Benjamin Van Durme. 2013. Using conceptual class attributes to characterize social media users. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 710–720.
- Su Lin Blodgett, Lisa Green, and Brendan O’Connor. 2016. Demographic dialectal variation in social media: A case study of African-American English. In *Proc. EMNLP*.
- Tolga Bolukbasi, Kai-Wei Chang, James Y Zou, Venkatesh Saligrama, and Adam T Kalai. 2016. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In *Advances in neural information processing systems*, pages 4349–4357.
- Antoine Bosselut, Hannah Rashkin, Maarten Sap, Chaitanya Malaviya, Asli Celikyilmaz, and Yejin Choi. 2019. Comet: Commonsense transformers for automatic knowledge graph construction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4762–4779.
- Samuel R. Bowman, Luke Vilnis, Oriol Vinyals, Andrew Dai, Rafal Jozefowicz, and Samy Bengio. 2016. [Generating sentences from a continuous space](#). In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 10–21, Berlin, Germany. Association for Computational Linguistics.
- John D Burger, John Henderson, George Kim, and Guido Zarrella. 2011. Discriminating gender on twitter. In *Proceedings of the conference on empirical methods in natural language processing*, pages 1301–1309. Association for Computational Linguistics.
- Jill Burstein, Joel Tetreault, and Slava Andreyev. 2010. Using entity-based features to model coherence in student essays. In *Human language technologies: The 2010 annual conference of the North American chapter of the Association for Computational Linguistics*, pages 681–684.
- Jordan Carpenter, Daniel Preotiuc-Pietro, Lucie Flekova, Salvatore Giorgi, Courtney Hagan, Margaret L Kern, Anneke EK Buffone, Lyle Ungar, and Martin EP Seligman. 2016. Real men don’t say “cute” using automatic language analysis to isolate inaccurate aspects of stereotypes. *Social Psychological and Personality Science*.
- Khyathi Chandu, Eric Nyberg, and Alan W Black. 2019a. [Storyboarding of recipes: Grounded contextual generation](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6040–6046, Florence, Italy. Association for Computational Linguistics.
- Khyathi Chandu, Shrimai Prabhumoye, Ruslan Salakhutdinov, and Alan W Black. 2019b. “my way of telling a story”: Persona based grounded story generation. In *Proceedings of the Second Workshop on Storytelling*, pages 11–21.
- Xinchi Chen, Xipeng Qiu, and Xuanjing Huang. 2016. Neural sentence ordering. *arXiv preprint arXiv:1607.06952*.

- Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734.
- Alexandra Chouldechova. 2017. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *Big data*, 5(2):153–163.
- Jennifer Coates. 2015. *Women, men and language: A sociolinguistic account of gender differences in language*. Routledge.
- Sam Corbett-Davies, Emma Pierson, Avi Feller, Sharad Goel, and Aziz Huq. 2017. Algorithmic decision making and the cost of fairness. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 797–806. ACM.
- Nikolas Coupland. 2007. *Style: Language Variation and Identity*. Key Topics in Sociolinguistics. Cambridge University Press.
- Kate Crawford. 2017. The trouble with bias, 2017. URL <http://blog.revolutionanalytics.com/2017/12/the-trouble-with-bias-by-kate-crawford.html>. Invited Talk by Kate Crawford at NIPS.
- Baiyun Cui, Yingming Li, Ming Chen, and Zhongfei Zhang. 2018. Deep attentive sentence ordering network. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4340–4349.
- Cristian Danescu-Niculescu-Mizil and Lillian Lee. 2011. Chameleons in imagined conversations: A new approach to understanding coordination of linguistic style in dialogs. In *Proceedings of the Workshop on Cognitive Modeling and Computational Linguistics, ACL 2011*.
- Hoa Trang Dang and Karolina Owczarzak. 2008. Overview of the TAC 2008 update summarization task. In *TAC 2008 Workshop - Notebook papers and results*, pages 10–23.
