Institut für Maschinelle Sprachverarbeitung
Universität Stuttgart
Pfaffenwaldring 5B
D-70569 Stuttgart

Emotion Style Transfer in Text

Ngoc Khanh Van Hoang

Studiengang: M.Sc. Computational Linguistics

Prüfer: Prof. Dr. Thang Vu
Betreuer: Dr. Fritz Hohl

Beginn der Arbeit: 15.10.2021
Ende der Arbeit: 15.04.2022

Erklärung (Statement of Authorship)

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig verfasst habe und dabei keine andere als die angegebene Literatur verwendet habe. Alle Zitate und sinngemäßen Entlehnungen sind als solche unter genauer Angabe der Quelle gekennzeichnet. Die eingereichte Arbeit ist weder vollständig noch in wesentlichen Teilen Gegenstand eines anderen Prüfungsverfahrens gewesen. Sie ist weder vollständig noch in Teilen bereits veröffentlicht. Die beigefügte elektronische Version stimmt mit dem Druckexemplar überein.[1]

(Ngoc Khanh Van Hoang)

[1] Non-binding translation for convenience: This thesis is the result of my own independent work, and any material from work of others which is used either verbatim or indirectly in the text is credited to the author including details about the exact source in the text. This work has not been part of any other previous examination, neither completely nor in parts. It has neither completely nor partially been published before. The submitted electronic version is identical to this print version.

Contents

Abstract

While emotion generation in dialogues for empathetic systems has received special attention from the research community, emotion transfer in texts is a relatively new task. This thesis aims to explore the methods to emotionalise the texts while ensuring fluency and preserving the original semantic meaning. Instead of using unsupervised methods, together with a data-driven approach to the problem of *"style"* and *"content"* as it is normally pursued in literature, we attempt to differentiate the two terms. Our effort, thus, leads to a parallel neutral-emotional corpus. Two Transformer-based sequence-to-sequence architectures are adopted for the implementation of our text emotionalisation models. An additional emotion style loss is employed to guide the generation towards more emotional words and phrases. Before fine-tuning the pre-trained sequence-to-sequence models into the emotionalisation models, we first train them on paraphrase data to refine their re-writing capacity and thus improve the preservation of original content in the generated candidates. The encouraging results of our initial experiments suggest the potential of our approach. Despite having a small-scale corpus, the models are able to emotionalise the input text. The ablation studies are further conducted to understand the contribution of two architecture designs, namely the emotion style loss during training and the pre-training paraphrasing stage. However, both automatic and human results show that their contribution is modest and unclear. We believe a more comprehensive evaluation is needed to investigate this issue further.

Acknowledgements

I would like to give my most sincere thanks to my supervisor at Sony, Fritz Hohl. Without his endless support and the AI, Speech and Sound Group at Sony Stuttgart, this thesis, or even the topic, will not exist. I also want to express my gratitude to Prof. Thang Vu for his agreement on being my university examiner. And lastly, I appreciate all encouragement and advises from my friends.

1 Introduction

Emotion recognition is a key component in the development of human-machine interaction. Emotion-aware dialog systems would help strengthen its connections with human users and increase their engagements into the conversations. This, thus, has lead to an increasing interest in incorporating emotions into dialog systems (Ma et al., 2020).

Affective natural language generation is concerned with the task of generating texts which reflect emotional states of the producer or influence that of the receiver (Gatt and Krahmer, 2018). Recent research on this task has focused on response generation in dialog systems. Given an user utterance, the model should learn to generate responses according to the user emotional states. For example, if the listener feels *sad*, the system should not answer in *neutral* or *happy* manner. This task, however, is different from the *emotion style transfer* problem, which aims at changing the style in, or attributes of, a word sequence while keeping its semantics unchanged. On the other hand, for dialog generation, there are countless ways to respond to an utterance; therefore, preserving the original content is not a requirement.

In Text-To-Speech systems (TTS), emotions are manipulated via prosodic features. Therefore, the same words can evoke different emotions depending on how one speaks. For example, *Okay* can convey *happy*, *angry*, or *neutral* feelings (Hsu et al., 2018). However, the inter-annotator agreement in emotion recognition is low, especially if it is done via text or speech only (Cavicchio and Poesio, 2008). In other words, for a successful transfer, it might not be enough to control the emotions solely via prosody. Multimodality facilitates the detection of emotions in conversations (Poria et al., 2019).

The objective of this thesis is to investigate the transformation from neutral textual utterances into emotional ones as a pre-processing step for TTS systems. We call it *"text emotionalisation"* and consider it as a subtask of emotion style transfer in text. We explore the task with the question of what makes an utterance to have a specific emotion. A manual inspection of the DailyDialog corpus (Li et al., 2017) leads us to an observation that an utterance can be of an emotion due to three reasons: (1) it explicitly expresses the emotional state of the speaker (e.g. *"This is awesome!"*); (2) it suggests an emotion (e.g. *"Come home late one more time, and you'll find your stuff outside."*); or (3) there is an emotion cause from previous utterances (e.g. *"Is it waterproof?"* is of *happiness* emotion because the speaker mentioned buying a new watch with *flashy red lines*.) (Poria et al., 2021). As our focus is on single utterances in dialogues, we believe the first and second reasons are more relevant to our project. Furthermore, taking into account the third reason would require the existence of the context of the utterance (i.e. the previous utterances in a dialogue).

