Deep Learning for Text Style Transfer: A Survey

Di Jin*
Amazon
Alexa AI
djinamzn@amazon.com

Zhijing Jin*
Max Planck Institute for
Intelligent Systems
Empirical Inference Department
and ETH Zürich Department of
Computer Science
zjin@tue.mpg.de

Zhiting Hu
UC San Diego
Halıcıoğlu Data Science Institute (HDSI)
zhh019@ucsd.edu

Olga Vechtomova
University of Waterloo
Faculty of Engineering
ovechtom@uwaterloo.ca

Rada Mihalcea
University of Michigan
EECS, College of Engineering
mihalcea@umich.edu

Text style transfer is an important task in natural language generation, which aims to control certain attributes in the generated text, such as politeness, emotion, humor, and many others. It has a long history in the field of natural language processing, and recently has re-gained significant attention thanks to the promising performance brought by deep neural models. In this article, we present a systematic survey of the research on neural text style transfer, spanning over 100 representative articles since the first neural text style transfer work in 2017. We discuss the

* Equal contribution.

Submission received: 25 April 2021; revised version received: 30 August 2021; accepted for publication: 4 December 2021.

https://doi.org/10.1162/COLI_a_00426

task formulation, existing datasets and subtasks, evaluation, as well as the rich methodologies in the presence of parallel and non-parallel data. We also provide discussions on a variety of important topics regarding the future development of this task.[1]

1. Introduction

Language is situational. Every utterance fits in a specific time, place, and scenario, conveys specific characteristics of the speaker, and typically has a well-defined intent. For example, someone who is uncertain is more likely to use tag questions (e.g., "This is true, isn't it?") than declarative sentences (e.g., "This is definitely true."). Similarly, a professional setting is more likely to include formal statements (e.g., "Please consider taking a seat.") as compared to an informal situation (e.g., "Come and sit!"). For artificial intelligence systems to accurately understand and generate language, it is necessary to model language with style/attribute,[2] which goes beyond merely verbalizing the semantics in a non-stylized way. The values of the attributes can be drawn from a wide range of choices depending on pragmatics, such as the extent of formality, politeness, simplicity, personality, emotion, partner effect (e.g., reader awareness), genre of writing (e.g., fiction or non-fiction), and so on.

The goal of TST is to automatically control the style attributes of text while preserving the content. TST has a wide range of applications, as outlined by McDonald and Pustejovsky (1985) and Hovy (1987). The style of language is crucial because it makes natural language processing more user-centered. TST has many immediate applications. For instance, one such application is intelligent bots, for which users prefer distinct and consistent persona (e.g., empathetic) instead of emotionless or inconsistent persona. Another application is the development of intelligent writing assistants; for example, non-expert writers often need to polish their writing to better fit their purpose, for example, more professional, polite, objective, humorous, or other advanced writing requirements, which may take years of experience to master. Other applications include automatic text simplification (where the target style is "simple"), debiasing online text (where the target style is "objective"), fighting against offensive language (where the target style is "non-offensive"), and so on.

To formally define TST, let us denote the target utterance as x' and the target discourse style attribute as a'. TST aims to model $p(x'|a, x)$, where x is a given text carrying a source attribute value a. Consider the previous example of text expressed by two different extents of formality:

Source sentence x:	*"Come and sit!"*	Source attribute a:	*Informal*
Target sentence x':	*"Please consider taking a seat."*	Target attribute a':	*Formal*

In this case, a TST model should be able to modify the formality and generate the formal sentence x' = *"Please consider taking a seat."* given the informal input x = *"Come and sit!"*. Note that the key difference of TST from another NLP task, style-conditioned language modeling, is that the latter is conditioned on only a style token, whereas TST

1 Our curated paper list is at `https://github.com/zhijing-jin/Text_Style_Transfer_Survey`.

2 Note that we interchangeably use the terms *style* and *attribute* in this survey. *Attribute* is a broader terminology that can include content preferences, e.g., sentiment, topic, and so on. This survey uses *style* in the same broad way, following the common practice in recent papers (see Section 2.1).

takes as input both the target style attribute a' and a source sentence x that constrains the content.

Crucial to the definition of style transfer is the distinction of "style" and "content," for which there are two common practices. The first one is by linguistic definition, where non-functional linguistic features are classified into the style (e.g., formality), and the semantics are classified into the content. In contrast, the second practice is data-driven—given two corpora (e.g., a positive review set and a negative review set), the invariance between the two corpora is the content, whereas the variance is the style (e.g., sentiment, topic) (Mou and Vechtomova 2020).

Driven by the growing needs for TST, active research in this field has emerged, from the traditional linguistic approaches, to the more recent neural network–based approaches. Traditional approaches rely on term replacement and templates. For example, early work in NLG for weather forecasts builds domain-specific templates to express different types of weather with different levels of uncertainty for different users (Sripada et al. 2004; Reiter et al. 2005; Belz 2008; Gkatzia, Lemon, and Rieser 2017). Research that more explicitly focuses on TST starts from the frame language-based systems (McDonald and Pustejovsky 1985), and schema-based NLG systems (Hovy 1987, 1990) which generate text with pragmatic constraints such as formality under small-scale well-defined schema. Most of this earlier work required domain-specific templates, hand-featured phrase sets that express a certain attribute (e.g., friendly), and sometimes a look-up table of expressions with the same meaning but multiple different attributes (Bateman and Paris 1989; Stamatatos et al. 1997; Power, Scott, and Bouayad-Agha 2003; Reiter, Robertson, and Osman 2003; Sheikha and Inkpen 2011; Mairesse and Walker 2011).

With the success of deep learning in the last decade, a variety of neural methods have been recently proposed for TST. If parallel data are provided, standard sequence-to-sequence models are often directly applied (Rao and Tetreault 2018) (see Section 4). However, most use cases do not have parallel data, so TST on non-parallel corpora has become a prolific research area (see Section 5). The first line of approaches **disentangle** text into its content and attribute in the latent space, and apply generative modeling (Hu et al. 2017; Shen et al. 2017). This trend was then joined by another distinctive line of approach, **prototype editing** (Li et al. 2018), which extracts a sentence template and its attribute markers to generate the text. Another paradigm soon followed, namely, **pseudo-parallel corpus construction** to train the model as if in a supervised way with the pseudo-parallel data (Zhang et al. 2018d; Jin et al. 2019). These three directions, (1) disentanglement, (2) prototype editing, and (3) pseudo-parallel corpus construction, are further advanced with the emergence of Transformer-based models (Sudhakar, Upadhyay, and Maheswaran 2019; Malmi, Severyn, and Rothe 2020).

Given the advances in TST methodologies, it now starts to expand its impact to downstream applications, such as persona-based dialog generation (Niu and Bansal 2018; Huang et al. 2018), stylistic summarization (Jin et al. 2020a), stylized language modeling to imitate specific authors (Syed et al. 2020), online text debiasing (Pryzant et al. 2020; Ma et al. 2020), simile generation (Chakrabarty, Muresan, and Peng 2020), and many others.

Motivation of a Survey on TST. The increasing interest in modeling the style of text can be regarded as a trend reflecting the fact that NLP researchers start to focus more on user-centeredness and personalization. However, despite the growing interest in TST, the existing literature shows a large diversity in the selection of benchmark datasets,

Table 1
Overview of the survey.

Motivation	Data	Method	Extended Applications
• Artistic writing • Communication • Mitigating social issues	**Tasks** • Formality • Toxicity • Politeness • Authorship • Gender • Simplicity • Humor • Sentiment • Romance • Topic • Biasedness • Political slant **Key Properties** • Parallel vs. non-parallel • Uni- vs. bi-directional • Dataset size • Large vs. small word overlap	**On Parallel Data** • Multi-tasking • Inference techniques • Data augmentation **On Non-Parallel Data** • Disentanglement • Prototype editing • Pseudo data construction	**Helping Other NLP Tasks** • Paraphrasing • Data augmentation • Adversarial robustness • Persona-consistent dialog • Anonymization • Summarization • Style-specific MT

methodological frameworks, and evaluation metrics. Thus, the aim of this survey is to provide summaries and potential standardizations on some important aspects of TST, such as the terminology, problem definition, benchmark datasets, and evaluation metrics. We also aim to provide different perspectives on the methodology of TST, and suggest some potential cross-cutting research questions for our proposed research agenda of the field. As shown in Table 1, the key contributions targeted by this survey are as follows:

1. We conduct the first comprehensive review that covers most existing works (more than 100 papers) on deep learning-based TST.
2. We provide an overview of the task setting, terminology definition, benchmark datasets (Section 2), and evaluation metrics for which we proposed standard practices that can be helpful for future works (Section 3).
3. We categorize the existing approaches on parallel data (Section 4) and non-parallel data (Section 5) for which we distill some unified methodological frameworks.
4. We discuss a potential research agenda for TST (Section 6), including expanding the scope of styles, improving the methodology, loosening dataset assumptions, and improving evaluation metrics.
5. We provide a vision for how to broaden the impact of TST (Section 7), including connecting to more NLP tasks, and more specialized downstream applications, as well as considering some important ethical impacts.

Paper Selection. The neural TST papers reviewed in this survey are mainly from top conferences in NLP and artificial intelligence (AI), including ACL, EMNLP, NAACL, COLING, CoNLL, NeurIPS, ICML, ICLR, AAAI, and IJCAI. Other than conference papers, we also include some non-peer-reviewed preprint papers that can offer some

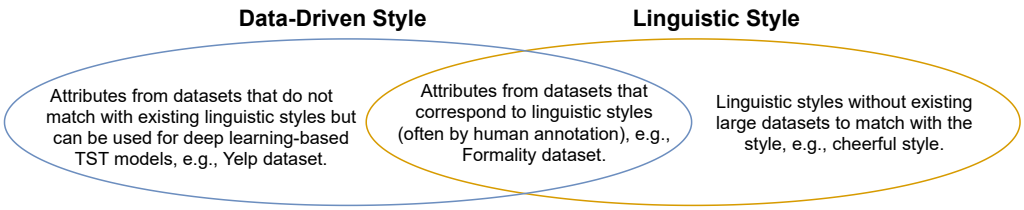


Figure 1
Venn diagram of the linguistic definition of style and data-driven definition of style.

insightful information about the field. The major factors for selecting non-peer-reviewed preprint papers include novelty and completeness, among others.

2. What Is Text Style Transfer?

This section provides an overview of the style transfer task. Section 2.1 goes through the definition of styles and the scope of this survey. Section 2.2 gives a task formulation and introduces the notations that will be used across the survey. Finally, Section 2.3 lists all the common subtasks for neural TST which can save the literature review efforts for future researchers.

2.1 How to Define Style?

Linguistic Definition of Style. An intuitive notion of style refers to the manner in which the semantics is expressed (McDonald and Pustejovsky 1985). Just as everyone has their own signatures, style originates as the characteristics inherent to every person's utterance, which can be expressed through the use of certain stylistic devices such as metaphors, as well as choice of words, syntactic structures, and so on. Style can also go beyond the sentence level to the discourse level, such as the stylistic structure of the entire piece of the work, for example, stream of consciousness, or flashbacks.

Beyond the intrinsic personal styles, for pragmatic uses, style further becomes a protocol to regularize the manner of communication. For example, for academic writing, the protocol requires formality and professionalism. Hovy (1987) defines style by its pragmatic aspects, including both personal (e.g., personality, gender) and interpersonal (e.g., humor, romance) aspects. Most existing literature also takes these well-defined categories of styles.

Data-Driven Definition of Style as the Scope of this Survey. This survey aims to provide an overview of existing neural TST approaches. To be concise, we will limit the scope to the most common settings of existing literature. Specifically, most deep learning work on TST adopts a **data-driven definition of style**, and the scope of this survey covers the styles in currently available TST datasets. The data-driven definition of style is different from the linguistic or rule-based definition of style, which theoretically constrains what constitutes a style and what not, such as a style guide (e.g., American Psychological Association 2020) that requires that formal text not include any contraction, e.g., "isn't." The distinction of the two defintions of style is shown in Figure 1.

With the rise of deep learning methods of TST, the data-driven definition of style extends the linguistic style to a broader concept—the general attributes in text. It regards "style" as the attributes that vary across datasets, as opposed to the characteristics that stay invariant (Mou and Vechtomova 2020). The reason is that deep learning models (which are the focus of this survey) need large corpora to learn the style from, but not all styles have well-matched large corpora. Therefore, apart from the very few manually annotated datasets with linguistic style definitions, such as formality (Rao and Tetreault 2018) and humor & romance (Gan et al. 2017), many recent dataset collection works automatically look for meta-information to link a corpus to a certain attribute. A typical example is the widely used Yelp review dataset (Shen et al. 2017), where reviews with low ratings are put into the negative corpus, and reviews with high ratings are put into the positive corpus, although the negative vs. positive opinion is not a style that belongs to the linguistic definition, but more of a content-related attribute.

Most methods mentioned in this survey can be applied to scenarios that follow this data-driven definition of style. As a double-edged sword, the prerequisite for most methods is that there *exist* style-specific corpora for each style of interest, either parallel or non-parallel. Note that there can be future works that do not take such an assumption, which will be discussed in Section 6.3.

Comparison of the Two Definitions. There are two phenomena rising from the data-driven definition of style as opposed to the linguistic style. One is that the data-driven definition of style can include a broader range of attributes including content and topic preferences of the text. The other is that data-driven styles, if collected through automatic classification by meta-information such as ratings, user information, and source of text, can be more ambiguous than the linguistically defined styles. As shown in Jin et al. (2019, Section 4.1.1), some automatically collected datasets have a concerningly high undecideable rate and inter-annotator disagreement rate when the annotators are asked to associate the dataset with human-defined styles such as political slant and gender-specific tones.

The advantage of the data-driven style is that it can marry well with deep learning methods because most neural models learn the concept of style by learning to distinguish the multiple style corpora. For the (non-data-driven) linguistic style, although it is under-explored in the existing deep learning works of TST, we provide in Section 6.3 a discussion of how potential future works can learn TST of linguistics styles with no matched data.

2.2 Task Formulation

We define the main notations used in this survey in Table 2.

As mentioned previously in Section 2.1, most neural approaches assume a given set of attribute values $\mathbb{A}$, and each attribute value has its own corpus. For example, if the task is about formality transfer, then for the attribute of text formality, there are two attribute values, a = "formal" and a' = "informal," corresponding to a corpus X_1 of formal sentences and another corpus X_2 of informal sentences. The style corpora can be parallel or non-parallel. Parallel data means that each sentence with the attribute a is paired with a counterpart sentence with another attribute a'. In contrast, non-parallel data only assumes mono-style corpora.

Table 2
Notation of each variable and its corresponding meaning.

Category	Notation	Meaning
Attribute	a	An attribute value, e.g., the formal style
	a'	An attribute value different from a
	$\mathbb{A}$	A predefined set of attribute values
	a_i	The i-th attribute value in $\mathbb{A}$
Sentence	x	A sentence with attribute value a
	x'	A sentence with attribute value a'
	X_i	A corpus of sentences with attribute value a_i
	x_i	A sentence from the corpus X_i
	$\widehat{x'}$	Attribute-transferred sentence of x learned by the model
Model	E	Encoder of a TST model
	G	Generator of a TST model
	f_c	Attribute classifier
	θ_E	Parameters of the encoder
	θ_G	Parameters of the generator
	θ_{f_c}	Parameters of the attribute classifier
Embedding	z	Latent representation of text, i.e., $z \triangleq E(x)$
	a	Latent representation of the attribute value in text

2.3 Existing Subtasks with Datasets

We list the common subtasks and corresponding datasets for neural TST in Table 3. The attributes of interest vary from style features (e.g., formality and politeness) to content preferences (e.g., sentiment and topics). Each task of which will be elaborated below.

Formality. Adjusting the extent of formality in text was first proposed by Hovy (1987). It is one of the most distinctive stylistic aspects that can be observed through many linguistic phenomena, such as more full names (e.g., "television") instead of abbreviations (e.g., "TV"), and more nouns (e.g., "solicitation") instead of verbs (e.g., "request"). The formality dataset, Grammarly's Yahoo Answers Formality Corpus (GYAFC) (Rao and Tetreault 2018), contains 50K formal-informal pairs retrieved by first getting 50K informal sentences from the Yahoo Answers corpus, and then recruiting crowdsource workers to rewrite them in a formal way. Briakou et al. (2021b) extend the formality dataset to a multilingual version with three more languages, Brazilian Portuguese, French, and Italian.

Politeness. Politeness transfer (Madaan et al. 2020) aims to control the politeness in text. For example, "Could you please send me the data?" is a more polite expression than "send me the data!". Madaan et al. (2020) compiled a dataset of 1.39 million automatically labeled instances from the raw Enron corpus (Shetty and Adibi 2004). As politeness is culture-dependent, this dataset mainly focuses on politeness in North American English.

Table 3
List of common subtasks of TST and their corresponding attribute values and datasets. For datasets with multiple attribute-specific corpora, we report their sizes by the number of sentences of the smallest of all corpora. We also report whether the dataset is parallel (*Pa?*).

Task	Attribute Values	Datasets	Size	Pa?
Style Features				
Formality	Informal↔Formal	GYAFC[3] (Rao and Tetreault 2018)	50K	✓
		XFORMAL[4] (Briakou et al. 2021b)	1K	✓
Politeness	Impolite→Polite	Politeness[5] (Madaan et al. 2020)	1M	✗
Gender	Masculine↔Feminine	Yelp Gender[6] (Prabhumoye et al. 2018)	2.5M	✗
Humor & Romance	Factual↔Humorous↔Romantic	FlickrStyle[7] (Gan et al. 2017)	5K	✓
Biasedness	Biased→Neutral	Wiki Neutrality[8] (Pryzant et al. 2020)	181K	✓
Toxicity	Offensive→Non-offensive	Twitter (dos Santos, Melnyk, and Padhi 2018)	58K	✗
		Reddit (dos Santos, Melnyk, and Padhi 2018)	224K	
		Reddit Politics (Tran, Zhang, and Soleymani 2020)	350K	
Authorship	Shakespearean↔Modern	Shakespeare (Xu et al. 2012)	18K	✓
	Different Bible translators	Bible[9] (Carlson, Riddell, and Rockmore 2018)	28M	
Simplicity	Complicated→Simple	PWKP (Zhu, Bernhard, and Gurevych 2010)	108K	✓
		Expert (den Bercken, Sips, and Lofi 2019)	2.2K	✓
		MIMIC-III[10] (Weng, Chung, and Szolovits 2019)	59K	✗
		MSD[11] (Cao et al. 2020)	114K	✓
Engagingness	Plain→Attractive	Math[12] (Koncel-Kedziorski et al. 2016)	<1K	✓
		TitleStylist[13] (Jin et al. 2020a)	146K	✗
Content Preferences				
Sentiment	Positive↔Negative	Yelp[14] (Shen et al. 2017)	250K	✗
		Amazon[15] (He and McAuley 2016)	277K	
Topic	Entertainment↔Politics	Yahoo! Answers[16] (Huang et al. 2020)	153K	✗
Politics	Democratic↔Republican	Political[17] (Voigt et al. 2018)	540K	✗

[3]GYAFC data: `https://github.com/raosudha89/GYAFC-corpus`.
[4]GYAFC data: `https://github.com/Elbria/xformal-FoST`.
[5]Politeness data: `https://github.com/tag-and-generate/politeness-dataset`.
[6]The Yelp Gender dataset is from the Yelp Challenge `https://www.yelp.com/dataset` and its preprocessing needs to follow Prabhumoye et al. (2018).
[7]FlickrStyle data: `https://github.com/lijuncen/Sentiment-and-Style-Transfer/tree/master/data/imagecaption`.
[8]Wiki Neutrality data: `http://bit.ly/bias-corpus`.
[9]Bible data: `https://github.com/keithecarlson/StyleTransferBibleData`.
[10]MIMIC-III data: Request access at `https://mimic.physionet.org/gettingstarted/access/` and follow the preprocessing of Weng, Chung, and Szolovits (2019).
[11]MSD data: `https://srhthu.github.io/expertise-style-transfer/`.
[12]Math data: `https://gitlab.cs.washington.edu/kedzior/Rewriter/`.
[13]TitleStylist data: `https://github.com/jind11/TitleStylist`.
[14]Yelp data: `https://github.com/shentianxiao/language-style-transfer`.
[15]Amazon data: `https://github.com/lijuncen/Sentiment-and-Style-Transfer/tree/master/data/amazon`.
[16]Yahoo! Answers data: `https://webscope.sandbox.yahoo.com/catalog.php?datatype=l&did=11`.
[17]Political data: `https://nlp.stanford.edu/robvoigt/rtgender/`.