- Michael Denkowski and Alon Lavie. 2011. Meteor 1.3: Automatic metric for reliable optimization and evaluation of machine translation systems. In *Proc. WMT*, pages 85–91.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. 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 4171–4186.
- Emily Dinan, Angela Fan, Adina Williams, Jack Urbanek, Douwe Kiela, and Jason Weston. 2019. Queens are powerful too: Mitigating gender bias in dialogue generation. *arXiv preprint arXiv:1911.03842*.
- Emily Dinan, Stephen Roller, Kurt Shuster, Angela Fan, Michael Auli, and Jason Weston. 2018. Wizard of wikipedia: Knowledge-powered conversational agents. In *International Conference on Learning Representations*.

- Li Dong, Jonathan Mallinson, Siva Reddy, and Mirella Lapata. 2017. Learning to paraphrase for question answering. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 875–886. Association for Computational Linguistics.
- Penelope Eckert. 2019. The limits of meaning: Social indexicality, variation, and the cline of interiority. *Language*, 95(4):751–776.
- Penelope Eckert and Sally McConnell-Ginet. 2003. *Language and gender*. Cambridge University Press.
- Layla El Asri, Hannes Schulz, Shikhar Sharma, Jeremie Zumer, Justin Harris, Emery Fine, Rahul Mehrotra, and Kaheer Suleman. 2017. [Frames: a corpus for adding memory to goal-oriented dialogue systems](#). In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, pages 207–219, Saarbrücken, Germany. Association for Computational Linguistics.
- Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 889–898, Melbourne, Australia.
- Angela Fan, Mike Lewis, and Yann Dauphin. 2019. Strategies for structuring story generation. *arXiv preprint arXiv:1902.01109*.
- Jessica Fidler and Yoav Goldberg. 2017. Controlling linguistic style aspects in neural language generation. In *Proceedings of the Workshop on Stylistic Variation*, pages 94–104.
- Seeger Fisher and Brian Roark. 2008. Query-focused supervised sentence ranking for update summaries. In *TAC*.
- Susan T Fiske. 1993. Controlling other people: The impact of power on stereotyping. *American psychologist*, 48(6):621.
- Lucie Flekova and Iryna Gurevych. 2013. Can we hide in the web? large scale simultaneous age and gender author profiling in social media. In *CLEF 2012 Labs and Workshop, Notebook Papers*. Citeseer.
- Zhenxin Fu, Xiaoye Tan, Nanyun Peng, Dongyan Zhao, and Rui Yan. 2017. [Style transfer in text: Exploration and evaluation](#). *CoRR*, abs/1711.06861.
- Zhenxin Fu, Xiaoye Tan, Nanyun Peng, Dongyan Zhao, and Rui Yan. 2018. Style transfer in text: Exploration and evaluation. In *Thirty-Second AAAI Conference on Artificial Intelligence*.
- Chuang Gan, Zhe Gan, Xiaodong He, Jianfeng Gao, and Li Deng. 2017. Stylenet: Generating attractive visual captions with styles. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3137–3146.
- Juri Ganitkevitch and Chris Callison-Burch. 2014. [The multilingual paraphrase database](#). In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC’14)*, pages 4276–4283, Reykjavik, Iceland. European Language Resources Association (ELRA).

- Juri Ganitkevitch, Chris Callison-Burch, Courtney Napoles, and Benjamin Van Durme. 2011. [Learning sentential paraphrases from bilingual parallel corpora for text-to-text generation](#). In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 1168–1179, Edinburgh, Scotland, UK. Association for Computational Linguistics.
- Marjan Ghazvininejad, Chris Brockett, Ming-Wei Chang, Bill Dolan, Jianfeng Gao, Wen-tau Yih, and Michel Galley. 2018. A knowledge-grounded neural conversation model. In *Thirty-Second AAAI Conference on Artificial Intelligence*.