In textual language, one of the strategies to put feelings into words is to ex-

plicitly use emotional words to express feelings. In style transfer task, such words are called style or attribute markers and the task is carried out by changing the markers of one style to another. Our belief is that the same strategy can be applied to transforming neutral to emotional sentences by incorporating the emotional words or phrases. Nevertheless, we hypothesise that such transformation is only possible if the models possess a deep understanding of the topics being discussed (e.g. for the correct usage of *"delicious"*), and syntax (e.g. for the paraphrase from *"I live here"* to *"It is nice living here"*). Furthermore, it is unclear whether any neutral sentence can be converted to any emotion.

Our main contributions include the creation of a parallel neutral-emotional corpus and the implementation of text emotionalisation models using a sequence-to-sequence framework. Given a neutral utterance and a target emotion, the models generate outputs with the target emotion while preserving the original meaning and ensuring the fluency of the texts. Our main experiments utilize the loss from emotion classification on the output to guide the decoding towards more emotional tokens, and a paraphrasing stage as an intermediate training step to boost the preservation of the original content. During inference, filtering and ranking methods are adopted to find the best candidate among all the generated outputs. The generated sentences might carry the expected emotions at the cost of changing too much original content. Furthermore, the outputs might not achieve the same level of fluency as if they were written by a human. Therefore, our evaluation methods aim to verify the expected emotions in the outputs while ensuring the original content is well kept and the wording sounds natural. We perform both automatic and human evaluation. Our further analysis into the evaluation metrics shows room of improvement in the scoring and selection of the generated outputs.

In this thesis, unless their difference is explicitly emphasised, the two tasks *"text emotionalisation"* and *"(emotion) style transfer"* are used interchangeably as our goal is to develop a model to transform neutral sentences into emotional ones. Occasionally, two terms *"emotion"* and *"style"* refer to the same concept though *"style"* conveys a more general meaning.

In addition to the introduction in this section, and our conclusion in Section 12, the thesis is divided into three main parts with details as follows.

- Section 2 introduces the background knowledge in emotions and two related tasks in affective natural language generation, namely dialog generation and text style transfer. In Section 3, we will formally define our emotionalisation task and set our goals for the thesis. After that, Section 4 explains briefly our approaches to the task, from defining the "emotion style" to creating the corpus and implementing the models.

- The next three sections present the datasets, the models, and the evaluation metrics. Section 5 illustrates our efforts in creating a parallel

neutral-emotional corpus from publicly available emotion corpora. The model architectures, the training procedure, and generation process are explained in Section 6. Section 7.1 explains the metrics adopted for our task.

- The last four sections describe our experiments and discuss results and result analysis. We will present the baseline and all the models configurations in Section 8.2. Section 9 shows our main experiments and results, including several ablation studies to understand the impact of architecture design on the performance of our models. Additionally, results on human evaluation are presented in Section 10. Lastly, Section 11 discusses further our thoughts on common metrics of style transfer task, the definition of style and content, as well as the possibility to adopt our models for the EST task.

Our main contributions include:

- defining *"emotion style"*, and its difference to *"emotion content"*.
- creating a small-scale parallel neutral-emotional dataset based on the aforementioned definition.
- illustrating that text emotionalisation can be performed with such small-scale corpus with promising initial results, using Transformer-based sequence-to-sequence architectures.

The codes, the models, the data, and the outputs can be found at our github repository at `https://github.com/VanHoang85/text-emotionalisation`.

2 Related Work

2.1 Emotion Representations and Emotion Recognition

Rooted in psychology, computational models represent emotions in one of three categories: the dimensional approach, the discrete approach, and the appraisal approach (McTear et al., 2016; Ma et al., 2020). In the dimensional approach, emotions are represented as a 3D vector space of arousal, valence, and dominance. Research in linguistics has shown that these dimensions represent the most important lexical aspects, capturing almost 70% of the variance in word meanings (Osgood, 1952; Osgood et al., 1957). In the field of automatic emotion recognition, a vector representation in this three-dimension space is believed to belong to one of the defined emotion categories (Abbaschian et al., 2021). The discrete approach, however, aims to identify a group of *basic* emotions, based on different criteria. Two most widely used

theories are developed by Ekman (1992) and Plutchik (1980). According to Ekman (1992), the six fundamental emotions are anger, disgust, fear, joy, sadness, and surprise, corresponding to universal facial expressions. Plutchik (1980)'s Wheel of Emotions includes two further emotions, acceptance and anticipation, based on adaptive biological processes. The last approach is called the appraisal theory, which claims that emotions are caused by what one perceives, or appraises, as the changes in their environments with respect to their goals and expectations (Scherer, 2005). In other words, emotions are the results of our evaluation of objects and events that we have experienced. In recent years, with the rise of distributional representation methods and deep learning, attempts have been made to encode more emotional information into the embeddings (Agrawal et al., 2018; Wang and Zong, 2021).