Gender. Linguistic phenomena related to gender is a heated research area (Trudgill 1972; Lakoff 1973; Tannen 1990; Argamon et al. 2003; Boulis and Ostendorf 2005). The gender-related TST dataset is proposed by Prabhumoye et al. (2018), who compiled 2.5M reviews from Yelp Dataset Challenge that are labeled with the gender of the user.

Humor & Romance. Humor and romance are some artistic attributes that can provide readers with joy. Li et al. (2018) first propose to borrow the FlickrStyle stylized caption dataset (Gan et al. 2017) from the computer vision domain. In the FlickrStyle image caption dataset, each image has three captions, with a factual, a humorous, and a romantic style, respectively. By keeping only the captions of the three styles, Li et al. (2018) created a subset of the FlickrStyle dataset of 5K parallel (factual, humorous, romantic) triplets.

Biasedness. Wiki Neutrality Corpus (Pryzant et al. 2020) is the first corpus of biased and neutralized sentence pairs. It is collected from Wikipedia revisions that adjusted the tone of existing sentences to a more neutral voice. The types of bias in the biased corpus include framing bias, epistemological bias, and demographic bias.

Toxicity. Another important use of TST is to fight against offensive language. Tran, Zhang, and Soleymani (2020) collect 350K offensive sentences and 7M non-offensive sentences by crawling sentences from Reddit using a list of restricted words.

Authorship. Changing the tone of the author is an artistic use of TST. Xu et al. (2012) created an aligned corpus of 18K pairs of Shakespearean English and their modern English translation. Carlson, Riddell, and Rockmore (2018) collected 28M parallel data from English versions of the Bible by different translators.

Simplicity. Another important use of TST is to lower the language barrier for readers, such as translating legalese, medical jargon, or other professional text into simple English, to avoid discrepancies between expert wordings and lay understanding (Tan and Goonawardene 2017). Common tasks include converting standard English Wikipedia into Simple Wikipedia, whose dataset contains 108K samples (Zhu, Bernhard, and Gurevych 2010). Another task is to simplify medical descriptions to patient-friendly text, including a dataset with 2.2K samples (den Bercken, Sips, and Lofi 2019), another non-parallel dataset with 59K free-text discharge summaries compiled from MIMIC-III (Weng, Chung, and Szolovits 2019), and a more recent parallel dataset with 114K samples compiled from the health reference Merck Manuals (MSD), where discussions on each medical topic has one version for professionals, and the other for consumers (Cao et al. 2020).

Sentiment. Sentiment modification is the most popular task in previous work on TST. It aims to change the sentiment polarity in reviews, for example, from a negative review to a positive review, or vice versa. There is also work on transferring sentiments on fine-grained review ratings (e.g., 1–5 scores). Commonly used datasets include Yelp reviews (Shen et al. 2017) and Amazon product reviews (He and McAuley 2016).

Topic. There are a few works that cover topic transfer. For example, Huang et al. (2020) form a two-topic corpus by compiling Yahoo! Answers under two topics, entertainment

and politics, respectively. There is also a recent dataset with 21 text styles such as Sciences, Sport, Politics, and others (Zeng, Shoeybi, and Liu 2020).

Political Slant. Political slant transfer proposed by Prabhumoye et al. (2018) aims to transfer the political view in text. For example, a Republican's comment can be "defund all illegal immigrants," while Democrats are more likely to support humanistic actions towards immigrants. The political slant dataset (Voigt et al. 2018) is collected from comments on Facebook posts of the United States Senate and House members. The dataset uses top-level comments directly responding to the posts of a Democratic or Republican congressperson. There are 540K training, 4K development, and 56K test instances in the dataset.

Combined Attributes. Lample et al. (2019) propose a more challenging setting of text attribute transfer: multi-attribute transfer. For example, the source sentence can be a positive review on an Asian restaurant written by a male reviewer, and the target sentence is a negative review on an American restaurant written by a female. Each of their datasets has 1–3 independent categories of attributes. Their first dataset is FYelp, which is compiled from the Yelp Dataset Challenge, labeled with sentiment (positive or negative), gender (male or female), and eatery category (American, Asian, Mexican, bar, or dessert). Their second dataset, Amazon, which is based on the Amazon product review dataset (Li et al. 2018), contains the following attributes: sentiment (positive or negative), and product category (book, clothing, electronics, movies, or music). Their third dataset, Social Media Content dataset, collected from internal Facebook data that is private, contains gender (male or female), age group (18–24 or 65+), and writer-annotated feeling (relaxed or annoyed).

3. How to Evaluate Style Transfer?

A successful style-transferred output not only needs to demonstrate the correct target style; but also, due to the uncontrollability of neural networks, we need to verify that it preserves the original semantics, and maintains natural language fluency. Therefore, the commonly used practice of evaluation considers the following three criteria: (1) transferred style strength, (2) semantic preservation, and (3) fluency.

We will first introduce the practice of automatic evaluation on the three criteria, discuss the benefits and caveats of automatic evaluation, and then introduce human evaluation as a remedy for some of the intrinsic weaknesses of automatic evaluation. Finally, we will suggest some standard practice of TST evaluation for future work. The overview of evaluation methods regarding each criterion is listed in Table 4.

Table 4
Overview of evaluation methods for each criterion.

Criterion	Automatic Evaluation	Human Evaluation
Overall	BLEU with gold references	Rating or ranking
- Transferred Style Strength	Accuracy by a separately trained style classifier	Rating or ranking
- Semantic Preservation	BLEU/ROUGE/etc. with (modified) inputs	Rating or ranking
- Fluency	Perplexity by a separately trained language model	Rating or ranking

3.1 Automatic Evaluation

Automatic evaluation provides an economic, reproducible, and scalable way to assess the quality of generation results. However, due to the complexities of natural language, each metric introduced below can address certain aspects, but also has intrinsic blind spots.

BLEU with Gold References. Similar to many text generation tasks, TST also has human-written references on several datasets (Yelp, Captions, etc.), so it is common to use the BLEU score (Papineni et al. 2002) between the gold references and model outputs. Using BLEU to evaluate TST models has been seen across pre-deep learning works (Xu et al. 2012; Jhamtani et al. 2017) and deep learning approaches (Rao and Tetreault 2018; Li et al. 2018; Jin et al. 2019).

There are three problems with using BLEU between the gold references and model outputs:

Problem 1. It mainly evaluates content and simply copying the input can result in high BLEU scores.
Problem 2. BLEU is shown to have low correlation with human evaluation.
Problem 3. Some datasets do not have human-written references.

Problem 1: Different from machine translation, where using BLEU only is sufficient, TST has to consider the caveat that simply copying the input sentence can achieve high BLEU scores with the gold references on many datasets (e.g., $\sim$40 on Yelp, $\sim$20 on Humor & Romance, $\sim$50 for informal-to-formal style transfer, and $\sim$30 for formal-to-informal style transfer). This is because most text rewrites have a large extent of n-gram overlap with the source sentence. In contrast, machine translation does not have this concern, because the vocabulary of its input and output are different, and copying the input sequence does not give high BLEU scores. A possible fix to consider is to combine BLEU with PINC (Chen and Dolan 2011) as in paraphrasing (Xu et al. 2012; Jhamtani et al. 2017). By using PINC and BLEU as a 2-dimensional metric, we can minimize the n-gram overlap with the source sentence but maximize the n-gram overlap with the reference sentences.

Problems 2 & 3: Other problems include insufficient correlation of BLEU with human evaluations (e.g., $\leq$0.30 with respect to human-rated grammaticality shown in Li et al. [2018] and $\leq$0.45 with respect to human evaluations shown in Mir et al. [2019]), and the unavailability of human-written references for some datasets (e.g., gender and political datasets [Prabhumoye et al. 2018], and the politeness dataset [Madaan et al. 2020]). A commonly used fix is to make the evaluation more fine-grained using three different independent aspects, namely, transferred style strength, semantic preservation, and fluency, which will be detailed below.

Transferred Style Strength. To automatically evaluate the transferred style strength, most works separately train a style classifier to distinguish the attributes (Hu et al. 2017; Shen et al. 2017; Fu et al. 2018; Li et al. 2018; Prabhumoye et al. 2018).[18] This classifier is used to judge whether each sample generated by the model conforms to the target attribute. The

18 Note that this style classifier usually reports 80+% or 90+% accuracy, and we will discuss the problem of false positives and false negatives in the last paragraph of this section.

transferred style strength is calculated as $\frac{\text{\# test samples correctly classified}}{\text{\# all test samples}}$. Li et al. (2018) shows that the attribute classifier correlates well with human evaluation on some datasets (e.g., Yelp and Captions), but has almost no correlation with others (e.g., Amazon). The reason is that some product genres has a dominant number of positive or negative reviews.

Semantic Preservation. Many metrics can be applied to measure the similarity between the input and output sentence pairs, including BLEU (Papineni et al. 2002), ROUGE (Lin and Och 2004), METEOR (Banerjee and Lavie 2005), chrF (Popović 2015), and Word Mover Distance (WMD) (Kusner et al. 2015). Recently, some additional deep-learning-based metrics have been proposed, such as cosine similarity based on sentence embeddings (Fu et al. 2018), and BERTScore (Zhang et al. 2020). There are also evaluation metrics that are specific for TST such as the Part-of-Speech distance (Tian, Hu, and Yu 2018). Another newly proposed metric is to first delete all attribute-related expressions in the text, and then apply the above similarity evaluations (Mir et al. 2019). Among all the metrics, Mir et al. (2019) and Yamshchikov et al. (2021) showed that METEOR and WMD have better correlation with human evaluation than BLEU, although, in practice, BLEU is the most widely used metric to evaluate the semantic similarity between the source sentence and style-transferred output (Yang et al. 2018; Madaan et al. 2020).

Fluency. Fluency is a basic requirement for natural language outputs. To automate this evaluation, perplexity is calculated via a language model (LM) pretrained on the training data of all attributes (Yang et al. 2018). However, the effectiveness of perplexity remains debatable, as Pang and Gimpel (2019) showed its high correlation with human ratings of fluency, whereas Mir et al. (2019) suggested no significant correlation between perplexity and human scores. We note that perplexity by LM can suffer from the following undesired properties:

1. Biased toward shorter sentences than longer sentences.
2. For the same meaning, less frequent words will have worse perplexity (e.g., agreeable) than frequent words (e.g., good).
3. A sentence's own perplexity will change if the sentence prior to it changes.
4. LMs are not good enough yet.
5. LMs do not necessarily handle well the domain shift between their training corpus and the style-transferred text.
6. Perplexity scores produced by LMs are sensitive to the training corpora, LM architecture and configuration, as well as optimization configuration. Therefore, different models' outputs must be evaluated by exactly the same LM for fair comparison, which adds more difficulty to benchmarking.

Such properties will bias against certain models, which is not desired for an evaluation metric. As a potential remedy, future researchers can try grammaticality checker to score the generated text.

Task-Specific Criteria. As TST can serve as a component for other downstream applications, some task-specific criteria are also proposed to evaluate the quality of generated text. For example, Reiter, Robertson, and Osman (2003) evaluated the effect of their

tailored text on reducing smokers' intent to smoke through clinical trials. Jin et al. (2020a) applied TST to generate eye-catchy headlines so they have an attractive score, and future works in this direction can also test the click-through rates. Hu et al. (2017) evaluated how the generated text as augmented data can improve the downstream attribute classification accuracy.

Tips for Automatic Metrics. For the evaluation metrics that rely on the pretrained models, namely, the style classifier and LM, we need to beware of the following:

1. The pretrained models for automatic evaluation should be separate from the proposed TST model.
2. Machine learning models can be imperfect, so we should be aware of the potential false positives and false negatives.
3. The pretrained models are imperfect in the sense that they will favor toward a certain type of methods.

For the first point, it is important to not use the same style classifier or LM in the proposed TST approach, otherwise it can overfit or hack the metrics.

For the second point, we need to understand what the false positives and false negatives of the generated outputs can be. An illustrative example is that if the style classifier only reports 80+% performance (e.g., on the gender dataset [Prabhumoye et al. 2018] and Amazon dataset [Li et al. 2018]), even perfect style rewrites can only score 80+%, but maybe an imperfect model can score 90% because it can resemble the imperfect style classification model more and takes advantage of the *false positives*. Other reasons for false positives can be adversarial attacks. Jin et al. (2020b) showed that merely paraphrasing using synonyms can drop the performance of high-accuracy classification models from TextCNN (Kim 2014) to BERT (Devlin et al. 2019) by 90+%. Therefore, higher scores by the style classifier do not necessarily indicate more successful transfer. Moreover, the style classifier can produce *false negatives* if there is a distribution shift between the training data and style-transferred outputs. For example, in the training corpus, a product may appear often with the *positive* attribute, and in the style-transferred outputs, this product co-occurs with the opposite, *negative* attribute. Such false negatives are observed on the Amazon product review dataset (Li et al. 2018). On the other hand, the biases of the LM correlate with sentence length, synonym replacement, and prior context.

The third point is a direct result implied by the second point, so in practice, we need to keep in mind and check whether the proposed model takes advantage of the evaluation metrics or makes improvements that are generalizable.

3.2 Human Evaluation

Compared with the pros and cons of the automatic evaluation metrics mentioned above, human evaluation stands out for its flexibility and comprehensiveness. For example, when asking humans to evaluate the fluency, we do not need to worry for the bias toward shorter sentences as in the LM. We can also design criteria that are not computationally easy such as comparing and ranking the outputs of multiple models. There are several ways to conduct human evaluation. In terms of evaluation types, there are pointwise scoring, namely, asking humans to provide absolute scores of the model outputs, and pairwise comparison, namely asking humans to judge which of the two

outputs is better, or providing a ranking for multiple outputs. In terms of the criteria, humans can provide overall evaluation, or separate scores for transferred style strength, semantic preservation, and fluency.

However, the well-known limitations of human evaluation are cost and irreproducibility. Performing human evaluations can be time consuming, which may result in significant time and financial costs. Moreover, the human evaluation results in two studies are often not directly comparable, because human evaluation results tend to be subjective and not easily irreproducible (Belz et al. 2020). Moreover, some styles are very difficult to evaluate without expertise and extensive reading experience.

As a remedy, we encourage future researchers to report inter-rater agreement scores such as the Cohen's kappa (Cohen 1960) and Krippendorff's alpha (Krippendorff 2018). Briakou et al. (2021a) also recommends standardizing and describing evaluation protocols (e.g., linguistic background of the annotators, compensation, detailed annotation instructions for each evaluation aspect), and releasing annotations.

Tips for Human Evaluation. As common practice, most works use 100 outputs for each style transfer direction (e.g., 100 outputs for formal→informal, and 100 outputs for informal→formal), and two human annotators for each task (Shen et al. 2017; Fu et al. 2018; Li et al. 2018).

3.3 Suggested Evaluation Settings for Future Work

Currently, the experiments of various TST work do not adopt the same setting, making it difficult to do head-to-head comparison among the empirical results of multiple studies. Although it is reasonable to customize the experimental settings according to the needs of a certain study, it is suggested to at least use the standard setting in at least one of the many reported experiments, to make it easy to compare with previous and future studies. For example, at least (1) experiment on at least one commonly used dataset, (2) list up-to-date best-performing previous models as baselines, (3) report on a superset of the most commonly used metrics, and (4) release system outputs.

For (1), we suggest that future work use at least one of the most commonly used benchmark datasets, such as the Yelp data preprocessed by Shen et al. (2017) and its five human references provided by Jin et al. (2019), Amazon data preprocessed by Li et al. (2018), and formality data provided by Rao and Tetreault (2018).

For (2), we suggest that future studies actively check the latest style transfer papers curated at `https://github.com/fuzhenxin/Style-Transfer-in-Text` and our repository `https://github.com/zhijing-jin/Text_Style_Transfer_Survey`, and compare with the state-of-the-art performances instead of older ones. We also call for more reproducibility in the community, including source codes and evaluation codes, because, for example, there are several different scripts to evaluate the BLEU scores.

For (3), because no single evaluation metric is perfect and comprehensive enough for TST, it is strongly suggested to use both human and automatic evaluation on three criteria. In evaluation, apart from customized use of metrics, we suggest that most future work include at least the following evaluation practices:

- Human evaluation: Rate at least two state-of-the-art models according to the curated paper lists.
- Automatic evaluation: At least report the BLEU score with all available references if there exist human-written references (e.g., the five references

for the Yelp dataset provided by Jin et al. [2019]), and BLEU with the input only when there are no human-written references.

For (4), it will also be very helpful to provide system outputs for each TST paper, so that future works can better reproduce both human and automatic evaluation results. Note that releasing system outputs can help future studies' comparison of automatic evaluation results, because there can be different scripts to evaluate the BLEU scores, as well as different style classifiers and LM. It will be a great addition to the TST community if future work can establish an online leaderboard, let existing research groups upload their output files, and automatically evaluate the model outputs using a standard set of automatic evaluation scripts.

4. Methods on Parallel Data

Over the last several years, various methods have been proposed for TST. In general, they can be categorized based on whether the dataset has parallel text with different styles or several non-parallel mono-style corpora. The rightmost column "Pa?" in Table 3 shows whether there exist parallel data for each TST subtask. In this section, we will cover TST methods on parallel datasets, and in Section 5 we will detail the approaches on non-parallel datasets. To ease the understanding for the reader, we will in most cases explain TST on one attribute between two values, such as transferring the formality between informal and formal tones, which can potentially be extended to multiple attributes.