- Sayan Ghosh, Mathieu Chollet, Eugene Laksana, Louis-Philippe Morency, and Stefan Scherer. 2017. Affect-LM: A neural language model for customizable affective text generation. In *ACL*, volume 1, pages 634–642.
- Hongyu Gong, Suma Bhat, Lingfei Wu, JinJun Xiong, and Wen-mei Hwu. 2019. [Reinforcement learning based text style transfer without parallel training corpus](#). 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 3168–3180, Minneapolis, Minnesota. Association for Computational Linguistics.
- Jingjing Gong, Xinchu Chen, Xipeng Qiu, and Xuanjing Huang. 2016. End-to-end neural sentence ordering using pointer network. *arXiv preprint arXiv:1611.04953*.
- Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Advances in Neural Information Processing Systems 27*, pages 2672–2680.
- David Graff and Christopher Cieri. 2003. English Gigaword LDC2003T05. In *Philadelphia: Linguistic Data Consortium*.
- Vrindavan Harrison, Lena Reed, Shereen Oraby, and Marilyn Walker. 2019. Maximizing stylistic control and semantic accuracy in nlg: Personality variation and discourse contrast. In *Proceedings of the 1st Workshop on Discourse Structure in Neural NLG*, pages 1–12.
- Junxian He, Xinyi Wang, Graham Neubig, and Taylor Berg-Kirkpatrick. 2020. [A probabilistic formulation of unsupervised text style transfer](#). In *International Conference on Learning Representations*.
- John Hewitt and Christopher D. Manning. 2019. [A structural probe for finding syntax in word representations](#). 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 4129–4138, Minneapolis, Minnesota. Association for Computational Linguistics.
- E.T. Higgins and G.R. Semin. 2001. [Communication and social psychology](#). In Neil J. Smelser and Paul B. Baltes, editors, *International Encyclopedia of the Social & Behavioral Sciences*, pages 2296 – 2299. Pergamon, Oxford.

- Cong Duy Vu Hoang, Trevor Cohn, and Gholamreza Haffari. 2016. Incorporating side information into recurrent neural network language models. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1250–1255.
- Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.
- Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2020. [The curious case of neural text degeneration](#). In *International Conference on Learning Representations*.
- Ari Holtzman, Jan Buys, Maxwell Forbes, Antoine Bosselut, David Golub, and Yejin Choi. 2018. Learning to write with cooperative discriminators. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1638–1649.
- John Hooker. 2018a. *Taking Ethics Seriously: Why Ethics Is an Essential Tool for the Modern Workplace*. Taylor and Francis.
- John Hooker. 2018b. Truly autonomous machines are ethical. *arXiv preprint arXiv:1812.02217*.
- John N Hooker and Tae Wan N Kim. 2018. Toward non-intuition-based machine and artificial intelligence ethics: A deontological approach based on modal logic. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, pages 130–136. ACM.
- Dirk Hovy. 2015. Demographic factors improve classification performance. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, volume 1, pages 752–762.
- Dirk Hovy, Anders Johannsen, and Anders Søgaard. 2015. User review sites as a resource for large-scale sociolinguistic studies. In *Proceedings of the 24th International Conference on World Wide Web*, pages 452–461. International World Wide Web Conferences Steering Committee.
- Dirk Hovy and Anders Søgaard. 2015. Tagging performance correlates with author age. 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 483–488.
- Dirk Hovy and Shannon L Spruit. 2016. The social impact of natural language processing. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 591–598.
- Ed Hovy. 1993. From interclausal relations to discourse structure - a long way behind, a long way ahead. In Helmut Horacek and Michael Zock, editors, *New Concepts in Natural Language Generation: Planning, Realization and Systems*. St. Martin’s Press, Inc., USA.

- Eduard Hovy. 1987. Generating natural language under pragmatic constraints. *Journal of Pragmatics*, 11(6):689–719.
- Eduard H. Hovy. 1988. [Planning coherent multisentential text](#). In *26th Annual Meeting of the Association for Computational Linguistics*, pages 163–169, Buffalo, New York, USA. Association for Computational Linguistics.
- Junjie Hu, Yu Cheng, Zhe Gan, Jingjing Liu, Jianfeng Gao, and Graham Neubig. 2019. What makes a good story? designing composite rewards for visual storytelling. *arXiv preprint arXiv:1909.05316*.
- Zhiting Hu, Zichao Yang, Xiaodan Liang, Ruslan Salakhutdinov, and Eric P Xing. 2017. Toward controlled generation of text. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1587–1596. JMLR. org.
- Ting-Hao K. Huang, Francis Ferraro, Nasrin Mostafazadeh, Ishan Misra, Jacob Devlin, Aishwarya Agrawal, Ross Girshick, Xiaodong He, Pushmeet Kohli, Dhruv Batra, et al. 2016. Visual storytelling. In *15th Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL 2016)*.
- Eric Jardine. 2016. Tor, what is it good for? political repression and the use of online anonymity-granting technologies. *New Media & Society*.
- Jianshu Ji, Qinlong Wang, Kristina Toutanova, Yongen Gong, Steven Truong, and Jianfeng Gao. 2017. A nested attention neural hybrid model for grammatical error correction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 753–762, Vancouver, Canada.
- Vineet John, Lili Mou, Hareesh Bahuleyan, and Olga Vechtomova. 2019. [Disentangled representation learning for non-parallel text style transfer](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 424–434, Florence, Italy. Association for Computational Linguistics.
- Anna Jørgensen, Dirk Hovy, and Anders Søgaard. 2015. Challenges of studying and processing dialects in social media. In *Proc. of the Workshop on Noisy User-generated Text*, pages 9–18.
- Junbo, Zhao, Y. Kim, K. Zhang, A. M. Rush, and Y. LeCun. 2017. [Adversarially Regularized Autoencoders for Generating Discrete Structures](#). *ArXiv e-prints*.
- Dongyeop Kang and Eduard Hovy. 2019. [xslue: A benchmark and analysis platform for cross-style language understanding and evaluation](#). In <https://arxiv.org>.
- Immanuel Kant. 1785. *Groundwork for the Metaphysics of Morals*. Yale University Press.
- Geoff F Kaufman and Lisa K Libby. 2012. Changing beliefs and behavior through experience-taking. *Journal of personality and social psychology*, 103(1):1.

- Shari Kendall, Deborah Tannen, et al. 1997. Gender and language in the workplace. *Gender and Discourse*. London: Sage, pages 81–105.
- Chloé Kiddon, Luke Zettlemoyer, and Yejin Choi. 2016. [Globally coherent text generation with neural checklist models](#). In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 329–339, Austin, Texas. Association for Computational Linguistics.
- Yuta Kikuchi, Graham Neubig, Ryohei Sasano, Hiroya Takamura, and Manabu Okumura. 2016. Controlling output length in neural encoder-decoders. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1328–1338, Austin, Texas.
- Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
- Diederik P Kingma and Max Welling. 2014. Auto-encoding variational bayes. In *Proc. ICLR*.
- Donald Ervin Knuth. 1998. [The art of computer programming, , Volume III, 2nd Edition](#). Addison-Wesley.
- Ioannis Konstas and Mirella Lapata. 2012. Concept-to-text generation via discriminative reranking. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 369–378. Association for Computational Linguistics.
- Ilia Kulikov, Alexander Miller, Kyunghyun Cho, and Jason Weston. 2019. [Importance of search and evaluation strategies in neural dialogue modeling](#). In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 76–87, Tokyo, Japan. Association for Computational Linguistics.
- Sachin Kumar, Shuly Wintner, Noah A. Smith, and Yulia Tsvetkov. 2019. [Topics to avoid: Demoting latent confounds in text classification](#). 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 4153–4163, Hong Kong, China. Association for Computational Linguistics.