Most methods adopt the standard neural sequence-to-sequence (seq2seq) model with the encoder-decoder architecture, which was initially developed for neural machine translation (NMT) (Sutskever, Vinyals, and Le 2014; Bahdanau, Cho, and Bengio 2015; Cho et al. 2014) and extensively seen on text generation tasks such as summarization (Rush, Chopra, and Weston 2015) and many others (Song et al. 2019). The encoder-decoder seq2seq model can be implemented by either LSTM as in Rao and Tetreault (2018); and Shang et al. (2019) or Transformer (Vaswani et al. 2017) as in Xu, Ge, and Wei (2019). Copy mechanism (Gülçehre et al. 2016; See, Liu, and Manning 2017) is also added to better handle stretches of text that should not be changed (e.g., some proper nouns and rare words) (Gu et al. 2016; Merity et al. 2017). Based on this architecture, recent work has developed multiple directions of improvement: **multi-tasking**, **inference techniques**, and **data augmentation**, which will be introduced below.

Multi-tasking. In addition to the seq2seq learning on paired attributed-text, Xu, Ge, and Wei (2019) propose adding three other loss functions: (1) classifier-guided loss, which is calculated using a well-trained attribute classifier and encourages the model to generate sentences conforming to the target attribute, (2) self-reconstruction loss, which encourages the seq2seq model to reconstruct the input itself by specifying the desired style the same as the input style, and (3) cycle loss, which first transfers the input sentence to the target attribute and then transfers the output back to its original attribute. Each of the three losses can gain performance improvement of 1–5 BLEU points with the human references (Xu, Ge, and Wei 2019). Another type of multi-tasking is to jointly learn TST and machine translation from French to English, which improves the performance by 1 BLEU score with human-written references (Niu, Rao, and Carpuat 2018). Specifically for formality transfer, Zhang, Ge, and Sun (2020) multi-task TST and grammar error correction (GEC) so that knowledge from GEC data can be transferred to the informal-to-formal style transfer task.

Apart from the additional loss designs, using the pretrained language model GPT-2 (Radford et al. 2019) can lead to improvement by at least 7 BLEU points with human references Wang et al. (2019).

Inference Techniques. To avoid the model copying too many parts of the input sentence and not performing sufficient edits to flip the attribute, Kajiwara (2019) first identifies words in the source sentence requiring replacement, and then changes the words by negative lexically constrained decoding (Post and Vilar 2018) that avoids naive copying. Because this method only changes the beam search process for model inference, it can be applied to any TST model without model re-training.

Data Augmentation. Because style transfer data is expensive to annotate, there are not as many parallel datasets as in machine translation. Hence, various methods have been proposed for data augmentation to enrich the data. For example, Rao and Tetreault (2018) first train a phrase-based machine translation (PBMT) model on a given parallel dataset and then use back-translation (Sennrich, Haddow, and Birch 2016b) to construct a pseudo-parallel dataset as additional training data, which leads to an improvement of around 9.7 BLEU points with respect to human written references.

Most recently, Zhang, Ge, and Sun (2020) use a data augmentation technique by making use of largely available online text. They scrape informal text from online forums and generate back-translations, that is, informal English $\rightarrow$ a pivot language such as French $\rightarrow$ formal English, where the formality of the back-translated English text is ensured with a formality classifier that is used to only keep text that is classified as formal text.

5. Methods on Non-Parallel Data

Parallel data for TST is difficult to obtain, and for some styles impossible to crowd-source (e.g., Mark Twain novels rewritten in Hemmingway's style). Hence, the majority of TST methods assume only non-parallel mono-style corpora, and investigate how to build deep learning models based on this constraint. In this section, we will introduce three main branches of TST methods: disentanglement (Section 5.1), prototype editing (Section 5.2), and pseudo-parallel corpus construction (Section 5.3).

5.1 Disentanglement

Disentanglement-based models usually perform the following three actions:

- Encode the text x with attribute a into a latent representation z (i.e., $x \rightarrow z$)
- Manipulate the latent representation z to remove the source attribute (i.e., $z \rightarrow z'$)
- Decode into text x' with the target attribute a' (i.e., $z' \rightarrow x'$)

To build such models, the common workflow in disentanglement papers consists of the following three steps:

Step 1. Select a model as the backbone for the encoder-decoder learning (Section 5.1.1).
Step 2. Select a manipulation method of the latent representation (Section 5.1.2).

Step 3. For the manipulation method chosen above, select (multiple) appropriate loss functions (Section 5.1.3).

The organization of this section starts with Section 5.1.1, which introduces the encoder-decoder training objectives that is used for Step 1. Next, Section 5.1.2 overviews three main approaches to manipulate the latent representation for Step 2, and Section 5.1.3 goes through a plethora of training objectives for Step 3. Table 5 provides an overview of existing models and their corresponding configurations. To give a rough idea of the effectiveness of each model, we show their performance on the Yelp dataset.

5.1.1 Encoder-Decoder Training Method. There are three model choices to obtain the latent representation z from the discrete text x and then decode it into the new text x' via reconstruction training: auto-encoder (AE), variational auto-encoder (VAE), and generative adversarial networks (GANs).

Auto-Encoder (AE). Auto-encoding is a commonly used method to learn the latent representation z, which first encodes the input sentence x into a latent vector z and then reconstructs a sentence as similar to the input sentence as possible. AE is used in many TST works (e.g., Shen et al. 2017; Hu et al. 2017; Fu et al. 2018; Zhao et al. 2018; Prabhumoye et al. 2018; Yang et al. 2018). To avoid auto-encoding from blindly copying

Table 5
Summary of existing disentanglement-based methods and the setting they adopted, with a reference of their performance on the Yelp dataset. For the settings, we include the encoder-decoder training method (Enc-Dec) in Section 5.1.1, the disentanglement method (Disen.) in Section 5.1.2, and the loss types used to control style (Style Control) and content (Content Control) in Section 5.1.3. For the model performance, we report automatic evaluation scores including BLEU with the one human reference (BL-Ref) provided by Li et al. (2018), accuracy (Acc.), BLEU with the input (BL-Inp), and perplexity (PPL). * marks numbers reported by Liu et al. (2020). Readers can refer to Hu, Lee, and Aggarwal (2020) for more complete performance results on Yelp.

	Settings				Performance on Yelp			
	Enc-Dec	Disen.	Style Control	Content Control	BL-Ref	Acc. (%)	BL-Inp	PPL↓
Mueller, Gifford, and Jaakkola (2017)	VAE	LRE	–	–	–	–	–	–
Hu et al. (2017)	VAE	ACC	ACO	–	22.3	86.7	58.4	–
Shen et al. (2017)	AE&GAN	ACC	AdvR‖AdvO	–	7.8	73.9	20.7	72*
Fu et al. (2018)	AE	ACC	AdvR	–	12.9	46.9	40.1	166.5*
Prabhumoye et al. (2018)	AE	ACC	ACO	–	6.8	87.2	–	32.8*
Zhao et al. (2018)	GAN	ACC	AdvR	–	–	73.4	31.2	29.7
Yang et al. (2018)	AE	ACC	LMO	–	–	91.2	57.8	47.0&60.9
Logeswaran, Lee, and Bengio (2018)	AE	ACC	AdvO	Cycle	–	90.5	–	133
Tian, Hu, and Yu (2018)	AE	ACC	AdvO	Noun	24.9	92.7	63.3	–
Liao et al. (2018)	VAE	LRE	–	–	–	88.3	–	–
Romanov et al. (2019)	AE	LRS	ACR&AdvR	–	–	–	–	–
John et al. (2019)	AE&VAE	LRS	ACR&AdvR	BoW&AdvBoW	–	93.4	–	–
Bao et al. (2019)	VAE	LRS	ACR&AdvR	BoW&AdvBoW	–	–	–	–
Dai et al. (2019)	AE	ACC	ACO	Cycle	20.3	87.7	54.9	73
Wang, Hua, and Wan (2019)	AE	LRE	–	–	24.6	95.4	–	46.2
Li et al. (2020)	GAN	ACC	ACO&AdvR	–	–	95.5	53.3	–
Liu et al. (2020)	VAE	LRE	–	–	18.8	92.3	–	18.3
Yi et al. (2020)	VAE	ACC	ACO	Cycle	26.0	90.8	–	109
Jin et al. (2020a)	AE	LRE	–	–	–	–	–	–

all the elements from the input, Hill, Cho, and Korhonen (2016) adopt denoising auto-encoding (DAE) (Vincent et al. 2010) to replace AE in NLP tasks. Specifically, DAE first passes the input sentence x through a noise model to randomly drop, shuffle, or mask some words, and then reconstructs the original sentence from this corrupted sentence. This idea is used in later TST works, for example, Lample et al. (2019) and Jin et al. (2020a). As pre-trained models became prevalent in recent years, the DAE training method has increased in popularity relative to its counterparts such as GAN and VAE, because pre-training over large corpora can grant models better performance in terms of semantic preservation and fluency (Lai, Toral, and Nissim 2021; Riley et al. 2021).

Variational Auto-Encoder (VAE). Instead of reconstructing data based on the deterministic latent representations by AE, a variational auto-encoder (VAE) (Kingma and Welling 2014; Rezende, Mohamed, and Wierstra 2014) reconstructs data based on the sampled latent vector from its posterior, and uses the regularization by Kullback–Leibler divergence. VAE is also commonly used in TST works (Mueller, Gifford, and Jaakkola 2017; Hu et al. 2017; Liu et al. 2020; Liao et al. 2018; Yi et al. 2020; Tikhonov et al. 2019). The VAE loss is formulated as

$$\mathcal{L}_{\text{VAE}}(\theta_{\text{E}}, \theta_{\text{G}}) = -\mathbb{E}_{q_{\text{E}}(z|x)} \log p_{\text{G}}(x|z) + \lambda \text{KL}\Big[q_{\text{E}}(z|x)||p(z)\Big] \tag{1}$$

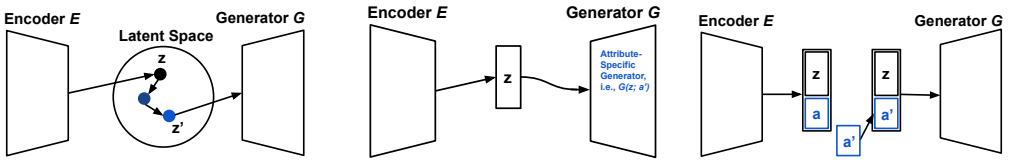
where λ is the hyper-parameter to balance the reconstruction loss and the KL term, $p(z)$ is the prior drawn from the standard normal distribution of $\mathcal{N}(0, I)$, and $q_{\text{E}}(z|x)$ is the posterior in the form of $\mathcal{N}(\mu, \sigma)$, where μ and σ are predicted by the encoder.

Generative Adversarial Networks (GANs). GANs (Goodfellow et al. 2014) can also be applied to TST (Shen et al. 2017; Zhao et al. 2018; Li et al. 2020). The way GANs work is to first approximate the samples drawn from a true distribution z by using a noise sample s and a generator function G to produce $\widehat{z} = G(s)$. Next, a critic/discriminator $f_c(z)$ is used to distinguish real data and generated samples. The critic is trained to distinguish the real samples from generated samples, and the generator is trained to fool the critic. Formally, the training process is expressed as a min-max game played among the encoder E, generator G, and the critic f_c:

$$\max_{c} \min_{E,G} \mathcal{L}_{\text{GAN}} = -\mathbb{E}_{p(z)} \log p_{\text{G}}(x|z) + \mathbb{E}_{p(z)} f_c(z) - \mathbb{E}_{p(\widehat{z})} f_c(\widehat{z}) \tag{2}$$

5.1.2 Latent Representation Manipulation. Based on the general encoder and decoder training method, the core element of disentanglement is the manipulation of latent representation z. Figure 2 illustrates three main methods: latent representation editing, attribute code control, and latent representation splitting. In addition, the "Disen." column of Table 5 shows the type of latent representation manipulation for each work in disentanglement.

The first approach, **Latent Representation Editing** (LRE), shown in Figure 2a, is achieved by ensuring two properties of the latent representation z. The first property is that z should be able to serve as the latent representation for auto-encoding, namely, aligning $f_c(z)$ with the input x, where $z \triangleq E(x)$. The second property is that z should be learned such that it incorporates the new attribute value of interest a'. To achieve this, the common practice is to first learn an attribute classifier f_c, for example, a multilayer



(a) Latent Representation Editing (b) Attribute Code Control (c) Latent Representation Splitting

Figure 2
Three methods to manipulate the latent space based on disentanglement for TST.

perceptron that takes the latent representation z as input, and then iteratively update z within the constrained space by the first property and at the same time maximize the prediction confidence score regarding a' by this attribute classifier (Mueller, Gifford, and Jaakkola 2017; Liao et al. 2018; Wang, Hua, and Wan 2019; Liu et al. 2020). An alternative way to achieve the second property is to multi-task by another auto-encoding task on the corpus with the attribute a' and share most layers of the transformer except the query transformation and layer normalization layers (Jin et al. 2020a).

The second approach, **Attribute Code Control** (ACC), as shown in Figure 2b, first enforces the latent representation z of the sentence x to contain all information except its attribute value a via adversarial learning, and then the transferred output is decoded based on the combination of z and a structured attribute code a corresponding to the attribute value a. During the decoding process, the attribute code vector a controls the attribute of generated text by acting as either the initial state (Shen et al. 2017; Yi et al. 2020) or the embedding (Fu et al. 2018; Dai et al. 2019).

The third approach, **Latent Representation Splitting** (LRS), as illustrated in Figure 2c, first disentangles the input text into two parts: the latent attribute representation a, and semantic representation z that captures attribute-independent information. We then replace the source attribute a with the target attribute a', and the final transferred text is generated using the combination of z and a' (John et al. 2019; Romanov et al. 2019).

5.1.3 Training Objectives. When disentangling the attribute information a and the attribute-independent semantic information z, we need to achieve two aims:

Aim 1. The target attribute is *fully* and *exclusively* controlled by a (and not z). We typically use style-oriented losses to achieve this aim (Section 5.1.3.1).

Aim 2. The attribute-independent information is *fully* and *exclusively* captured by z (and not a). Content-oriented losses are more often used for this aim (Section 5.1.3.2).

We describe the various style-oriented and content-oriented losses below.

5.1.3.1 Style-Oriented Losses

To achieve Aim 1, many different *style-oriented losses* have been proposed, to nudge the model to learn a more clearly disentangled a and exclude the attribute information from z.

Attribute Classifier on Outputs (ACO). ACO aims to make sentences generated by the generator G carry the target attribute a' according to a pre-trained attribute classifier

f_c (Hu et al. 2017; Prabhumoye et al. 2018; Yamshchikov et al. 2019). The generator G takes as input the learned attribute vector $\widehat{a'}$, which can be either an attribute code vector trained from scratch (as in the ACC approach) or the attribute representation disentangled from text (by the LRS approach). We denote the generation process to obtain the transferred sentence $\widehat{x'}$ as $\widehat{x'} \triangleq G(E(x); a')$. Correspondingly, ACO minimizes the following learning objective:

$$\mathcal{L}_{\text{ACO}}(\theta_{\text{G}}, a') = -\mathbb{E}_{p(x)} \log f_c(x') \tag{3}$$

In training, ACO can be trained in two ways: either a normal loss function trained by Gumbel-softmax distribution to approximate the discrete training (Jang, Gu, and Poole 2017), or a negative reward for reinforcement learning by policy gradient training (Williams 1992) as in Luo et al. (2019).

Attribute Classifier on Representations (ACR). Different from the previous ACO objective, whose training signal is from the output sentence $\widehat{x'}$, ACR directly enforces the disentangled attribute representation a to be correctly classified by the attribute classifier, by the following objective (John et al. 2019; Romanov et al. 2019):

$$\mathcal{L}_{\text{ACR}}(\theta_{\text{E}}, \theta_{f_c}) = -\mathbb{E}_{p(a)} \log f_c(a) \tag{4}$$

Adversarial Learning on Representations (AdvR). As the previous ACR explicitly requires the latent a to be classified by f_c, AdvR trains from another perspective—enforcing that no attribute-related information is contained in z (Fu et al. 2018; Zhao et al. 2018; Romanov et al. 2019; John et al. 2019; Tikhonov et al. 2019; Li et al. 2020). Note that by combining ACR and AdvR, we can make attribute information captured *fully* and *exclusively* in a. To achieve AdvR, the encoder E is trained to generate the latent representation $z \triangleq E(x)$ so that z cannot be discriminated by the attribute classifier f_c, which is expressed by the following learning objective:

$$\max_{E} \min_{f_c} \mathcal{L}_{\text{AdvR}}(\theta_{\text{E}}, \theta_{f_c}) = -\mathbb{E}_{p(x)} \log f_c(E(x)) \tag{5}$$

Since AdvR can be imbalanced if the number of samples of each attribute value differs largely, an extension of AdvR is to treat different attribute values with equal weight (Shen et al. 2017):

$$\begin{aligned} \max_{E} \min_{f_c} \mathcal{L}_{\text{AAE}}(\theta_{\text{E}}, \theta_{f_c}) = &-\mathbb{E}_{p(x)} \Big[\log f_c(E(x)) \Big] \\ &- \mathbb{E}_{p(x')} \Big[\log(1 - f_c(E(x'))) \Big] \end{aligned} \tag{6}$$

Note that $p(x)$ is the distribution of sentences of one attribute, and $p(x')$ is the distribution of sentences of the other attribute.

Adversarial Learning on Outputs (AdvO). Apart from AdvR that adversarially learn the latent representations, we can also use AdvO to perform adversarial training on the outputs, to make them undistinguishable from the real data (Shen et al. 2017; Logeswaran, Lee, and Bengio 2018; Tian, Hu, and Yu 2018). Specifically, for each attribute a_i, we train a classifier $f_c^{(i)}$ to distinguish between true x_i from the mono-style corpus of attribute a_i, and the generated sentence $\widehat{x_i} \triangleq G(E(x_k); a_i)$, where $k \neq i$, which aims to have the attribute a_i. The loss function is

$$\begin{aligned}\max_{E,G} \min_{f_c^{(i)}} \mathcal{L}_{\text{AdvO}}^{(i)}(\theta_{\text{E}}, \theta_{\text{G}}, \theta_{f_c^{(i)}}) = &-\mathbb{E}_{p(x_i)}\Big[\log f_c^{(i)}(x_i)\Big] \\ &- \mathbb{E}_{p(x_k)}\Big[\log(1 - f_c^{(i)}(G(E(x_k); a_i)))\Big]\end{aligned} \tag{7}$$

In the training process, usually we first optimize all attribute classifiers $f_c^{(i)}$, and then train the encoder, generator, and the attribute classifiers together by optimizing the sum the all AdvO training losses:

$$\max_{E,G} \sum_{i}^{|\mathbb{A}|} \min_{f_c^{(i)}} \mathcal{L}_{\text{AdvO}}^{(i)}(\theta_{\text{E}}, \theta_{\text{G}}, \theta_{f_c^{(i)}}) \tag{8}$$

Note that in order to propagate the gradients, it is feasible to use the sequence of hidden states in the generator instead of discrete text for $G(E(x_k); a_i)$ (Shen et al. 2017).