- Robin Tolmach Lakoff and Mary Bucholtz. 2004. *Language and woman’s place: Text and commentaries*, volume 3. Oxford University Press, USA.
- Guillaume Lample, Alexis Conneau, Ludovic Denoyer, and Marc’Aurelio Ranzato. 2018. Unsupervised machine translation using monolingual corpora only. In *International Conference on Learning Representations*.
- Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations. In *International Conference on Learning Representations*.

- Mirella Lapata. 2003. Probabilistic text structuring: Experiments with sentence ordering. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, pages 545–552. Association for Computational Linguistics.
- Mirella Lapata. 2006. Automatic evaluation of information ordering: Kendall’s tau. *Computational Linguistics*, 32(4):471–484.
- Rémi Lebre, David Grangier, and Michael Auli. 2016. Neural text generation from structured data with application to the biography domain. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1203–1213. Association for Computational Linguistics.
- Dave Lewis, Joss Moorkens, and Kaniz Fatema. 2017. Integrating the management of personal data protection and open science with research ethics. In *Proceedings of the First ACL Workshop on Ethics in Natural Language Processing*, pages 60–65.
- Chen Li, Yang Liu, and Lin Zhao. 2015. Improving update summarization via supervised ILP and sentence reranking. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1317–1322.
- Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016a. A diversity-promoting objective function for neural conversation models. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 110–119, San Diego, California. Association for Computational Linguistics.
- Jiwei Li, Michel Galley, Chris Brockett, Georgios Spithourakis, Jianfeng Gao, and Bill Dolan. 2016b. A persona-based neural conversation model. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 994–1003.
- Jiwei Li and Eduard Hovy. 2014. A model of coherence based on distributed sentence representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2039–2048.
- Jiwei Li and Dan Jurafsky. 2017. Neural net models of open-domain discourse coherence. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 198–209.
- Juncen Li, Robin Jia, He He, and Percy Liang. 2018a. [Delete, retrieve, generate: a simple approach to sentiment and 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)*, pages 1865–1874, New Orleans, Louisiana. Association for Computational Linguistics.

- Wei Li, Xinyan Xiao, Yajuan Lyu, and Yuanzhuo Wang. 2018b. [Improving neural abstractive document summarization with structural regularization](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4078–4087, Brussels, Belgium. Association for Computational Linguistics.
- Xiang Lisa Li and Jason Eisner. 2019. Specializing word embeddings (for parsing) by information bottleneck. 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 2744–2754.
- Chin-Yew Lin and Eduard Hovy. 2002. Manual and automatic evaluation of summaries. In *Proceedings of the ACL-02 Workshop on Automatic Summarization-Volume 4*, pages 45–51. Association for Computational Linguistics.
- Peter J. Liu, Mohammad Saleh, Etienne Pot, Ben Goodrich, Ryan Sepassi, Lukasz Kaiser, and Noam Shazeer. 2018. Generating wikipedia by summarizing long sequences. In *International Conference on Learning Representations*.
- Yang Liu and Mirella Lapata. 2018. Learning structured text representations. *Transactions of the Association for Computational Linguistics*, 6:63–75.
- Lajanugen Logeswaran, Honglak Lee, and Samy Bengio. 2018a. Content preserving text generation with attribute controls. In *Advances in Neural Information Processing Systems*, pages 5103–5113.
- Lajanugen Logeswaran, Honglak Lee, and Dragomir Radev. 2018b. Sentence ordering and coherence modeling using recurrent neural networks. In *Thirty-Second AAAI Conference on Artificial Intelligence*.
- Minh-Thang Luong, Hieu Pham, and Christopher D Manning. 2015a. Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*.
- Minh-Thang Luong, Hieu Pham, and Christopher D. Manning. 2015b. Effective approaches to attention-based neural machine translation. In *Proc. EMNLP*.
- Aman Madaan, Amrith Setlur, Tanmay Parekh, Barnabas Póczos, Graham Neubig, Yiming Yang, Ruslan Salakhutdinov, Alan W Black, and Shrimai Prabhumoye. 2020. Politeness transfer: A tag and generate approach. In *Under review at ACL*.
- Nitin Madnani and Bonnie J Dorr. 2010. Generating phrasal and sentential paraphrases: A survey of data-driven methods. *Computational Linguistics*, 36(3):341–387.
- Jonathan Mallinson, Rico Sennrich, and Mirella Lapata. 2017. Paraphrasing revisited with neural machine translation. In *Proce. EACL*, volume 1, pages 881–893.
- William Mann and Sandra Thompson. 1988. [Rhetorical structure theory: Toward a functional theory of text organization](#). *Text*, 8:243–281.

- Daniel Marcu. 1997. From local to global coherence: A bottom-up approach to text planning. In *AAAI/IAAI*.
- T. McEnery. 2005. [Swearing in english: Bad language, purity and power from 1586 to the present](#). *Swearing in English: Bad Language, Purity and Power from 1586 to the Present*, pages 1–248.
- Hongyuan Mei, TTI UChicago, Mohit Bansal, and Matthew R Walter. 2016. What to talk about and how? selective generation using LSTMs with coarse-to-fine alignment. In *Proceedings of NAACL-HLT*, pages 720–730.
- Margot Mieskes. 2017. A quantitative study of data in the nlp community. In *Proceedings of the First ACL Workshop on Ethics in Natural Language Processing*, pages 23–29.
- Eleni Miltsakaki and Karen Kukich. 2004. Evaluation of text coherence for electronic essay scoring systems. *Natural Language Engineering*, 10(1):25–55.
- Burt L. Monroe, Michael P. Colaresi, and Kevin M. Quinn. 2008. Fightin words: Lexical feature selection and evaluation for identifying the content of political conflict. *Political Analysis*.
- Nasrin Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. 2016. A corpus and cloze evaluation for deeper understanding of commonsense stories. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 839–849, San Diego, California.
- Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. 2017. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. In *Thirty-First AAAI Conference on Artificial Intelligence*.
- 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*, pages 280–290. Association for Computational Linguistics.
- Ani Nenkova and Lucy Vanderwende. 2005. The impact of frequency on summarization. Technical report, Microsoft Research.
- Dong Nguyen, A. Seza Doğruöz, Carolyn P. Rosé, and Franciska de Jong. 2016. Computational sociolinguistics: A survey. *Computational Linguistics*, 42(3):537–593.
- Xing Niu and Marine Carpuat. 2019. Controlling neural machine translation formality with synthetic supervision. *arXiv preprint arXiv:1911.08706*.
- 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.

- Thanapon Noraset, Chen Liang, Larry Birnbaum, and Doug Downey. 2017. Definition modeling: Learning to define word embeddings in natural language. In *Thirty-First AAAI Conference on Artificial Intelligence*.
- Franz Josef Och, Daniel Gildea, Sanjeev Khudanpur, Anoop Sarkar, Kenji Yamada, Alexander Fraser, Shankar Kumar, Libin Shen, David A Smith, Katherine Eng, et al. 2004. A smorgasbord of features for statistical machine translation. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*, pages 161–168.
- 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.
- Romain Paulus, Caiming Xiong, and Richard Socher. 2018. [A deep reinforced model for abstractive summarization](#). In *International Conference on Learning Representations*.
- Claudia Peersman, Walter Daelemans, and Leona Van Vaerenbergh. 2011. Predicting age and gender in online social networks. In *Proceedings of the 3rd International Workshop on Search and Mining User-generated Contents*, pages 37–44. ACM.
- Nanyun Peng, Marjan Ghazvininejad, Jonathan May, and Kevin Knight. 2018. Towards controllable story generation. In *Proceedings of the First Workshop on Storytelling*, pages 43–49.
- Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. [Language models as knowledge bases?](#) 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 2463–2473, Hong Kong, China. Association for Computational Linguistics.