Language Modeling on Outputs (LMO). The above AdvO learns classifiers to distinguish between true samples and generated samples. Such discriminative classification can be alternatively achieved by generative language modeling, namely, LM_i for each mono-style corpus with the attribute a_i (Yang et al. 2018). Specifically, the training objective for each attribute is

$$\mathcal{L}_{\text{LMO}}^{(i)}(\theta_{\text{E}}, \theta_{\text{G}}, \theta_{\text{LM}_i}) = -\mathbb{E}_{p(x_i)}\Big[\log p_{\text{LM}_i}(x_i)\Big] + \gamma \mathbb{E}_{p(z_k)}\Big[\log p_{\text{LM}_i}(G(E(x_k); a_i))\Big] \tag{9}$$

where γ is a hyperparameter to weight the two terms. The total training objective sums over the losses of all attributes:

$$\max_{E,G} \sum_{i}^{|\mathbb{A}|} \min_{\text{LM}^{(i)}} \mathcal{L}_{\text{LMO}}^{(i)}(\theta_{\text{E}}, \theta_{\text{G}}, \theta_{\text{LM}_i}) \tag{10}$$

5.1.3.2 Content-Oriented Losses

The style-oriented losses introduced above ensures the attribute information to be contained in a, but not necessarily putting constraints on the style-independent semantics z. To learn the attribute-independent information fully and exclusively in z, the following *content-oriented losses* are proposed:

Cycle Reconstruction (Cycle). The cycle reconstruction loss (dos Santos, Melnyk, and Padhi 2018; Logeswaran, Lee, and Bengio 2018; Luo et al. 2019; Dai et al. 2019; Yi et al. 2020; Huang et al. 2020) first encodes a sentence x to its latent representation $z \triangleq E(x)$, and then feeds z to the generator G to obtain the generated sentence $G(z)$. Since the

alignment of the input and the generated sentence is to preserve attribute-independent semantic information, the generator can be conditioned on any attribute, namely, a or a'. The cycle loss constrains the output $\widehat{x'}$ to align with the input x (and, similarly, the output $\widehat{x}$ to align with the input x') so that the content information can be preserved:

$$\mathcal{L}_{\text{Cycle}}(\theta_E, \theta_G) = -\mathbb{E}_{p(x)}\Big[\log p_G(x|E(x))\Big] - \mathbb{E}_{p(x')}\Big[\log p_G(x'|E(x'))\Big] \tag{11}$$

One way to train the above cycle loss is by reinforcement learning as done by Luo et al. (2019), who use the loss function as a negative for content preservation.

Bag-of-Words Overlap (BoW). To approximately measure content preservation, bag-of-words (BoW) features are used by John et al. (2019), Bao et al. (2019). To focus on content information only, John et al. (2019) exclude stopwords and style-specific words.

Let us denote the vocabulary set as $\mathbb{V}$. We first predict the distribution of BoW features $q_{\text{BoW}}(z)$ of the latent representation z using softmax on the $1 \times |\mathbb{V}|$ BoW features. We then calculate the cross entropy loss of this BoW distribution $q_{\text{BoW}}(z)$ against the ground-truth BoW distribution $p_{\text{BoW}}(x)$ in the input sentence x. The BoW loss is formulated as follows:

$$\mathcal{L}_{\text{BoW}}(\theta_{\text{E}}, \theta_{q_{\text{BoW}}}) = -p_{\text{BoW}}(x) \log q_{\text{BoW}}(z) \tag{12}$$

Adversarial BoW Overlap (AdvBoW). BoW ensures the content to be *fully* captured in z. As a further step, we want to ensure that the content information is exclusively captured in z, namely, not contained in a at all, via the following AdvBow loss on a (John et al. 2019; Bao et al. 2019).

When disentangling z and a in the LRS framework, we train an adversarial classifier $q_{\text{BoW}}(a)$ to predict the BoW features given a by aligning it with the ground-truth BoW distribution $p_{\text{BoW}}(x)$, namely, minimizing

$$\mathcal{L}_{\text{AdvBoW}}(\theta_{\text{E}}, \theta_{q_{\text{BoW}}}) = -p_{\text{BoW}}(x) \log q_{\text{BoW}}(z) \tag{13}$$

The final min-max objective is

$$\max_{E} \min_{q_{\text{BoW}}} \mathcal{L}_{\text{AdvBoW}}(\theta_{\text{E}}, \theta_{q_{\text{BoW}}}) \tag{14}$$

Other Losses/Rewards. There are also other losses/rewards in recent work such as the noun overlap loss (Noun) (Tian, Hu, and Yu 2018), as well as rewards for semantics and fluency (Xu et al. 2018; Gong et al. 2019; Sancheti et al. 2020). We do not discuss them in much detail because they do not directly operate on the disentanglement of latent representations.

5.2 Prototype Editing

Despite a plethora of models that use end-to-end training of neural networks, the prototype-based text editing approach still attracts lots of attention, since the proposal of a pipeline method called *delete, retrieve*, and *generate* (Li et al. 2018).

Prototype editing is reminiscent of early word replacement methods used for TST, such as synonym matching using a style dictionary (Sheikha and Inkpen 2011),

WordNet (Khosmood and Levinson 2010; Mansoorizadeh et al. 2016), hand-crafted rules (Khosmood and Levinson 2008; Castro, Ortega, and Muñoz 2017), or using hypernyms and definitions to replace the style-carrying words (Karadzhov et al. 2017).

Featuring more controllability and interpretability, prototype editing builds an explicit pipeline for TST from x with attribute a to its counterpart x' with attribute a':

Step 1. Detect attribute markers of a in the input sentence x, and delete them, resulting in a content-only sentence (Section 5.2.1).
Step 2. Retrieve candidate attribute markers carrying the desired attribute a' (Section 5.2.2).
Step 3. Infill the sentence by adding new attribute markers and make sure the generated sentence is fluent (Section 5.2.3).

5.2.1 Attribute Marker Detection. Extracting attribute markers is a non-trivial NLP task. Traditional ways to do it involve first using tagging, parsing, and morphological analysis to select features, and then filtering by mutual information and chi-square testing. In recent deep learning pipelines, there are three major types of approaches to identify attribute markers: frequency-ratio methods, attention-based methods, and fusion methods.

Frequency-ratio methods calculate some statistics for each n-gram in the corpora. For example, Li et al. (2018) detect the attribute markers by calculating its relative frequency of co-occurrence with attribute a versus a', and those with frequencies higher than a threshold are considered the markers of a. Using a similar approach, Madaan et al. (2020) first calculate the ratio of mean TF-IDF between the two attribute corpora for each n-gram, then normalize this ratio across all possible n-grams, and finally mark those n-grams with a normalized ratio p higher than a pre-set threshold as attribute markers.

Attention-based methods train an attribute classifier using the attention mechanism (Bahdanau, Cho, and Bengio 2015), and consider words with attention weights higher than average as markers (Xu et al. 2018). For the architecture of the classifier, Zhang et al. (2018c) use LSTM, and Sudhakar, Upadhyay, and Maheswaran (2019) use a BERT classifier, where the BERT classifier has shown higher detection accuracy for the attribute markers.

Fusion methods combine the advantages of the above two methods. For example, Wu et al. (2019) prioritize the attribute markers predicted by frequency-ratio methods, and use attention-based methods as an auxiliary back up. One use case is when frequency-ratio methods fail to identify any attribute markers in a given sentence, they will use the attention-based methods as a secondary choice to generate attribute markers. Another case is to reduce false positives. To reduce the number of attribute markers that are wrongly recognized, Wu et al. (2019) set a threshold to filter out low-quality attribute markers by frequency-ratio methods, and in cases where all attribute markers are deleted, they use the markers predicted by attention-based methods.

There are still remaining limitations of the previous methods, such as imperfect accuracy of the attribute classifier, and unclear relation between attribute and attention scores. Hence, Lee (2020) propose word importance scoring, similar to what is used by Jin et al. (2020b) for adversarial paraphrasing, to measure how important a token is to the attribute by the difference in the attribute probability of the original sentence and that after deleting a token.

5.2.2 Target Attribute Retriever. After deleting the attribute markers $\text{Marker}_a(x)$ of the sentence x with attribute a, we need to find a counterpart attribute marker $\text{Marker}_{a'}(x')$ from another sentence x' carrying a different attribute a'. Denote the sentence template with all attribute markers deleted as $\text{Template}(x) \triangleq x \backslash \text{Marker}_a(x)$. Similarly, the template of the sentence x' is $\text{Template}(x') \triangleq x' \backslash \text{Marker}_{a'}(x')$. A common approach is to find the counterpart attribute marker by its context, because the templates of the original attribute and its counter attribute marker should be similar. Specifically, we first match a template $\text{Template}(x)$ with the most similar template $\text{Template}(x')$ in the opposite attribute corpus, and then identify the attribute markers $\text{Marker}_a(x)$ and $\text{Marker}_{a'}(x')$ as counterparts of each other. To match templates with their counterparts, most previous works find the nearest neighbors by the cosine similarity of sentence embeddings. Commonly used sentence embeddings include TF-IDF as used in Li et al. (2018) and Sudhakar, Upadhyay, and Maheswaran (2019), averaged GloVe embedding distance used in Li et al. (2018) and Sudhakar, Upadhyay, and Maheswaran (2019), and Universal Sentence Encoder (Cer et al. 2018) used in Sudhakar, Upadhyay, and Maheswaran (2019). Apart from sentence embeddings, Tran, Zhang, and Soleymani (2020) use Part-of-Speech templates to match several candidates in the opposite corpus, and conduct an exhaustive search to fill parts of the candidate sentences into the masked positions of the original attribute markers.

5.2.3 Generation from Prototypes. Li et al. (2018) and Sudhakar, Upadhyay, and Maheswaran (2019) feed the content-only sentence template and new attribute markers into a pretrained language model that rearranges them into a natural sentence. This infilling process can naturally be achieved by a masked language model (MLM) (Malmi, Severyn, and Rothe 2020). For example, Wu et al. (2019) use a MLM of the template conditioned on the target attribute, and this MLM is trained on an additional attribute classification loss using the model output and a fixed pre-trained attribute classifier. Because these generation practices are complicated, Madaan et al. (2020) propose a simpler way. They skip Step 2 that explicitly retrieves attribute candidates, and, instead, directly learn a generation model that only takes attribute-masked sentences as inputs. This generation model is trained on data where the attribute-carrying sentences x are paired with their templates $\text{Template}(x)$. Training on the pairs of $(\text{Template}(x), x)$ constructed in this way can make the model learn how to fill the masked sentence template with the target attribute a.

5.3 Pseudo-Parallel Corpus Construction

To provide more signals for training, it is also helpful to generate pseudo-parallel data for TST. Two major approaches are retrieval-based and generation-based methods.

Retrieval-Based Corpora Construction. One common way to construct pseudo-parallel data is through retrieval, namely, extracting aligned sentence pairs from two mono-style corpora. Jin et al. (2019) empirically observe that semantically similar sentences in the two mono-style corpora tend to be the attribute-transferred counterparts of each other. Hence, they construct the initial pseudo corpora by matching sentence pairs in the two attributed corpora according to the cosine similarity of pretrained sentence embeddings. Formally, for each sentence x, its pseudo counterpart $\widehat{x'}$ is its most similar sentence in the other attribute corpus X', namely, $\widehat{x'} = \text{argmax}_{x' \in X'} \text{Similarity}(x, x')$. This

approach is extended by Nikolov and Hahnloser (2019), who use large-scale hierarchical alignment to extract pseudo-parallel style transfer pairs. Such retrieval-based pseudo-parallel data construction is also useful for machine translation (Munteanu and Marcu 2005; Uszkoreit et al. 2010; Marie and Fujita 2017; Grégoire and Langlais 2018; Ren et al. 2020).

Generation-Based Corpora Construction. Another way is through generation, such as iterative back-translation (IBT) (Hoang et al. 2018). IBT is a widely used method in machine translation (Artetxe et al. 2018; Lample et al. 2018a, 2018b; Dou, Anastasopoulos, and Neubig 2020) that adopts an iterative process to generate pseudo-parallel corpora.

Before starting the iterative process, IBT needs to first initialize two style transfer models: $M_{a \to a'}$, which transfers from the attribute a to the other attribute a', and $M_{a' \to a}$, which transfers from a' to a. Then, in each iteration, it executes the following steps:

Step 1. Use the models to generate pseudo-parallel corpora. Specifically, $M_{a \to a'}(x)$ generates pseudo pairs $(x, \widehat{x'})$ for all $x \in X$, and $M_{a' \to a}(x')$ generates pairs of $(\widehat{x}, x')$ for all $x' \in X'$.

Step 2. Re-train these two style transfer models on the datasets generated by 1, that is, re-train $M_{a \to a'}(x)$ on $(\widehat{x}, x')$ pairs and $M_{a' \to a}(x')$ on $(\widehat{x'}, x)$ pairs.

For Step 1, in order to generate the initial pseudo-parallel corpora, a simple baseline is to randomly initialize the two models $M_{a \to a'}$ and $M_{a' \to a}$, and use them to translate the attribute of each sentence in $x \in X$ and $x' \in X'$. However, this simple initialization is subject to randomness and may not bootstrap well. Another way, adopted by Zhang et al. (2018d), borrows the idea from unsupervised machine translation (Lample et al. 2018a) that first learns an unsupervised word-to-word translation table between attribute a and a', and uses it to generate an initial pseudo-parallel corpora. Based on such initial corpora, they train initial style transfer models and bootstrap the IBT process. Another model, Iterative Matching and Translation (IMaT) (Jin et al. 2019), does not learn the word translation table, and instead trains the initial style transfer models on a retrieval-based pseudo-parallel corpora introduced in the *retrieval-based corpora construction* above.

For Step 2, during the iterative process, it is possible to encounter divergence, as there is no constraint to ensure that each iteration will produce better pseudo-parallel corpora than the previous iteration. One way to enhance the convergence of IBT is to add additional losses. For example, Zhang et al. (2018d) use the attribute classification loss ACO, as in Equation (3), to check whether the generated sentence by back-translation fits the desired attribute according to a pre-trained style classifier. Alternatively, IMaT (Jin et al. 2019) uses a checking mechanism instead of additional losses. At the end of each iteration, IMaT looks at all candidate pseudo-pairs of an original sentence, and uses WMD (Kusner et al. 2015) to select the sentence that has the desired attribute and is the closest to the original sentence.

6. Research Agenda

In this section, we will propose some potential directions for future TST research, including expanding the scope of styles (Section 6.1), improving the methodology (Section 6.2), loosening the style-specific data assumptions (Section 6.3), and improving evaluation metrics (Section 6.4).

6.1 Expanding the Scope of Styles

More Styles. Extending the list of styles for TST is one popular research direction. Existing research originally focused on styles such as simplification (Zhu, Bernhard, and Gurevych 2010), formality (Sheikha and Inkpen 2011), and sentiment transfer (Shen et al. 2017), while the recent two years have seen a richer set of styles such as politeness (Madaan et al. 2020), biasedness (Pryzant et al. 2020), medical text simplification (Cao et al. 2020), and so on.

Such extension of styles is driven by the advancement of TST methods, and also various downstream needs, such as persona-based dialog generation, customized text rewriting applications, and moderation of online text. Apart from the styles that have been researched as listed in Table 3, there are also many other new styles that can be interesting to conduct new research on, including but not limited to the following:

- Factual-to-empathetic transfer, to improve counseling dialogs (after the first version of this survey in 2020, we gladly found that this direction has now a preliminary exploration by Sharma et al. [2021])
- Non-native-to-native transfer (i.e., reformulating grammatical error correction with TST)
- Sentence disambiguation, to resolve nuance in text

More Difficult Forms of Style. Another direction is to explore more complicated forms of styles. As covered by this survey, the early work on deep learning-based TST explores relatively simple styles, such as verb tenses (Hu et al. 2017) and positive-vs.-negative Yelp reviews (Shen et al. 2017). In these tasks, each data point is one sentence with a clear, categorized style, and the entire dataset is in the same domain. Moreover, the existing datasets can decouple style and style-independent contents relatively well.

We propose that TST can potentially be extended into the following settings:

- Aspect-based style transfer (e.g., transferring the sentiment on one aspect but not the other aspects on aspect-based sentiment analysis data)
- Authorship transfer (which has tightly coupled style and content)
- Document-level style transfer (which includes discourse planning)
- Domain adaptive style transfer (which is preceded by Li et al. [2019])

Style Interwoven with Semantics. In some cases, it can be difficult or impossible to separate attributes from meaning, namely, the subject matter or the argument that the author wants to convey. One reason is that the subject that the author is going to write about can influence the choice of writing style. For example, science fiction writing can use the first person voice and fancy, flowery tone when describing a place. Another reason is that many stylistic devices such as allusion depend on content words.

Currently, it is a simplification of the problem setting to limit it to scenarios where the attribute and semantics can be approximately separated. For evaluation, so far researchers have allowed the human judges to decide the scores of transferred style strength and the content preservation.

In future work, it will be an interesting direction to address the more challenging scenarios where the style and semantics are interwoven.

6.2 Improving the Methodology on Non-Parallel Data

Because the majority of TST research focuses on non-parallel data, we discuss here its strengths and limitations.

6.2.1 Understanding the Strengths and Limitations of Existing Methods. To come up with improvement directions for TST methods, it is important to first investigate the strengths and limitations of existing methods. We analyze the three major streams of approaches for unsupervised TST in Table 6, including their strengths, weaknesses, and future directions.

Table 6
The strengths (+), weaknesses (−), and improvement directions (?) of the three mainstreams of TST methods on non-parallel data.

Method	Strengths & Weaknesses
Disentanglement	+ More profound in theoretical analysis, e.g., disentangled representation learning − Difficulties of training deep generative models (VAEs, GANs) for text − Hard to represent all styles as latent code − Computational cost rises with the number of styles to model
Prototype Editing	+ High BLEU scores due to large word preservation − Attribute marker detection step can fail if the style and semantics are confounded − The step target attribute retrieval by templates can fail if there are large rewrites for styles, e.g., Shakespearean English vs. modern English − Target attribute retrieval step has large complexity (quadratic to the number of sentences) − Large computational cost if there are many styles, each of which needs a pre-trained LM for the generation step ? Future work can enable matchings for syntactic variation ? Future work can use grammatical error correction to post-edit the output
Pseudo-Parallel Corpus Construction	+ Performance can approximate supervised model performance, if the pseudo-parallel data are of good quality − May fail for small corpora − May fail if the mono-style corpora do not have many samples with similar contents − For IBT, divergence is possible, and sometimes needs special designs to prevent it − For IBT, time complexity is high (due to iterative pseudo data generation) ? Improve the convergence of the IBT

Challenges for Disentanglement. Theoretically, although disentanglement is impossible without inductive biases or other forms of supervision (Locatello et al. 2019), disentanglement is achievable with some weak signals, such as only knowing how many factors have changed, but not which ones (Locatello et al. 2020).

In practice, some big challenges for disentanglement-based methods include, for example, the difficulty to train deep text generative models such as VAEs and GANs. Also, it is not easy to represent all styles as latent code. Moreover, if targeting multiple

styles, the computational complexity linearly increases with the number of styles to model.

Challenges for Prototype Editing. Prototype-editing approaches usually result in relatively high BLEU scores, partly because the output text largely overlaps with the input text. This line of methods is likely to perform well on tasks such as sentiment modification, for which it is easy to identify "attribute markers," and the input and output sentences share an attribute-independent template.

However, prototype editing cannot be applied to all types of style transfer tasks. The first step, attribute marker retrieval, might not work if the datasets have confounded style and contents, because they may lead to wrong extraction of attribute markers, such as some content words or artifacts which can also be used to distinguish the style-specific data.

The second step, target attribute retrieval by templates, will fail if there is too little word overlap between a sentence and its counterpart carrying another style. An example is the TST task to "Shakespearize" modern English. There is little lexical overlap between a Shakespearean sentence written in early modern English and its corresponding modern English expression. In such cases, the retrieval step is likely to fail, because there is a large number of rewrites between the two styles, and the template might be almost hollow. Moreover, this step is also computationally expensive, if there are a large number of sentences in the data (e.g., all Wikipedia text), since this step needs to calculate the pair-wise similarity among all available sentences across style-specific corpora.

The third step, generation from prototype, requires a separate pretrained LM for each style corpus. When there are multiple styles of interest (e.g., multiple persona), this will induce a large computational cost.

The last limitation of prototype editing is that it amplifies the intrinsic problem of using BLEU to evaluate TST (Problem 1, namely, the fact that simply copying the input can result in a high BLEU score, as introduced in Section 3.1). For the retrieval-based method, some can argue that there is some performance gain because this method in practice copies more expressions in the input sentence than other lines of methods.

As future study, there can be many interesting directions to explore, for example, investigating the performance of existing prototype editing models under a challenging dataset that reveals the above shortcomings, proposing new models to improve this line of approaches, and better evaluation methods for prototype editing models.

Challenges for Pseudo-Parallel Corpus Construction. The method to construct pseudo-parallel data can be effective, especially when the pseudo-parallel corpora resemble supervised data. The challenge is that this approach may not work if the non-parallel corpora do not have enough samples that can be matched to create the pseudo-parallel corpora, or when the IBT cannot bootstrap well or fails to converge. The time complexity for training IBT is also very high because it needs to iteratively generate pseudo-parallel corpus and re-train models. Interesting future directions can be reducing the computational cost, designing more effective bootstrapping, and improving the convergence of IBT.

6.2.2 Understanding the Evolution from Traditional NLG to Deep Learning Methods. Despite the exciting methodological revolution led by deep learning recently, we are also interested in the merging point of traditional computational linguistics and the deep learning techniques (Henderson 2020). Specific to the context of TST, we will introduce the

traditional NLG framework, and its impact on the current TST approaches, especially the prototype editing method.

Traditional NLG Framework. The traditional NLG framework stages sentence generation into the following steps (Reiter and Dale 1997):

1. Content determination (not applicable)
2. Discourse planning (not applicable)
3. Sentence aggregation
4. Lexicalization
5. Referring expression generation
6. Linguistic realization

The first two steps, content determination and discourse planning, are not applicable to most datasets because the current focus of TST is sentence-level and not discourse-level.

Among Steps 3 to 6, *sentence aggregation* groups necessary information into a single sentence, *lexicalization* chooses the right word to express the concepts generated by sentence aggregation, *referring expression generation* produces surface linguistic forms for domain entities, and *linguistic realization* edits the text so that it conforms to grammar, including syntax, morphology, and orthography. This framework is widely applied to NLG tasks (e.g., Zue and Glass 2000; Mani 2001; McTear 2002; Gatt and Reiter 2009; Androutsopoulos and Malakasiotis 2010).

Re-Viewing Prototype-Based TST. Among the approaches introduced so far, the most relevant for the traditional NLG is the prototype-based text editing, which has been introduced in Section 5.2.

Using the language of the traditional NLG framework, the prototype-based techniques can be viewed as a combination of *sentence aggregation*, *lexicalization*, and *linguistic realization*. Specifically, prototype-based techniques first prepare an attribute-free sentence template, and supply it with candidate attribute markers that carry the desired attribute, both of which are *sentence aggregation*. Then, using language models to infill the prototype with the correct expressions corresponds to *lexicalization* and *linguistic realization*. Note that the existing TST systems do not explicitly deal with *referring expression generation* (e.g., generating co-references), leaving it to be handled by language models.

Meeting Point of Traditional and New Methods. Viewing prototype-based editing as a merging point where traditional, controllable framework meets deep learning models, we can see that it takes advantage of the powerful deep learning models and the interpretable pipeline of the traditional NLG. There are several advantages in merging the traditional NLG with the deep learning models. First, sentence planning–like steps make the generated contents more controllable. For example, the template of the original sentence is saved, and the counterpart attributes can also be explicitly retrieved, as a preparation for the final rewriting. Such a controllable, white-box approach can be easy to tune, debug, and improve. The accuracy of attribute marker extraction, for example, is constantly improving across literature (Sudhakar, Upadhyay, and Maheswaran 2019) and different ways to extract attribute markers can be easily fused (Wu et al. 2019).

Second, sentence planning–like steps ensure the truthfulness of information. As most content words are kept and no additional information is hallucinated by the black-box neural networks, we can better ensure that the information of the attribute-transferred output is consistent with the original input.

6.2.3 Inspiration from Tasks with Similar Nature. An additional perspective that can inspire new methodological innovation is insights from other tasks that share a similar nature as TST. We will introduce in this section several closely related tasks, including machine translation, image style transfer, style-conditioned language modeling, counterfactual story rewriting, contrastive text generation, and prototype-based text editing.

Machine Translation. The problem settings of machine translation and TST share much in common: The source and target language in machine translation is analogous to the original and desired attribute, a and a', respectively. The major difference is that in NMT, the source and target corpora are in completely different languages, which have almost disjoint word vocabulary, whereas in TST, the input and output are in the same language, and the model is usually encouraged to copy most content words from input such as the BoW loss introduced in Section 5.1.3.2. Some TST works have been inspired by MT, such as the pseudo-parallel construction (Nikolov and Hahnloser 2019; Zhang et al. 2018d), and in the future there may be more interesting intersections.

Data-to-Text Generation. Data-to-text generation is another potential domain that can draw inspiration from and to TST. The data-to-text generation task is to generate textual descriptions from structured data such as tables (Wiseman, Shieber, and Rush 2017; Parikh et al. 2020), meaning representations (Novikova, Dusek, and Rieser 2017), or Resource Description Framework triples (Gardent et al. 2017; Ferreira et al. 2020). With the recent rise of pretrained seq2seq models for transfer learning (Raffel et al. 2020), it is common to formulate data-to-text as a seq2seq task by serializing the structured data into a sequence (Kale and Rastogi 2020; Ribeiro et al. 2020; Guo et al. 2020). Then data-to-text generation can be seen as a special form of TST from structured information to text. This potential connection has not yet been investigated but worth exploring.

Neural Style Transfer. Neural style transfer first originates in image style transfer (Gatys, Ecker, and Bethge 2016), and its disentanglement ideas inspired some early TST research (Shen et al. 2017). The difference between image style transfer and TST is that, for images, it is feasible to disentangle the explicit representation of the image texture as the gram matrix of image neural feature vectors, but for text, styles do not have such an explicit representation, but more abstract attributes. Besides this difference, many other aspects of style transfer research can have shared nature. Note that there are style transfer works across different modalities, including images (Gatys, Ecker, and Bethge 2016; Zhu et al. 2017; Chen et al. 2017b), text, voice (Gao, Singh, and Raj 2018; Qian et al. 2019; Yuan et al. 2021), handwriting (Azadi et al. 2018; Zhang and Liu 2013), and videos (Ruder, Dosovitskiy, and Brox 2016; Chen et al. 2017a). Many new advances in one style transfer field can inspire another style transfer field. For example, image style transfer has been used as a way for data augmentation (Zheng et al. 2019; Jackson et al. 2019) and adversarial attack (Xu et al. 2020), but TST has not yet been applied for such usage.

Style-Conditioned Language Modeling. Different from language modeling that learns how to generate general natural language text, conditional language modeling learns how to generate text given a condition, such as some context, or a control code (Pfaff 1979;

Poplack 2000). Recent advances of conditional language models (Keskar et al. 2019; Dathathri et al. 2020) also include text generation conditioned on a style token, such as positive or negative. Possible conditions include author style (Syed et al. 2020), speaker identity, persona and emotion (Li et al. 2016), genre, and attributes derived from text, topics, and sentiment (Ficler and Goldberg 2017). They are currently limited to a small set of pre-defined "condition" tokens and can only generate from scratch a sentence, but are not yet able to be conditioned on an original sentence for style rewriting. The interesting finding in this research direction is that it can make good use of a pretrained LM and just do some light-weight inference techniques to generate style-conditioned text, so perhaps such approaches can inspire future TST methods and reduce the carbon footprints of training TST models from scratch.

Counterfactual Story Rewriting. Counterfactual story rewriting aims to learn a new event sequence in the presence of a perturbation of a previous event (i.e., counterfactual condition) (Goodman 1947; Starr 2019). Qin et al. (2019) propose the first dataset, each sample of which takes an originally five-sentence story, and changes the event in the second sentence to a new, counterfactual event. The task is to generate the last three sentences of the story based on the newly altered second sentence that initiates the story. The criteria of the counterfactual story rewriting include relevance with the first two sentences, and minimal edits from the original story ending. This line of research is relatively difficult to directly apply to TST, because its motivation and dataset nature is different from the general TST, and more importantly, this task is not conditioned on a predefined categorized style token, but the free-form textual story beginning.

Contrastive Text Generation. As neural network-based NLP models more easily learn spurious statistical correlations in the data rather than achieve robust understanding (Jia and Liang 2017), there is recent work to construct auxillary datasets composed of near-misses of the original data. For example, Gardner et al. (2020) ask crowdsource workers to rewrite the input of the task with minimal changes but matching a different target label. To alleviate expensive human labor, Xing et al. (2020) develop an automatic text editing approach to generate contrast set for aspect-based sentiment analysis. The difference between contrastive text generation and TST is that the former does not require content preservation but mainly aims to construct a slightly textually different input that can result in a change of the ground-truth output, to test the model robustness. So the two tasks are not completely the same, although they have some intersections that might inspire future work, such as aspect-based style transfer suggested in Section 6.1.

Prototype-Based Text Editing. Prototype editing is not unique in TST, but also widely used in other NLP tasks. Knowing the new advances in prototype editing for other tasks can potentially inspire new method innovations in TST. Guu et al. (2018) first proposes the protype editing approach to improve LM by first sampling a lexically similar sentence prototype and then editing it using variational encoder and decoders. This prototype-and-then-edit approach can also be seen in summarization (Wang, Quan, and Wang 2019), machine translation (Cao and Xiong 2018; Wu, Wang, and Wang 2019; Gu et al. 2018; Zhang et al. 2018a; Bulté and Tezcan 2019), conversation generation (Weston, Dinan, and Miller 2018; Cai et al. 2019), code generation (Hashimoto et al. 2018), and question answering (Lewis et al. 2020). As an extension to the retrieve and edit steps, Hossain, Ghazvininejad, and Zettlemoyer (2020) use an ensemble approach to retrieve a set of relevant prototypes, edit, and finally rerank to pick the best output for machine translation. Such extension can also be potentially applied to TST.

6.3 Loosening the Style-Specific Dataset Assumptions

A common assumption for most deep learning-based TST works, as mentioned in Section 2.1, is the availability of style-specific corpora for each style of interest, either parallel or non-parallel. This assumption can potentially be loosened in two ways.

Linguistic Styles with No Matched Data. Because there are various concerns raised by the data-driven definition of style as described in Section 2.1, a potentially good research direction is to bring back the linguistic definition of style, and thus remove some of the concerns associated with large datasets. Several methods can be a potential fit for this: prompt design (Li and Liang 2021; Qin and Eisner 2021; Scao and Rush 2021) that passes a prompt to GPT (Radford et al. 2019; Brown et al. 2020) to obtain a style-transferred text; style-specific template design; or use templates to first generate synthetic data and make models learn from the synthetic data. Prompt design is not yet investigated as a direction for TST research, but it is an interesting direction to explore.

Distinguishing Styles from a Mixed Corpus. It might also be possible to distinguish styles from a mixed corpus with no style labels. For example, Riley et al. (2021) learn a style vector space from text; Xu, Cheung, and Cao (2020) use unsupervised representation learning to separate the style and contents from a mixed corpus of unspecified styles; and Guo et al. (2021) use cycle training with a conditional variational auto-encoder to unsupervisedly learn to express the same semantics through different styles. Theoretically, although disentanglement is impossible without inductive biases or other forms of supervision (Locatello et al. 2019), disentanglement is achievable with some weak signals, such as only knowing how many factors have changed, but not which ones (Locatello et al. 2020). A more advanced direction can be emergent styles (Kang, Wang, and de Melo 2020), since styles can be evolving, for example across dialog turns.

6.4 Improving Evaluation Metrics

There has been a lot of attention to the problems of evaluation metrics of TST and potential improvements (Pang and Gimpel 2019; Tikhonov and Yamshchikov 2018; Mir et al. 2019; Fu et al. 2019; Pang 2019; Yamshchikov et al. 2021; Jafaritazehjani et al. 2020). Recently, Gehrmann et al. (2021) have proposed a new framework that is a live environment to evaluate NLG in a principled and reproducible manner. Apart from the existing scoring methods, future work can also make use of linguistic rules such as a checklist to evaluate what capabilities the TST model has achieved. For example, there can be a checklist for formality transfer according to existing style guidelines, such as the APA style guide (American Psychological Association 2020). Such a checklist-based evaluation can make the performance of black-box deep learning models more interpretable, and also allow for more insightful error analysis.

7. Expanding the Impact of TST

In this last section of this survey, we highlight several directions to expand the impact of TST. First, TST can be used to help other NLP tasks such as paraphrasing, data augmentation, and adversarial robustness probing (Section 7.1). Moreover, many specialized downstream tasks can be achieved with the help of TST, such as persona-consistent dialog generation, attractive headline generation, style-specific machine translation,

and anonymization (Section 7.2). Last but not least, we overview the ethical impacts that are important to take into consideration for future development of TST (Section 7.3).

7.1 Connecting TST to More NLP Tasks

TST can be applied to other important NLP tasks, such as paraphrase generation, data augmentation, and adversarial robustness probing.

Paraphrase Generation. Paraphrase generation is to express the same information in alternative ways (Madnani and Dorr 2010). The nature of paraphrasing shares a lot in common with TST, which is to transfer the style of text while preserving the content. One of the common ways of paraphrasing is syntactic variation, such as "X wrote Y.", "Y was written by X.", and "X is the writer of Y." (Androutsopoulos and Malakasiotis 2010). Besides syntactic variation, it also makes sense to include stylistic variation as a form of paraphrases, which means that the linguistic style transfer (not the content preference transfer in Table 3) can be regarded as a subset of paraphrasing. The caution here is that if the paraphrasing is for a downstream task, researchers should first check if the downstream task is compatible with the used styles. For example, dialog generation may be sensitive to all linguistic styles, whereas summarization can allow linguistic style-varied paraphrases in the dataset.

There are three implications of this connection of TST and paraphrase generation. First, many trained TST models can be borrowed for paraphrasing, such as formality transfer and simplification. A second connection is that the method innovations proposed in the two fields can inspire each other. For example, Krishna, Wieting, and Iyyer (2020) formulate style transfer as a paraphrasing task. Thirdly, the evaluation metrics of the two tasks can also inspire each other. For example, Yamshchikov et al. (2021) associate the semantic similarity metrics for two tasks.

Data Augmentation. Data augmentation generates text similar to the existing training data so that the model can have larger training data. TST is a good method for data augmentation because TST can produce text with different styles but the same meaning. Image style transfer has already been used for data augmentation (Zheng et al. 2019; Jackson et al. 2019), so it can be interesting to see future works also apply TST for data augmentation.

Adversarial Robustness Probing. Another use of style transferred text is adversarial robustness probing. For example, styles that are task-agnostic can be used for general adversarial attack (e.g., politeness transfer to probe sentiment classification robustness) (Jin et al. 2020b), while the styles that can change the task output can be used to construct contrast sets (e.g., sentiment transfer to probe sentiment classification robustness) (Xing et al. 2020). Xu et al. (2020) apply image style transfer to adversarial attack, and future research can also explore the use of TST in the two ways suggested above.

7.2 Connecting TST to More Specialized Applications

TST can be applied not only to other NLP tasks as introduced in the previous section, but also can be helpful for specialized downstream applications. In practice, when applying NLP models, it is important to customize for some specific needs, such as generating dialog with a consistent persona, writing headlines that are attractive and engaging,

making machine translation models adapt to different styles, and anonymizing the user identity by obfuscating the style.

Persona-Consistent Dialog Generation. A useful downstream application of TST is persona-consistent dialog generation (Li et al. 2016; Zhang et al. 2018b; Shuster et al. 2020). Because conversational agents directly interact with users, there is a strong demand for human-like dialog generation. Previously, this has been done by encoding speaker traits into a vector and the conversation is then conditioned on this vector (Li et al. 2016). As future work, TST can also be used as part of the pipeline of persona-based dialog generation, where the persona can be categorized into distinctive style types, and then the generated text can be post-processed by a style transfer model.

Attractive Headline Generation. In journalism writing, it is crucial to generate engaging headlines. Jin et al. (2020a) first use TST to generate eye-catchy headlines with three different styles: humorous, romantic, and clickbaity styles. Li et al. (2021) follow this direction and propose a disentanglement-based model to generate attractive headlines for Chinese news.

Style-Specific Machine Translation. In machine translation, it is useful to have an additional control of the style for the translated text. Commonly used styles for TST in machine translation are politeness (Sennrich, Haddow, and Birch 2016a) and formality (Niu, Martindale, and Carpuat 2017; Wu, Wang, and Liu 2020). For example, Wu, Wang, and Liu (2020) translate from informal Chinese to formal English.

Anonymization. TST can also be used for anonymization, which is an important way to protect user privacy, especially since there are ongoing heated discussions of ethics in the AI community. Many concerns have been raised about the discriminative task of author profiling, which can mine the demographic identities of the author of a writing, even including privacy-invading properties such as gender and age (Schler et al. 2006). As a potential solution, TST can be applied to alter the text and obfuscate the real identity of the users (Reddy and Knight 2016; Gröndahl and Asokan 2020).

7.3 Ethical Implications of TST

Recently, there is more and more attention being paid to the ethics concerns associated with AI research. We discuss in the following two ethics considerations: (1) social impact of TST applications, and (2) data privacy problem of TST.

Fields that involve human subjects or direct application to humans work under a set of core principles and guidelines (Beauchamp, Childress et al. 2001). Before initiating a research project, responsible research bodies use these principles as a ruler to judge whether the research is ethically correct to start. NLP research and applications, including TST, that directly involve human users, is regulated under a central regulatory board, the Institutional Review Board (IRB). We also provide several guidelines below to avoid ethical misconduct in future publications on TST.

7.3.1 Social Impact of TST Applications. Technologies can have unintended negative consequences (Hovy and Spruit 2016). For example, TST can facilitate the automation of intelligent assistants with designed attributes, but can also be used to create fake text or fraud.

Thus, inventors of a technology should beware how other people very probably adopt this technology for their own incentives. For TST, because it has a wide range of subtasks and applications, we examine each of them with the following two questions:

- Who will benefit from such a technology?
- Who will be harmed by such a technology?