- Shrimai Prabhumoye, Chris Quirk, and Michel Galley. 2019. [Towards content transfer through grounded 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 2622–2632, Minneapolis, Minnesota. Association for Computational Linguistics.
- Shrimai Prabhumoye, Yulia Tsvetkov, Ruslan Salakhutdinov, and Alan W Black. 2018. Style transfer through back-translation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 866–876, Melbourne, Australia.
- Ella Rabinovich, Shachar Mirkin, Raj Nath Patel, Lucia Specia, and Shuly Wintner. 2016. Personalized machine translation: Preserving original author traits. In *Proc. EACL*.
- Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

- Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.
- Delip Rao, David Yarowsky, Abhishek Shreevats, and Manaswi Gupta. 2010. Classifying latent user attributes in twitter. In *Proceedings of the 2nd international workshop on Search and mining user-generated contents*, pages 37–44.
- 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.
- Sravana Reddy and Kevin Knight. 2016. [Obfuscating gender in social media writing](#). In *Proceedings of the First Workshop on NLP and Computational Social Science*, pages 17–26, Austin, Texas. Association for Computational Linguistics.
- Ehud Reiter and Robert Dale. 2000. *Building Natural Language Generation Systems*. Cambridge University Press, USA.
- Alexey Romanov, Anna Rumshisky, Anna Rogers, and David Donahue. 2019. [Adversarial decomposition of text representation](#). 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 815–825, Minneapolis, Minnesota. Association for Computational Linguistics.
- Stuart Rose, Dave Engel, Nick Cramer, and Wendy Cowley. 2010. [Automatic Keyword Extraction from Individual Documents](#).
- Rachel Rudinger, Chandler May, and Benjamin Van Durme. 2017. Social bias in elicited natural language inferences. In *Proc. of the First Workshop on Ethics in Natural Language Processing*, page 74.
- 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*, pages 379–389, Lisbon, Portugal.
- 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.
- Joseph Seering, Michal Luria, Geoff Kaufman, and Jessica Hammer. 2019. Beyond dyadic interactions: Considering chatbots as community members. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, pages 1–13.
- 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, San Diego, California. Association for Computational Linguistics.

- 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 6830–6841.
- Charese Smiley, Frank Schilder, Vassilis Plachouras, and Jochen L. Leidner. 2017. Say the right thing right: Ethics issues in natural language generation systems. In *Proceedings of the First ACL Workshop on Ethics in Natural Language Processing*, pages 103–108.
- Steven J. Spencer, Claude M. Steele, and Diane M. Quinn. 1999. Stereotype Threat and Women’s Math Performance. *Journal of Experimental Social Psychology*, 35:4–28.
- Andreas Stolcke. 2002. Srilm—an extensible language modeling toolkit. In *Seventh international conference on spoken language processing*.
- Akhilesh Sudhakar, Bhargav Upadhyay, and Arjun Maheswaran. 2019. “transforming” delete, retrieve, generate approach for controlled text style transfer. 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 3260–3270.
- 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.
- Deborah Tannen. 1991. *You just don’t understand: Women and men in conversation*. Virago London.
- Deborah Tannen. 1993. *Gender and conversational interaction*. Oxford University Press.
- Robert Endre Tarjan. 1976. [Edge-disjoint spanning trees and depth-first search](#). *Acta Informatica*, 6(2):171–185.
- Rachael Tatman. 2017. Gender and dialect bias in youtube’s automatic captions. In *Proceedings of the First ACL Workshop on Ethics in Natural Language Processing*, pages 53–59.
- Jack Urbanek, Angela Fan, Siddharth Karamcheti, Saachi Jain, Samuel Humeau, Emily Dinan, Tim Rocktäschel, Douwe Kiela, Arthur Szlam, and Jason Weston. 2019. Learning to speak and act in a fantasy text adventure game. 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 673–683.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.