Although many ethics issues are debatable, we try to categorize the text attribute tasks into three ethics levels: those with beneficial impacts, neutral impacts, and dual use.

Beneficial Impact. An important direction of NLP for social good is to fight against abusive online text. TST can serve as a very helpful tool as it can be used to transfer malicious text to normal language. Shades of abusive language include hate speech, offensive language, sexist and racist language, aggression, profanity, cyberbullying, harassment, trolling, and toxic language (Waseem et al. 2017). There is also other negative text such as propaganda (Bernays 2005; Carey 1997), and others. It is widely known that malicious text is harmful to people. For example, research shows that cyberbullying victims tend to have more stress and suicidal ideation (Kowalski et al. 2014), and also detachment from family and offline victimization (Oksanen et al. 2014). There are more and more efforts put into combating toxic language, such as 30K content moderators that Facebook and Instagram employ (Harrison 2019). Therefore, the automatic malicious-to-normal language transfer can be a helpful intelligent assistant to address such needs. Apart from purifying malicious text on social media, it can also be used on social chatbots to make sure there is no bad content in the language they generate (Roller et al. 2021).

Neutral Impact. Most TST tasks are neutral. For example, informal-to-formal transfer can be used as a writing assistant to help make writing more professional, and formal-to-informal transfer can tune the tone of bots to be more casual. Most applications to customize the persona of bots are also neutral with regard to their societal impact.

Dual Use. Besides positive and neutral applications, there are, unfortunately, several TST tasks that are double-edged swords. For example, take one of the most popular TST tasks, sentiment modification; although it can be used to change intelligent assistants or robots from a negative to positive mood (which is unlikely to harm any parties), the vast majority of research applies this technology to manipulate the polarity of reviews, such as Yelp (Shen et al. 2017) and Amazon reviews (He and McAuley 2016). This leads to a setting where a negative restaurant review is changed to a positive comment, or vice versa, with debatable ethics. Such a technique can be used as a cheating method for the commercial body to polish its reviews, or harm the reputation of their competitors. Once this technology is used, it will automatically manipulate the online text to contain polarity that the model owner desires. Hence, we suggest the research community raise serious concern against the review sentiment modification task.

Another task, political slant transfer, may induce concerns within some specific context. For example, social bots (i.e., autonomous bots on social media, such as Twitter bots and Facebook bots) are a big problem in the United States, even playing a significant role in the 2016 U.S. presidential election (Bessi and Ferrara 2016; Shao et al. 2018). It is reported that at least 400,000 bots were responsible for about 19% of the total

Tweets. Social bots usually target to advocate certain ideas, supporting campaigns, or aggregating other sources either by acting as a "follower" and/or gathering followers itself. So the political slant transfer task, which transfers the tone and content between Republican comments and Democratic ones, are highly sensitive and may face the risk of being used on social bots to manipulate political views of the mass.

Some more arguable ones are male-to-female tone transfer, which can be potentially used for identity deception. The cheater can create an online account and pretend to be an attractive young woman. There is also the reversed direction (female-to-male tone transfer), which can be used for applications such as authorship obfuscation (Shetty, Schiele, and Fritz 2018), anonymizing the author attributes by hiding the gender of a female author by re-synthesizing the text to use male textual attributes.

7.3.2 Data Privacy Issues for TST. Another ethics concern is the use of data in research practice. Researchers should not overmine user data, such as demographic identities. Such data privacy widely exists in the data science community as a whole, and there have been many ethics discussions (Tse et al. 2015; Russell, Dewey, and Tegmark 2015).

The TST task needs data containing *some* attributes along with the text content. Although it is acceptable to use ratings of reviews that are classified as positive or negative, user attributes are sensitive, including the gender of the user's account (Prabhumoye et al. 2018), and age (Lample et al. 2019). The collection and potential use of such sensitive user attributes can have implications that need to be carefully considered.

8. Conclusion

This article presented a comprehensive review of TST with deep learning methods. We have surveyed recent research efforts in TST and developed schemes to categorize and distill the existing literature. This survey has covered the task formulation, evaluation metrics, and methods on parallel and non-parallel data. We also discussed several important topics in the research agenda of TST, and how to expand the impact of TST to other tasks and applications, including ethical considerations. This survey provides a reference for future researchers working on TST.

Acknowledgments

We thank Qipeng Guo for his insightful discussions and the anonymous reviewers for their constructive suggestions.

References

American Psychological Association. 2020. *Publication Manual*, 7th ed. American Psychological Association Washington, DC.

Androutsopoulos, Ion and Prodromos Malakasiotis. 2010. A survey of paraphrasing and textual entailment methods. *Journal of Artificial Intelligence Research*, 38:135–187. https://doi.org/10.1613/jair.2985

Argamon, Shlomo, Moshe Koppel, Jonathan Fine, and Anat Rachel Shimoni. 2003. Gender, genre, and writing style in formal written texts. *Text & Talk*, 23(3):321–346.

Artetxe, Mikel, Gorka Labaka, Eneko Agirre, and Kyunghyun Cho. 2018. Unsupervised neural machine translation. In *6th International Conference on Learning Representations, ICLR 2018, Conference Track Proceedings*, OpenReview.net. https://doi.org/10.18653/v1/D18-1399

Azadi, Samaneh, Matthew Fisher, Vladimir G. Kim, Zhaowen Wang, Eli Shechtman, and Trevor Darrell. 2018. Multi-content GAN for few-shot font style transfer. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018*, pages 7564–7573, IEEE Computer Society. https://doi.org/10.1109/CVPR.2018.00789

Bahdanau, Dzmitry, 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, Conference Track Proceedings*.

Banerjee, Satanjeev and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72.

Bao, Yu, Hao Zhou, Shujian Huang, Lei Li, Lili Mou, Olga Vechtomova, Xin-yu Dai, and Jiajun Chen. 2019. Generating sentences from disentangled syntactic and semantic spaces. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6008–6019. https://doi.org/10.18653/v1/P19-1602

Bateman, John A. and Cecile Paris. 1989. Phrasing a text in terms the user can understand. In *IJCAI*, pages 1511–1517.

Beauchamp, Tom L., James F. Childress, et al. 2001. *Principles of Biomedical Ethics*, Oxford University Press.

Belz, Anja. 2008. Automatic generation of weather forecast texts using comprehensive probabilistic generation-space models. *Natural Language Engineering*, 14(4):431–455. https://doi.org/10.1017/S1351324907004664

Belz, Anya, Shubham Agarwal, Anastasia Shimorina, and Ehud Reiter. 2020. ReproGen: Proposal for a shared task on reproducibility of human evaluations in NLG. In *Proceedings of the 13th International Conference on Natural Language Generation*, pages 232–236.

den Bercken, Laurens Van, Robert-Jan Sips, and Christoph Lofi. 2019. Evaluating neural text simplification in the medical domain. In *The World Wide Web Conference, WWW 2019*, pages 3286–3292.

Bernays, Edward L. 2005. *Propaganda*, Ig Publishing.

Bessi, Alessandro and Emilio Ferrara. 2016. Social bots distort the 2016 US presidential election online discussion. *First Monday*, 21(11-7). https://doi.org/10.5210/fm.v21i11.7090

Boulis, Constantinos and Mari Ostendorf. 2005. A quantitative analysis of lexical differences between genders in telephone conversations. In *ACL 2005, 43rd Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference*, pages 435–442, The Association for Computer Linguistics. https://doi.org/10.3115/1219840.1219894

Briakou, Eleftheria, Sweta Agrawal, Ke Zhang, Joel Tetreault, and Marine Carpuat. 2021a. A review of human evaluation for style transfer. In *Proceedings of the 1st Workshop on Natural Language Generation, Evaluation, and Metrics (GEM 2021)*, pages 58–67. https://doi.org/10.18653/v1/2021.gem-1.6

Briakou, Eleftheria, Di Lu, Ke Zhang, and Joel Tetreault. 2021b. Olá, bonjour, salve! XFORMAL: A benchmark for multilingual formality style transfer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3199–3216. https://aclanthology.org/2021.naacl-main.256

Brown, Tom B., Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020*, pages 1877–1901.

Bulté, Bram and Arda Tezcan. 2019. Neural fuzzy repair: Integrating fuzzy matches into neural machine translation. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019*, pages 1800–1809. https://doi.org/10.18653/v1/P19-1175

Cai, Deng, Yan Wang, Wei Bi, Zhaopeng Tu, Xiaojiang Liu, Wai Lam, and Shuming Shi. 2019. Skeleton-to-response: Dialogue generation guided by retrieval memory. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Volume 1 (Long and Short Papers)*, pages 1219–1228. https://doi.org/10.18653/v1/N19-1124

Cao, Qian and Deyi Xiong. 2018. Encoding gated translation memory into neural machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*,

pages 3042–3047. https://doi.org/10.18653/v1/d18-1340

Cao, Yixin, Ruihao Shui, Liangming Pan, Min-Yen Kan, Zhiyuan Liu, and Tat-Seng Chua. 2020. Expertise style transfer: A new task towards better communication between experts and laymen. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1061–1071. https://www.aclweb.org/anthology/2020.acl-main.100

Carey, Alex. 1997. *Taking the Risk Out of Democracy: Corporate Propaganda versus Freedom and Liberty*, University of Illinois Press.

Carlson, Keith, Allen Riddell, and Daniel Rockmore. 2018. Evaluating prose style transfer with the bible. *Royal Society Open Science*, 5(10):171920. https://doi.org/10.1098/rsos.171920, PubMed: 30473797

Castro, Daniel, Reynier Ortega, and Rafael Muñoz. 2017. Author masking by sentence transformation—notebook for PAN at CLEF 2017. In *CLEF 2017 Evaluation Labs and Workshop–Working Notes Papers*, pages 11–14.

Cer, Daniel, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Brian Strope, and Ray Kurzweil. 2018. Universal sentence encoder for English. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, EMNLP 2018: System Demonstrations*, pages 169–174. https://doi.org/10.18653/v1/d18-2029

Chakrabarty, Tuhin, Smaranda Muresan, and Nanyun Peng. 2020. Generating similes effortlessly like a pro: A style transfer approach for simile generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020*, pages 6455–6469. https://doi.org/10.18653/v1/2020.emnlp-main.524

Chen, David L. and William B. Dolan. 2011. Collecting highly parallel data for paraphrase evaluation. In *49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference*, pages 190–200.

Chen, Dongdong, Jing Liao, Lu Yuan, Nenghai Yu, and Gang Hua. 2017a. Coherent online video style transfer. In *IEEE International Conference on Computer Vision, ICCV 2017*, pages 1114–1123. https://doi.org/10.1109/ICCV.2017.126

Chen, Dongdong, Lu Yuan, Jing Liao, Nenghai Yu, and Gang Hua. 2017b. StyleBank: An explicit representation for neural image style transfer. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, pages 2770–2779. https://doi.org/10.1109/CVPR.2017.296

Cho, Kyunghyun, Bart van Merrienboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the properties of neural machine translation: Encoder-decoder approaches. In *Proceedings of SSST@EMNLP 2014, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*, pages 103–111. https://doi.org/10.3115/v1/W14-4012

Cohen, Jacob. 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37–46. https://doi.org/10.1177/001316446002000104

Dai, Ning, Jianze Liang, Xipeng Qiu, and Xuanjing Huang. 2019. Style transformer: Unpaired text style transfer without disentangled latent representation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5997–6007. https://doi.org/10.18653/v1/P19-1601

Dathathri, Sumanth, Andrea Madotto, Janice Lan, Jane Hung, Eric Frank, Piero Molino, Jason Yosinski, and Rosanne Liu. 2020. Plug and play language models: A simple approach to controlled text generation. In *8th International Conference on Learning Representations, ICLR 2020*, OpenReview.net.

Devlin, Jacob, 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, NAACL-HLT 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. https://doi.org/10.18653/v1/n19-1423

dos Santos, Cícero Nogueira, Igor Melnyk, and Inkit Padhi. 2018. Fighting offensive language on social media with unsupervised text style transfer. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Volume 2: Short Papers*, pages 189–194.

Dou, Zi-Yi, Antonios Anastasopoulos, and Graham Neubig. 2020. Dynamic data selection and weighting for iterative back-translation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020*, pages 5894–5904. https://doi.org/10.18653/v1/2020.emnlp-main.475

Ferreira, Thiago, Claire Gardent, Nikolai Ilinykh, Chris van der Lee, Simon Mille, Diego Moussallem, and Anastasia Shimorina. 2020. The 2020 bilingual, bi-directional WebNLG+ shared task overview and evaluation results (WebNLG+ 2020). In *Proceedings of the 3rd International Workshop on Natural Language Generation from the Semantic Web (WebNLG+)*, pages 55–76.

Ficler, Jessica and Yoav Goldberg. 2017. Controlling linguistic style aspects in neural language generation. *CoRR*, abs/1707.02633. https://doi.org/10.18653/v1/W17-4912

Fu, Yao, Hao Zhou, Jiaze Chen, and Lei Li. 2019. Rethinking text attribute transfer: A lexical analysis. In *Proceedings of the 12th International Conference on Natural Language Generation, INLG 2019*, pages 24–33. https://doi.org/10.18653/v1/W19-8604

Fu, Zhenxin, Xiaoye Tan, Nanyun Peng, Dongyan Zhao, and Rui Yan. 2018. Style transfer in text: Exploration and evaluation. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18)*, pages 663–670.

Gan, Chuang, Zhe Gan, Xiaodong He, Jianfeng Gao, and Li Deng. 2017. StyleNet: Generating attractive visual captions with styles. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, pages 955–964. https://doi.org/10.1109/CVPR.2017.108

Gao, Yang, Rita Singh, and Bhiksha Raj. 2018. Voice impersonation using generative adversarial networks. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2018*, pages 2506–2510. https://doi.org/10.1109/ICASSP.2018.8462018

Gardent, Claire, Anastasia Shimorina, Shashi Narayan, and Laura Perez-Beltrachini. 2017. The WebNLG challenge: Generating text from RDF data. In *Proceedings of the 10th International Conference on Natural Language Generation, INLG 2017*, pages 124–133. https://doi.org/10.18653/v1/w17-3518

Gardner, Matt, Yoav Artzi, Victoria Basmova, Jonathan Berant, Ben Bogin, Sihao Chen, Pradeep Dasigi, Dheeru Dua, Yanai Elazar, Ananth Gottumukkala, Nitish Gupta, Hannaneh Hajishirzi, Gabriel Ilharco, Daniel Khashabi, Kevin Lin, Jiangming Liu, Nelson F. Liu, Phoebe Mulcaire, Qiang Ning, Sameer Singh, Noah A. Smith, Sanjay Subramanian, Reut Tsarfaty, Eric Wallace, Ally Zhang, and Ben Zhou. 2020. Evaluating models' local decision boundaries via contrast sets. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings, EMNLP 2020*, pages 1307–1323. https://doi.org/10.18653/v1/2020.findings-emnlp.117

Gatt, Albert and Ehud Reiter. 2009. SimpleNLG: A realisation engine for practical applications. In *ENLG 2009 - Proceedings of the 12th European Workshop on Natural Language Generation*, pages 90–93. https://doi.org/10.3115/1610195.1610208

Gatys, Leon A., Alexander S. Ecker, and Matthias Bethge. 2016. Image style transfer using convolutional neural networks. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016*, pages 2414–2423. https://doi.org/10.1109/CVPR.2016.265

Gehrmann, Sebastian, Tosin P. Adewumi, Karmanya Aggarwal, Pawan Sasanka Ammanamanchi, Aremu Anuoluwapo, Antoine Bosselut, Khyathi Raghavi Chandu, Miruna-Adriana Clinciu, Dipanjan Das, Kaustubh D. Dhole, Wanyu Du, Esin Durmus, Ondrej Dusek, Chris Emezue, Varun Gangal, Cristina Garbacea, Tatsunori Hashimoto, Yufang Hou, Yacine Jernite, Harsh Jhamtani, Yangfeng Ji, Shailza Jolly, Dhruv Kumar, Faisal Ladhak, Aman Madaan, Mounica Maddela, Khyati Mahajan, Saad Mahamood, Bodhisattwa Prasad Majumder, Pedro Henrique Martins, Angelina McMillan-Major, Simon Mille, Emiel van Miltenburg, Moin Nadeem, Shashi Narayan, Vitaly Nikolaev, Rubungo Andre Niyongabo, Salomey Osei, Ankur P. Parikh, Laura Perez-Beltrachini, Niranjan Ramesh Rao, Vikas Raunak, Juan Diego Rodriguez, Sashank Santhanam, João Sedoc, Thibault Sellam, Samira Shaikh, Anastasia Shimorina, Marco Antonio Sobrevilla Cabezudo, Hendrik Strobelt, Nishant Subramani, Wei Xu, Diyi Yang, Akhila Yerukola, and Jiawei Zhou. 2021. The GEM

benchmark: Natural language generation, its evaluation and metrics. *CoRR*, abs/2102.01672. https://doi.org/10.18653/v1/2021.gem-1.10

Gkatzia, Dimitra, Oliver Lemon, and Verena Rieser. 2017. Data-to-text generation improves decision-making under uncertainty. *IEEE Computational Intelligence Magazine*, 12(3):10–17. https://doi.org/10.1109/MCI.2017.2708998

Gong, Hongyu, 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. https://doi.org/10.18653/v1/N19-1320

Goodfellow, Ian J., Jean Pouget-Abadie, M. Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *NIPS*.

Goodman, Nelson. 1947. The problem of counterfactual conditionals. *The Journal of Philosophy*, 44(5):113–128. https://doi.org/10.2307/2019988

Grégoire, Francis and Philippe Langlais. 2018. Extracting parallel sentences with bidirectional recurrent neural networks to improve machine translation. In *Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018*, pages 1442–1453.

Gröndahl, Tommi and N. Asokan. 2020. Effective writing style transfer via combinatorial paraphrasing. In *Proceedings on Privacy Enhancing Technologies*, 2020(4):175–195. https://doi.org/10.2478/popets-2020-0068

Gu, Jiatao, Zhengdong Lu, Hang Li, and Victor O. K. Li. 2016. Incorporating copying mechanism in sequence-to-sequence learning. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1631–1640. https://doi.org/10.18653/v1/P16-1154

Gu, Jiatao, Yong Wang, Kyunghyun Cho, and Victor O. K. Li. 2018. Search engine guided neural machine translation. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th Innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18)*, pages 5133–5140.

Gülçehre, Çaglar, Sungjin Ahn, Ramesh Nallapati, Bowen Zhou, and Yoshua Bengio. 2016. Pointing the unknown words. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, Volume 1: Long Papers*, pages 140–149. https://doi.org/10.18653/v1/P16-1014

Guo, Qipeng, Zhijing Jin, Ning Dai, Xipeng Qiu, Xiangyang Xue, David Wipf, and Zheng Zhang. 2020. P2: A plan-and-pretrain approach for knowledge graph-to-text generation. In *Proceedings of the 3rd WebNLG Workshop on Natural Language Generation from the Semantic Web (WebNLG+ 2020)*, pages 100–106.