- S. Verberne, LWJ Boves, NHJ Oostdijk, and PAJM Coppen. 2007. Evaluating discourse-based answer extraction for why-question answering. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2007)*, pages 735–737. Amsterdam: Association for Computing Machinery.

- Ashwin K Vijayakumar, Michael Cogswell, Ramprasaath R Selvaraju, Qing Sun, Stefan Lee, David Crandall, and Dhruv Batra. 2018. Diverse beam search for improved description of complex scenes. In *Thirty-Second AAAI Conference on Artificial Intelligence*.
- Rob Voigt, David Jurgens, Vinodkumar Prabhakaran, Dan Jurafsky, and Yulia Tsvetkov. 2018. [RtGender: A corpus for studying differential responses to gender](#). In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).
- Svitlana Volkova, Yoram Bachrach, Michael Armstrong, and Vijay Sharma. 2015. Inferring latent user properties from texts published in social media. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*.
- Ke Wang, Hang Hua, and Xiaojun Wan. 2019a. Controllable unsupervised text attribute transfer via editing entangled latent representation. In *Advances in Neural Information Processing Systems*, pages 11034–11044.
- Wenlin Wang, Zhe Gan, Hongteng Xu, Ruiyi Zhang, Guoyin Wang, Dinghan Shen, Changyou Chen, and Lawrence Carin. 2019b. [Topic-guided variational auto-encoder for 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 166–177, Minneapolis, Minnesota. Association for Computational Linguistics.
- Tsung-Hsien Wen, Milica Gasic, Nikola Mrkšić, Pei-Hao Su, David Vandyke, and Steve Young. 2015. Semantically conditioned LSTM-based natural language generation for spoken dialogue systems. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1711–1721.
- John Wieting, Jonathan Mallinson, and Kevin Gimpel. 2017. Learning paraphrastic sentence embeddings from back-translated bitext. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 274–285.
- Ronald J Williams and David Zipser. 1989. A learning algorithm for continually running fully recurrent neural networks. *Neural computation*, 1(2):270–280.
- Sam Wiseman, Stuart Shieber, and Alexander Rush. 2017. Challenges in data-to-document generation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2253–2263, Copenhagen, Denmark. Association for Computational Linguistics.
- Florian Wolf and Edward Gibson. 2006. *Coherence in natural language: data structures and applications*. MIT Press.
- Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, R’emi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface’s transformers: State-of-the-art natural language processing. *ArXiv*, abs/1910.03771.

- Peng Xu, Yanshuai Cao, and Jackie Chi Kit Cheung. 2019. [Unsupervised controllable text generation with global variation discovery and disentanglement](#). *CoRR*, abs/1905.11975.
- Hayahide Yamagishi, Shin Kanouchi, Takayuki Sato, and Mamoru Komachi. 2016. Controlling the voice of a sentence in Japanese-to-English neural machine translation. In *Proceedings of the 3rd Workshop on Asian Translation (WAT2016)*, pages 203–210, Osaka, Japan. The COLING 2016 Organizing Committee.
- Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in neural information processing systems*, pages 5754–5764.
- Zichao Yang, Zhiting Hu, Chris Dyer, Eric P Xing, and Taylor Berg-Kirkpatrick. 2018. Unsupervised text style transfer using language models as discriminators. In *Advances in Neural Information Processing Systems*, pages 7287–7298.
- Jen-Yuan Yeh and Aaron Harnly. 2006. Email thread reassembly using similarity matching. Conference on Email and Anti-Spam.
- Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston. 2018. Personalizing dialogue agents: I have a dog, do you have pets too? In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2204–2213. Association for Computational Linguistics.
- Kangyan Zhou, Shrimai Prabhumoye, and Alan W Black. 2018. A dataset for document grounded conversations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 708–713.
- Yiheng Zhou, Yulia Tsvetkov, Alan W Black, and Zhou Yu. 2020. [Augmenting non-collaborative dialog systems with explicit semantic and strategic dialog history](#). In *International Conference on Learning Representations*.