Guo, Qipeng, Zhijing Jin, Ziyu Wang, Xipeng Qiu, Weinan Zhang, Jun Zhu, Zheng Zhang, and David Wipf. 2021. Fork or fail: Cycle-consistent training with many-to-one mappings. In *The 24th International Conference on Artificial Intelligence and Statistics, AISTATS 2021*, volume 130 of *Proceedings of Machine Learning Research*, pages 1828–1836.

Guu, Kelvin, Tatsunori B. Hashimoto, Yonatan Oren, and Percy Liang. 2018. Generating sentences by editing prototypes. *Transactions of the Association for Computational Linguistics*, 6:437–450. https://doi.org/10.1162/tacl_a_00030

Harrison, Sara. 2019. Twitter and Instagram unveil new ways to combat hate–again. https://www.wired.com/story/twitter-instagram-unveil-new-ways-combat-hate-again/

Hashimoto, Tatsunori B., Kelvin Guu, Yonatan Oren, and Percy Liang. 2018. A retrieve-and-edit framework for predicting structured outputs. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018*, pages 10073–10083.

He, Ruining and Julian J. McAuley. 2016. Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. In *Proceedings of the 25th International Conference on World Wide Web, WWW 2016*, pages 507–517. https://doi.org/10.1145/2872427.2883037

Henderson, James. 2020. The unstoppable rise of computational linguistics in deep learning. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020*, pages 6294–6306.

https://doi.org/10.18653/v1/2020.acl-main.561

Hill, Felix, Kyunghyun Cho, and Anna Korhonen. 2016. Learning distributed representations of sentences from unlabelled data. In *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1367–1377. https://doi.org/10.18653/v1/N16-1162

Hoang, Vu Cong Duy, Philipp Koehn, Gholamreza Haffari, and Trevor Cohn. 2018. Iterative back-translation for neural machine translation. In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, pages 18–24. https://doi.org/10.18653/v1/W18-2703

Hossain, Nabil, Marjan Ghazvininejad, and Luke Zettlemoyer. 2020. Simple and effective retrieve-edit-rerank text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020*, pages 2532–2538. https://doi.org/10.18653/v1/2020.acl-main.228

Hovy, Dirk 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, ACL 2016, Volume 2: Short Papers*, pages 591–598. https://doi.org/10.18653/v1/P16-2096

Hovy, Eduard. 1987. Generating natural language under pragmatic constraints. *Journal of Pragmatics*, 11(6):689–719. https://doi.org/10.1016/0378-2166(87)90099-3

Hovy, Eduard H. 1990. Pragmatics and natural language generation. *Artificial Intelligence*, 43(2):153–197. https://doi.org/10.1016/0004-3702(90)90084-D

Hu, Zhiqiang, R. K. Lee, and C. Aggarwal. 2020. Text style transfer: A review and experiment evaluation. *ArXiv*, abs/2010.12742.

Hu, Zhiting, Zichao Yang, Xiaodan Liang, R. Salakhutdinov, and E. Xing. 2017. Toward controlled generation of text. In *ICML*, pages 1587–1596.

Huang, Chenyang, Osmar Zaïane, Amine Trabelsi, and Nouha Dziri. 2018. Automatic dialogue generation with expressed emotions. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 49–54. https://doi.org/10.18653/v1/N18-2008

Huang, Yufang, Wentao Zhu, Deyi Xiong, Yiye Zhang, Changjian Hu, and Feiyu Xu. 2020. Cycle-consistent adversarial autoencoders for unsupervised text style transfer. In *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020*, pages 2213–2223. https://doi.org/10.18653/v1/2020.coling-main.201

Jackson, Philip T. G., Amir Atapour Abarghouei, Stephen Bonner, Toby P. Breckon, and Boguslaw Obara. 2019. Style augmentation: Data augmentation via style randomization. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2019*, pages 83–92.

Jafaritazehjani, Somayeh, Gwénolé Lecorvé, Damien Lolive, and John Kelleher. 2020. Style versus content: A distinction without a (learnable) difference? In *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020*, pages 2169–2180. https://doi.org/10.18653/v1/2020.coling-main.197

Jang, Eric, Shixiang Gu, and Ben Poole. 2017. Categorical reparameterization with Gumbel-Softmax. In *5th International Conference on Learning Representations, ICLR 2017*, OpenReview.net.

Jhamtani, Harsh, Varun Gangal, Eduard H. Hovy, and Eric Nyberg. 2017. Shakespearizing modern language using copy-enriched sequence-to-sequence models. *CoRR*, abs/1707.01161. https://doi.org/10.18653/v1/W17-4902

Jia, Robin and Percy Liang. 2017. Adversarial examples for evaluating reading comprehension systems. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017*, pages 2021–2031. https://doi.org/10.18653/v1/d17-1215

Jin, Di, Zhijing Jin, Joey Tianyi Zhou, Lisa Orii, and Peter Szolovits. 2020a. Hooks in the headline: Learning to generate headlines with controlled styles. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5082–5093. https://doi.org/10.18653/v1/2020.acl-main.456

Jin, Di, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. 2020b. Is BERT really robust? A strong baseline for natural language attack on text classification and entailment. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI*

2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, pages 8018–8025. https://doi.org/10.1609/aaai.v34i05.6311

Jin, Zhijing, Di Jin, Jonas Mueller, Nicholas Matthews, and Enrico Santus. 2019. IMaT: Unsupervised text attribute transfer via iterative matching and translation. 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 2019*, pages 3095–3107. https://doi.org/10.18653/v1/D19-1306

John, Vineet, 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. https://doi.org/10.18653/v1/P19-1041

Kajiwara, Tomoyuki. 2019. Negative lexically constrained decoding for paraphrase generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6047–6052. https://doi.org/10.18653/v1/P19-1607

Kale, Mihir and Abhinav Rastogi. 2020. Text-to-text pre-training for data-to-text tasks. In *Proceedings of the 13th International Conference on Natural Language Generation, INLG 2020*, pages 97–102.

Kang, Yipeng, Tonghan Wang, and Gerard de Melo. 2020. Incorporating pragmatic reasoning communication into emergent language. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020*, pages 10348–10359.

Karadzhov, Georgi, Tsvetomila Mihaylova, Yasen Kiprov, Georgi Georgiev, Ivan Koychev, and Preslav Nakov. 2017. The case for being average: A mediocrity approach to style masking and author obfuscation - (best of the labs track at CLEF-2017). In *Experimental IR Meets Multilinguality, Multimodality, and Interaction - 8th International Conference of the CLEF Association, CLEF 2017, Proceedings*, volume 10456 of *Lecture Notes in Computer Science*, pages 173–185. https://doi.org/10.1007/978-3-319-65813-1_18

Keskar, Nitish Shirish, Bryan McCann, Lav R. Varshney, Caiming Xiong, and Richard Socher. 2019. CTRL: A conditional transformer language model for controllable generation. *CoRR*, abs/1909.05858.

Khosmood, Foaad and Robert Levinson. 2010. Automatic synonym and phrase replacement show promise for style transformation. In *Ninth International Conference on Machine Learning and Applications, ICMLA 2010*, pages 958–961. https://doi.org/10.1109/ICMLA.2010.153

Khosmood, Foaad and Robert A. Levinson. 2008. Automatic natural language style classification and transformation. In *BCS-IRSG Workshop on Corpus Profiling*, pages 1–11. https://doi.org/10.1109/ICMLA.2010.153

Kim, Yoon. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014*, pages 1746–1751. https://doi.org/10.3115/v1/D14-1181

Kingma, Diederik P. and M. Welling. 2014. Auto-encoding variational Bayes. *CoRR*, abs/1312.6114.

Koncel-Kedziorski, Rik, Ioannis Konstas, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2016. A theme-rewriting approach for generating algebra word problems. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016*, pages 1617–1628. https://doi.org/10.18653/v1/d16-1168

Kowalski, Robin M., Gary W. Giumetti, Amber N. Schroeder, and Micah R. Lattanner. 2014. Bullying in the digital age: A critical review and meta-analysis of cyberbullying research among youth. *Psychological Bulletin*, 140(4):1073. https://doi.org/10.1037/a0035618, PubMed: 24512111

Krippendorff, Klaus. 2018. *Content Analysis: An Introduction to Its Methodology*. Sage Publications.

Krishna, Kalpesh, John Wieting, and Mohit Iyyer. 2020. Reformulating unsupervised style transfer as paraphrase generation. *CoRR*, abs/2010.05700. https://doi.org/10.18653/v1/2020.emnlp-main.55

Kusner, Matt, Yu Sun, Nicholas Kolkin, and Kilian Weinberger. 2015. From word embeddings to document distances. In *International Conference on Machine Learning*, pages 957–966.

Lai, Huiyuan, Antonio Toral, and M. Nissim. 2021. Thank you BART! Rewarding pre-trained models improves formality style transfer. In *ACL/IJCNLP*, pages 484–494. https://doi.org/10.18653/v1/2021.acl-short.62

Lakoff, Robin. 1973. Language and woman's place. *Language in Society*, 2(1):45–79.

Lample, Guillaume, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018a. Unsupervised machine translation using monolingual corpora only. In *6th International Conference on Learning Representations, ICLR 2018*, OpenReview.net.

Lample, Guillaume, Myle Ott, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018b. Phrase-based & neural unsupervised machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5039–5049. https://doi.org/10.18653/v1/D18-1549

Lample, Guillaume, Sandeep Subramanian, Eric Michael Smith, Ludovic Denoyer, Marc'Aurelio Ranzato, and Y-Lan Boureau. 2019. Multiple-attribute text rewriting. In *ICLR*.

Lee, Joosung. 2020. Stable style transformer: Delete and generate approach with encoder-decoder for text style transfer. *CoRR*, abs/2005.12086.

Lewis, Patrick S. H., Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive NLP tasks. *CoRR*, abs/2005.11401.

Li, Dianqi, Yizhe Zhang, Zhe Gan, Yu Cheng, Chris Brockett, Bill Dolan, and Ming-Ting Sun. 2019. Domain adaptive 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 2019*, pages 3302–3311. https://doi.org/10.18653/v1/D19-1325

Li, Jiwei, Michel Galley, Chris Brockett, Georgios P. Spithourakis, Jianfeng Gao, and William B. Dolan. 2016. A persona-based neural conversation model. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, Volume 1: Long Papers*, pages 994–1003. https://doi.org/10.18653/v1/p16-1094

Li, Juncen, Robin Jia, He He, and Percy Liang. 2018. 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. https://doi.org/10.18653/v1/N18-1169

Li, Mingzhe, Xiuying Chen, Min Yang, Shen Gao, Dongyan Zhao, and Rui Yan. 2021. The style-content duality of attractiveness: Learning to write eye-catching headlines via disentanglement. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021*, pages 13252–13260.

Li, Xiang Lisa and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers)*, pages 4582–4597. https://doi.org/10.18653/v1/2021.acl-long.353

Li, Yuan, Chunyuan Li, Yizhe Zhang, Xiujun Li, Guoqing Zheng, Lawrence Carin, and Jianfeng Gao. 2020. Complementary auxiliary classifiers for label-conditional text generation. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020*, pages 8303–8310. https://doi.org/10.1609/aaai.v34i05.6346

Liao, Yi, Lidong Bing, Piji Li, Shuming Shi, Wai Lam, and Tong Zhang. 2018. QuaSE: Sequence editing under quantifiable guidance. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3855–3864. https://doi.org/10.18653/v1/D18-1420

Lin, Chin Yew and Franz Josef Och. 2004. Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)*, pages 605–612.

https://doi.org/10.3115/1218955.1219032

Liu, Dayiheng, Jie Fu, Yidan Zhang, Chris Pal, and Jiancheng Lv. 2020. Revision in continuous space: Unsupervised text style transfer without adversarial learning. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020*, pages 8376–8383. https://doi.org/10.1609/aaai.v34i05.6355

Locatello, Francesco, Stefan Bauer, Mario Lucic, Gunnar Rätsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. 2019. Challenging common assumptions in the unsupervised learning of disentangled representations. In *Proceedings of the 36th International Conference on Machine Learning, ICML 019*, volume 97 of *Proceedings of Machine Learning Research*, pages 4114–4124.

Locatello, Francesco, Ben Poole, Gunnar Rätsch, Bernhard Schölkopf, Olivier Bachem, and Michael Tschannen. 2020. Weakly-supervised disentanglement without compromises. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020*, volume 119 of *Proceedings of Machine Learning Research*, pages 6348–6359.

Logeswaran, Lajanugen, Honglak Lee, and Samy Bengio. 2018. Content preserving text generation with attribute controls. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018*, pages 5108–5118.

Luo, Fuli, Peng Li, Jie Zhou, Pengcheng Yang, Baobao Chang, Zhifang Sui, and Xu Sun. 2019. A dual reinforcement learning framework for unsupervised text style transfer. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence, IJCAI 2019*, pages 5116–5122. https://doi.org/10.24963/ijcai.2019/711

Ma, Xinyao, Maarten Sap, Hannah Rashkin, and Yejin Choi. 2020. PowerTransformer: Unsupervised controllable revision for biased language correction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020*, pages 7426–7441. https://doi.org/10.18653/v1/2020.emnlp-main.602

Madaan, Aman, Amrith Setlur, Tanmay Parekh, Barnabás Póczos, Graham Neubig, Yiming Yang, Ruslan Salakhutdinov, Alan W. Black, and Shrimai Prabhumoye. 2020. Politeness transfer: A tag and generate approach. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020*, pages 1869–1881. https://doi.org/10.18653/v1/2020.acl-main.169

Madnani, Nitin and Bonnie J. Dorr. 2010. Generating phrasal and sentential paraphrases: A survey of data-driven methods. *Compututational Linguistics*, 36(3):341–387. https://doi.org/10.1162/coli_a_00002

Mairesse, François and Marilyn A. Walker. 2011. Controlling user perceptions of linguistic style: Trainable generation of personality traits. *Computational Linguistics*, 37(3):455–488. https://doi.org/10.1162/COLI_a_00063

Malmi, Eric, Aliaksei Severyn, and Sascha Rothe. 2020. Unsupervised text style transfer with padded masked language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020*, pages 8671–8680. https://doi.org/10.18653/v1/2020.emnlp-main.699

Mani, Inderjeet. 2001. *Automatic Summarization*, volume 3. John Benjamins Publishing.

Mansoorizadeh, Muharram, Taher Rahgooy, Mohammad Aminiyan, and Mahdy Eskandari. 2016. Author obfuscation using WordNet and language models—Notebook for PAN at CLEF 2016. In *CLEF 2016 Evaluation Labs and Workshop—Working Notes Papers*, pages 5–8.

Marie, Benjamin and Atsushi Fujita. 2017. Efficient extraction of pseudo-parallel sentences from raw monolingual data using word embeddings. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Volume 2: Short Papers*, pages 392–398. https://doi.org/10.18653/v1/P17-2062

McDonald, David D. and James Pustejovsky. 1985. A computational theory of prose style for natural language generation. In *EACL 1985, 2nd Conference of the European Chapter of the Association for Computational Linguistics*, pages 187–193.

McTear, Michael F. 2002. Spoken dialogue technology: Enabling the conversational user interface. *ACM Computing Surveys*, 34(1):90–169. https://doi.org/10.1145/505282.505285

Merity, Stephen, Caiming Xiong, James Bradbury, and R. Socher. 2017. Pointer sentinel mixture models. *ArXiv*, abs/1609.07843.

Mir, Remi, Bjarke Felbo, Nick Obradovich, and Iyad Rahwan. 2019. Evaluating style transfer for text. 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 495–504. https://doi.org/10.18653/v1/N19-1049

Mou, Lili and Olga Vechtomova. 2020. Stylized text generation: Approaches and applications. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts*, pages 19–22. https://doi.org/10.18653/v1/2020.acl-tutorials.5

Mueller, Jonas, David K. Gifford, and Tommi S. Jaakkola. 2017. Sequence to better sequence: Continuous revision of combinatorial structures. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 2536–2544.

Munteanu, Dragos Stefan and Daniel Marcu. 2005. Improving machine translation performance by exploiting non-parallel corpora. *Computational Linguistics*, 31(4):477–504. https://doi.org/10.1162/089120105775299168

Nikolov, Nikola I. and Richard H. R. Hahnloser. 2019. Large-scale hierarchical alignment for data-driven text rewriting. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing, RANLP 2019*, pages 844–853. https://doi.org/10.26615/978-954-452-056-4_098

Niu, Tong and Mohit Bansal. 2018. Polite dialogue generation without parallel data. *Transactions of the Association for Computational Linguistics*, 6:373–389. https://doi.org/10.1162/tacl_a_00027

Niu, Xing, Marianna J. 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, EMNLP 2017*, pages 2814–2819. https://doi.org/10.18653/v1/d17-1299

Niu, Xing, Sudha Rao, and Marine Carpuat. 2018. Multi-task neural models for translating between styles within and across languages. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1008–1021.

Novikova, Jekaterina, Ondrej Dusek, and Verena Rieser. 2017. The E2E dataset: New challenges for end-to-end generation. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, pages 201–206. https://doi.org/10.18653/v1/w17-5525

Oksanen, Atte, James Hawdon, Emma Holkeri, Matti Näsi, and Pekka Räsänen. 2014. Exposure to online hate among young social media users. In *Soul of Society: A Focus on the Lives of Children & Youth*. Emerald Group Publishing Limited. https://doi.org/10.1108/S1537-466120140000018021

Pang, Richard Yuanzhe. 2019. The daunting task of real-world textual style transfer auto-evaluation. *CoRR*, abs/1910.03747. https://doi.org/10.18653/v1/D19-5557

Pang, Richard Yuanzhe and Kevin Gimpel. 2019. Unsupervised evaluation metrics and learning criteria for non-parallel textual transfer. In *Proceedings of the 3rd Workshop on Neural Generation and Translation@EMNLP-IJCNLP 2019*, pages 138–147. https://doi.org/10.18653/v1/D19-5614

Papineni, Kishore, 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 of the Association for Computational Linguistics*, pages 311–318. https://doi.org/10.3115/1073083.1073135

Parikh, Ankur P., Xuezhi Wang, Sebastian Gehrmann, Manaal Faruqui, Bhuwan Dhingra, Diyi Yang, and Dipanjan Das. 2020. ToTTo: A controlled table-to-text generation dataset. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020*, pages 1173–1186. https://doi.org/10.18653/v1/2020.emnlp-main.89

Pfaff, Carol W. 1979. Constraints on language mixing: Intrasentential code-switching and borrowing in Spanish/English. *Language*, 55(2):291–318. https://doi.org/10.2307/412586

Poplack, Shana. 2000. Sometimes I'll start a sentence in Spanish y termino en español: Toward a typology of code-switching. *The Bilingualism Reader*, 18(2):221–256.

Popović, Maja. 2015. chrF: Character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*,

pages 392–395. https://doi.org/10.18653/v1/W15-3049

Post, Matt and David Vilar. 2018. Fast lexically constrained decoding with dynamic beam allocation for neural machine translation. 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 1314–1324. https://doi.org/10.18653/v1/N18-1119

Power, Richard, Donia Scott, and Nadjet Bouayad-Agha. 2003. Generating texts with style. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 444–452. https://doi.org/10.1007/3-540-36456-0_47

Prabhumoye, Shrimai, 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. https://doi.org/10.18653/v1/P18-1080

Pryzant, Reid, Richard Diehl Martinez, Nathan Dass, Sadao Kurohashi, Dan Jurafsky, and Diyi Yang. 2020. Automatically neutralizing subjective bias in text. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020*, pages 480–489. https://doi.org/10.1609/aaai.v34i01.5385

Qian, Kaizhi, Yang Zhang, Shiyu Chang, Xuesong Yang, and Mark Hasegawa-Johnson. 2019. AutoVC: Zero-shot voice style transfer with only autoencoder loss. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019*, volume 97 of *Proceedings of Machine Learning Research*, pages 5210–5219.

Qin, Guanghui and Jason Eisner. 2021. Learning how to ask: Querying LMs with mixtures of soft prompts. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021*, pages 5203–5212. https://doi.org/10.18653/v1/2021.naacl-main.410

Qin, Lianhui, Antoine Bosselut, Ari Holtzman, Chandra Bhagavatula, Elizabeth Clark, and Yejin Choi. 2019. Counterfactual story reasoning and generation. 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 2019*, pages 5042–5052. https://doi.org/10.18653/v1/D19-1509

Radford, Alec, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8):9.

Raffel, Colin, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21:140:1–140:67.

Rao, Sudha 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)*, pages 129–140. https://doi.org/10.18653/v1/N18-1012

Reddy, Sravana and Kevin Knight. 2016. Obfuscating gender in social media writing. In *Proceedings of the First Workshop on NLP and Computational Social Science, NLP+CSS@EMNLP 2016*, pages 17–26. https://doi.org/10.18653/v1/W16-5603

Reiter, Ehud and Robert Dale. 1997. Building applied natural language generation systems. *Natural Language Engineering*, 3(1):57–87. https://doi.org/10.1017/S1351324997001502

Reiter, Ehud, Roma Robertson, and Liesl M. Osman. 2003. Lessons from a failure: Generating tailored smoking cessation letters. *Artificial Intelligence*, 144(1-2):41–58. https://doi.org/10.1016/S0004-3702(02)00370-3

Reiter, Ehud, Somayajulu Sripada, Jim Hunter, Jin Yu, and Ian Davy. 2005. Choosing words in computer-generated weather forecasts. *Artificial Intelligence*, 167(1–2):137–169. https://doi.org/10.1016/j.artint.2005.06.006

Ren, Shuo, Yu Wu, Shujie Liu, Ming Zhou, and Shuai Ma. 2020. A retrieve-and-rewrite initialization method for unsupervised machine translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics,*

ACL 2020, pages 3498–3504. https://doi.org/10.18653/v1/2020.acl-main.320

Rezende, Danilo Jimenez, Shakir Mohamed, and Daan Wierstra. 2014. Stochastic backpropagation and approximate inference in deep generative models. In *Proceedings of the 31th International Conference on Machine Learning, ICML 2014*, volume 32 of *JMLR Workshop and Conference Proceedings*, pages 1278–1286.

Ribeiro, Leonardo F. R., Martin Schmitt, Hinrich Schütze, and Iryna Gurevych. 2020. Investigating pretrained language models for graph-to-text generation. *CoRR*, abs/2007.08426. https://doi.org/10.18653/v1/2021.nlp4convai-1.20

Riley, Parker, Noah Constant, Mandy Guo, Girish Kumar, David Uthus, and Zarana Parekh. 2021. TextSETTR: Few-shot text style extraction and tunable targeted restyling. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3786–3800. https://doi.org/10.18653/v1/2021.acl-long.293

Roller, Stephen, Emily Dinan, Naman Goyal, Da Ju, Mary Williamson, Yinhan Liu, Jing Xu, Myle Ott, Eric Michael Smith, Y-Lan Boureau, and Jason Weston. 2021. Recipes for building an open-domain chatbot. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, EACL 2021*, pages 300–325. https://doi.org/10.18653/v1/2021.eacl-main.24

Romanov, Alexey, 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. https://doi.org/10.18653/v1/N19-1088

Ruder, Manuel, Alexey Dosovitskiy, and Thomas Brox. 2016. Artistic style transfer for videos. In *Pattern Recognition - 38th German Conference, GCPR 2016*, volume 9796 of *Lecture Notes in Computer Science*, pages 26–36. https://doi.org/10.1007/978-3-319-45886-1_3

Rush, Alexander M., 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, EMNLP 2015*, pages 379–389. https://doi.org/10.18653/v1/D15-1044

Russell, Stuart J., Daniel Dewey, and Max Tegmark. 2015. Research priorities for robust and beneficial artificial intelligence. *AI Magazine*, 36(4):105–114. https://doi.org/10.1609/aimag.v36i4.2577

Sancheti, Abhilasha, Kundan Krishna, Balaji Vasan Srinivasan, and Anandhavelu Natarajan. 2020. Reinforced rewards framework for text style transfer. In *European Conference on Information Retrieval*, pages 545–560. https://doi.org/10.1007/978-3-030-45439-5_36

Scao, Teven Le and Alexander M. Rush. 2021. How many data points is a prompt worth? In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021*, pages 2627–2636. https://doi.org/10.18653/v1/2021.naacl-main.208

Schler, Jonathan, Moshe Koppel, Shlomo Argamon, and James W. Pennebaker. 2006. Effects of age and gender on blogging. In *Computational Approaches to Analyzing Weblogs, Papers from the 2006 AAAI Spring Symposium, Technical Report SS-06-03*, pages 199–205.

See, Abigail, 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, ACL 2017, Volume 1: Long Papers*, pages 1073–1083. https://doi.org/10.18653/v1/P17-1099

Sennrich, Rico, Barry Haddow, and Alexandra Birch. 2016a. Controlling politeness in neural machine translation via side constraints. In *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 35–40. https://doi.org/10.18653/v1/n16-1005

Sennrich, Rico, Barry Haddow, and Alexandra Birch. 2016b. Edinburgh neural machine translation systems for WMT 16. In *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, pages 371–376. https://doi.org/10.18653/v1/W16-2323

Shang, Mingyue, Piji Li, Zhenxin Fu, Lidong Bing, Dongyan Zhao, Shuming Shi, and

Rui Yan. 2019. Semi-supervised text style transfer: Cross projection in latent space. 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 4937–4946. https://doi.org/10.18653/v1/D19-1499

Shao, Chengcheng, Giovanni Luca Ciampaglia, Onur Varol, Kai-Cheng Yang, Alessandro Flammini, and Filippo Menczer. 2018. The spread of low-credibility content by social bots. *Nature Communications*, 9(1):1–9. https://doi.org/10.1038/s41467-018-06930-7, PubMed: 30459415

Sharma, Ashish, Inna W. Lin, Adam S. Miner, David C. Atkins, and Tim Althoff. 2021. Towards facilitating empathic conversations in online mental health support: A reinforcement learning approach. *CoRR*, abs/2101.07714. https://doi.org/10.1145/3442381.3450097

Sheikha, Fadi Abu and Diana Inkpen. 2011. Generation of formal and informal sentences. In *ENLG 2011 - Proceedings of the 13th European Workshop on Natural Language Generation*, pages 187–193.

Shen, Tianxiao, 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.

Shetty, Jitesh and Jafar Adibi. 2004. The Enron email dataset database schema and brief statistical report. *Information Sciences Institute Technical Report, University of Southern California*, 4(1):120–128.

Shetty, Rakshith, Bernt Schiele, and Mario Fritz. 2018. A4NT: Author attribute anonymity by adversarial training of neural machine translation. In *27th USENIX Security Symposium, USENIX Security 2018*, pages 1633–1650.

Shuster, Kurt, Samuel Humeau, Antoine Bordes, and Jason Weston. 2020. Image-Chat: Engaging grounded conversations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020*, pages 2414–2429. https://doi.org/10.18653/v1/2020.acl-main.219

Song, Kaitao, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2019. MASS: Masked sequence to sequence pre-training for language generation. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019*, volume 97 of *Proceedings of Machine Learning Research*, pages 5926–5936.

Sripada, Somayajulu, Ehud Reiter, Ian Davy, and Kristian Nilssen. 2004. Lessons from deploying NLG technology for marine weather forecast text generation. In *Proceedings of the 16th European Conference on Artificial Intelligence, ECAI'2004, including Prestigious Applicants of Intelligent Systems, PAIS 2004*, pages 760–764.

Stamatatos, Efstathios, S. Michos, Nikos Fakotakis, and George Kokkinakis. 1997. A user-assisted business letter generator dealing with text's stylistic variations. In *Proceedings Ninth IEEE International Conference on Tools with Artificial Intelligence*, pages 182–189.

Starr, William. 2019. Counterfactuals. *The Stanford Encyclopedia of Philosophy*.

Sudhakar, Akhilesh, 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 2019*, pages 3267–3277.

Sutskever, Ilya, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems*, pages 3104–3112.

Syed, Bakhtiyar, Gaurav Verma, Balaji Vasan Srinivasan, Anandhavelu Natarajan, and Vasudeva Varma. 2020. Adapting language models for non-parallel author-stylized rewriting. In *AAAI*, pages 9008–9015. https://doi.org/10.1609/aaai.v34i05.6433

Tan, Sharon Swee Lin and Nadee Goonawardene. 2017. Internet health information seeking and the patient-physician relationship: A systematic review. *Journal of Medical Internet Research*, 19(1):e9. https://doi.org/10.2196/jmir.5729, PubMed: 28104579

Tannen, Deborah. 1990. Gender differences in topical coherence: Creating involvement in best friends' talk. *Discourse Processes*, 13(1):73–90. https://doi.org/10.1080/01638539009544747

Tian, Youzhi, Zhiting Hu, and Zhou Yu. 2018. Structured content preservation for unsupervised text style transfer. *CoRR*, abs/1810.06526.

Tikhonov, Alexey, Viacheslav Shibaev, Aleksander Nagaev, Aigul Nugmanova, and Ivan P. Yamshchikov. 2019. Style transfer for texts: Retrain, report errors, compare with rewrites. 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 3936–3945. https://doi.org/10.18653/v1/D19-1406

Tikhonov, Alexey and Ivan P. Yamshchikov. 2018. What is wrong with style transfer for texts? *CoRR*, abs/1808.04365

Tran, Minh, Yipeng Zhang, and Mohammad Soleymani. 2020. Towards a friendly online community: An unsupervised style transfer framework for profanity redaction. *CoRR*, abs/2011.00403.

Trudgill, Peter. 1972. Sex, covert prestige and linguistic change in the urban British English of Norwich. *Language in Society*, 1(2):179–195. https://doi.org/10.1017/S0047404500000488

Tse, Jonathan, Dawn E. Schrader, Dipayan P. Ghosh, Tony C. Liao, and David Lundie. 2015. A bibliometric analysis of privacy and ethics in IEEE Security and Privacy. *Ethics and Information Technology*, 17(2):153–163. https://doi.org/10.1007/s10676-015-9369-6

Uszkoreit, Jakob, Jay Ponte, Ashok C. Popat, and Moshe Dubiner. 2010. Large scale parallel document mining for machine translation. In *COLING 2010, 23rd International Conference on Computational Linguistics, Proceedings of the Conference*, pages 1101–1109.

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017*, pages 5998–6008.

Vincent, Pascal, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol. 2010. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of Machine Learning Research*, 11:3371–3408.

Voigt, Rob, 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*.

Wang, Kai, Xiaojun Quan, and Rui Wang. 2019. BiSET: Bi-directional selective encoding with template for abstractive summarization. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Volume 1: Long Papers*, pages 2153–2162. https://doi.org/10.18653/v1/p19-1207

Wang, Ke, Hang Hua, and Xiaojun Wan. 2019. Controllable unsupervised text attribute transfer via editing entangled latent representation. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019*, pages 11034–11044.

Wang, Yunli, Yu Wu, Lili Mou, Zhoujun Li, and Wenhan Chao. 2019. Harnessing pre-trained neural networks with rules for formality 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 3573–3578. https://doi.org/10.18653/v1/D19-1365

Waseem, Zeerak, Thomas Davidson, Dana Warmsley, and Ingmar Weber. 2017. Understanding abuse: A typology of abusive language detection subtasks. In *Proceedings of the First Workshop on Abusive Language Online, ALW@ACL 2017*, pages 78–84. https://doi.org/10.18653/v1/w17-3012

Weng, Wei-Hung, Yu-An Chung, and Peter Szolovits. 2019. Unsupervised clinical language translation. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019*, pages 3121–3131. https://doi.org/10.1145/3292500.3330710

Weston, Jason, Emily Dinan, and Alexander H. Miller. 2018. Retrieve and refine: Improved sequence generation models for dialogue. In *Proceedings of the 2nd International Workshop on Search-Oriented Conversational AI, SCAI@EMNLP 2018*, pages 87–92. https://doi.org/10.18653/v1/w18-5713

Williams, Ronald J. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement Learning. *Machine Learning*, 8(3–4):229–256. https://doi.org/10.1007/BF00992696

Wiseman, Sam, Stuart M. Shieber, and Alexander M. Rush. 2017. Challenges in data-to-document generation. In

Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, pages 2253–2263. https://doi.org/10.18653/v1/d17-1239

Wu, Jiawei, Xin Wang, and William Yang Wang. 2019. Extract and edit: An alternative to back-translation for unsupervised neural machine translation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Volume 1 (Long and Short Papers)*, pages 1173–1183. https://doi.org/10.18653/v1/n19-1120

Wu, Xing, Tao Zhang, Liangjun Zang, Jizhong Han, and Songlin Hu. 2019. "Mask and infill" : Applying masked language model to sentiment transfer. *CoRR*, abs/1908.08039. https://doi.org/10.24963/ijcai.2019/732

Wu, Yu, Yunli Wang, and Shujie Liu. 2020. A dataset for low-resource stylized sequence-to-sequence generation. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020*, pages 9290–9297. https://doi.org/10.1609/aaai.v34i05.6468

Xing, Xiaoyu, Zhijing Jin, Di Jin, Bingning Wang, Qi Zhang, and Xuanjing Huang. 2020. Tasty burgers, soggy fries: Probing aspect robustness in aspect-based sentiment analysis. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020*, pages 3594–3605. https://doi.org/10.18653/v1/2020.emnlp-main.292

Xu, Jingjing, Xu Sun, Qi Zeng, Xiaodong Zhang, Xuancheng Ren, Houfeng Wang, and Wenjie Li. 2018. Unpaired sentiment-to-sentiment translation: A cycled reinforcement learning approach. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 979–988. https://doi.org/10.18653/v1/P18-1090

Xu, Peng, Jackie Chi Kit Cheung, and Yanshuai Cao. 2020. On variational learning of controllable representations for text without supervision. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020*, volume 119 of *Proceedings of Machine Learning Research*, pages 10534–10543.

Xu, Qiuling, Guanhong Tao, Siyuan Cheng, Lin Tan, and Xiangyu Zhang. 2020. Towards feature space adversarial attack. *CoRR*, abs/2004.12385

Xu, Ruochen, Tao Ge, and Furu Wei. 2019. Formality style transfer with hybrid textual annotations. *CoRR*, abs/1903.06353.

Xu, Wei, Alan Ritter, Bill Dolan, Ralph Grishman, and Colin Cherry. 2012. Paraphrasing for style. In *COLING 2012, 24th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers*, pages 2899–2914.

Yamshchikov, Ivan P., Viacheslav Shibaev, Nikolay Khlebnikov, and Alexey Tikhonov. 2021. Style-transfer and paraphrase: Looking for a sensible semantic similarity metric. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021*, pages 14213–14220.

Yamshchikov, Ivan P., Viacheslav Shibaev, Aleksander Nagaev, Jürgen Jost, and Alexey Tikhonov. 2019. Decomposing textual information for style transfer. In *Proceedings of the 3rd Workshop on Neural Generation and Translation*, pages 128–137. https://doi.org/10.18653/v1/D19-5613

Yang, Zichao, 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 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018*, pages 7298–7309.

Yi, Xiaoyuan, Zhenghao Liu, Wenhao Li, and Maosong Sun. 2020. Text style transfer via learning style instance supported latent space. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI 2020*, pages 3801–3807. https://doi.org/10.24963/ijcai.2020/526

Yuan, Siyang, Pengyu Cheng, Ruiyi Zhang, Weituo Hao, Zhe Gan, and Lawrence Carin. 2021. Improving zero-shot voice style transfer via disentangled representation learning. *CoRR*, abs/2103.09420.

Zeng, Kuo-Hao, Mohammad Shoeybi, and Ming-Yu Liu. 2020. Style example-guided text generation using generative

adversarial transformers. *CoRR*, abs/2003.00674.

Zhang, Jingyi, Masao Utiyama, Eiichiro Sumita, Graham Neubig, and Satoshi Nakamura. 2018a. Guiding neural machine translation with retrieved translation pieces. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, Volume 1 (Long Papers)*, pages 1325–1335. https://doi.org/10.18653/v1/n18-1120

Zhang, Saizheng, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston. 2018b. 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, ACL 2018, Volume 1: Long Papers*, pages 2204–2213.

Zhang, Tianyi, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating text generation with BERT. In *8th International Conference on Learning Representations, ICLR 2020*, OpenReview.net.

Zhang, Xu-Yao and Cheng-Lin Liu. 2013. Writer adaptation with style transfer mapping. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(7):1773–1787. https://doi.org/10.1109/TPAMI.2012.239, PubMed: 23682002

Zhang, Yi, Tao Ge, and Xu Sun. 2020. Parallel data augmentation for formality style transfer. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3221–3228. https://doi.org/10.18653/v1/2020.acl-main.294

Zhang, Yi, Jingjing Xu, Pengcheng Yang, and Xu Sun. 2018c. Learning sentiment memories for sentiment modification without parallel data. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1103–1108. https://doi.org/10.18653/v1/d18-1138

Zhang, Zhirui, Shuo Ren, Shujie Liu, Jianyong Wang, Peng Chen, Mu Li, Ming Zhou, and Enhong Chen. 2018d. Style transfer as unsupervised machine translation. *CoRR*, abs/1808.07894.

Zhao, Junbo Jake, Yoon Kim, Kelly Zhang, Alexander M. Rush, and Yann LeCun. 2018. Adversarially regularized autoencoders. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 5897–5906.

Zheng, Xu, Tejo Chalasani, Koustav Ghosal, Sebastian Lutz, and Aljosa Smolic. 2019. STaDA: Style transfer as data augmentation. In *Proceedings of the 14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications, VISIGRAPP 2019, Volume 4: VISAPP*, pages 107–114. https://doi.org/10.5220/0007353401070114

Zhu, Jun-Yan, Taesung Park, Phillip Isola, and Alexei A. Efros. 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *IEEE International Conference on Computer Vision, ICCV 2017*, pages 2242–2251. https://doi.org/10.1109/ICCV.2017.244

Zhu, Zhemin, Delphine Bernhard, and Iryna Gurevych. 2010. A monolingual tree-based translation model for sentence simplification. In *COLING 2010, 23rd International Conference on Computational Linguistics, Proceedings of the Conference*, pages 1353–1361.

Zue, Victor W. and James R. Glass. 2000. Conversational interfaces: Advances and challenges. *Proceedings of the IEEE*, 88(8):1166–1180. https://doi.org/10.1109/5.880078