Emotion recognition in speech and language processing has long established itself as a classification task, aiming to assign a discrete emotion category to a given input. In text, earlier works exploit emotion dictionaries, which associate each word to its affect meaning (Mohammad and Turney, 2010). The underlying assumption is that emotions can be expressed via words. For example, *delightful* indicates *joy* while *heart-wrenching* conveys *sadness*. Under a machine-learning setting, the model is able to learn the association between words and emotions via a rich set of hand-crafted linguistic features (Alm et al., 2005). Recent advances in deep learning have lead to the dominance of word embeddings and large pre-trained language models in the research community (Acheampong et al., 2021).

If emotion analysis in text is concerned with *what is said*, the emphasis on the speech side is more about *how it is said*. In speech, emotions can be expressed by modifying the prosody, including pitch, intensity, and duration. Therefore, the same sentence can imply either *joy* or *anger* depending on how speakers choose to express themselves. These prosodic cues are served as features in speech emotion recognition. Two major approaches include recognizing based on the three dimensions of emotions (i.e. arousal, valence, and dominance) with the use of a hierarchical classifier, and using statistical pattern recognition techniques from machine learning field (Abbaschian et al., 2021).

2.2 Dialogue Generation with Emotions

Coined by Rosis and Grasso (2000), Affective Natural Language Generation (NLG) aims to generate texts with elements that can influence emotions and attitudes of the listeners. However, the definition of the term has been extended: texts with affect can also reflect the emotions of the producer (Gatt and Krahmer, 2018).

Though an overlap between Affective NLG and Style Transfer in text exists, the key difference is that while TST also tries to preserve the content of the

original input sentence, Affective NLG focuses on incorporating emotions into the output sentence in an appropriate way. Goswamy et al. (2020) adapt a transformer-based language model to generate texts based on a given emotion category, its intensity, and a topic. To ensure that the generated texts do include emotions, the authors utilize the NRC Emotion Intensity Lexicon (Mohammad, 2018), adjusting the sampling method to force the model to generate more emotion-relevant words. A large body of research in Affective NLG is done in the development of emotion-aware dialog systems.

An affect loss function is used by Asghar et al. (2018) to control the intensity of the intensity of the emotional responses. Their model, however, is unable to generate the responses with a desired emotion. Inspired by Pointer Networks, Dryjanski et al. (2018) implement a bi-directional GRU to infer which emotion phrase should be inserted into which position in the input sentence. Zhou et al. (2018) use an internal memory to capture the dynamics of the internal emotion state during decoding while the external memory works similar to an emotion lexicon, influencing the sampling process. Colombo et al. (2019) incorporate the emotional information also during the encoding process. The target emotion is represented as a probability distribution over Ekman's six basic emotions, obtained by an emotion classifier. On the other hand, Song et al. (2019) equip the decoder with a lexicon-based attention mechanism so that the emotional words can be *plugged* into the generated texts at the right time steps. Furthermore, during training, an emotion classifier helps to guide the generation process by increasing the intensify of the emotional expressions.

2.3 Text Style Transfer

2.3.1 Definition

Text Style Transfer (TST) can be defined as a subtask of Natural Language Generation (NLG) in which an input word sequence is re-written in a different style while its original content is preserved (Hu et al., 2020; Gatt and Krahmer, 2018). TST revolves around two components, namely style and content. The definition of *style* is not well-defined, often adopting a *data-driven approach in defining text style* (Hu et al., 2020; Jin et al., 2020). In other words, *style* refers to the text or label attributes of style-specific corpora. If the corpus is a collection of Shakespeare works, then it is Shakespeare style. From private messages on Facebook and Whatsapp, we can obtain texts of informal style. Reviews with one or two stars out of five are of negative style while those with more than four stars are considered as positive style. Additionally, each *style* should be realized via its attributes, which are the values belonging to a specific style (e.g. negative attributes can be expressed via the word *terrible*). *Content* is normally understood as the underlying, normalized

meaning without the presence of style attributes.

The most popular benchmarks for general purpose TST are formality transfer and sentiment transfer. The former refers to the task of adjusting the level of formality: from *formal* to *informal* style and vice versa. For example, in the *formal* style, one is expected to write full words (e.g. "cannot", "television") instead of abbreviated forms (e.g. "can't", "TV"). The later, sentiment transfer, aims to change the sentiment polarity in text. One possible topic is to change the movie reviews from *positive* to *negative* polarity and vice versa (Hu et al., 2020).

With respect to *emotion* style, most works have been done in the context of emotion-aware dialogue systems (See Section 2.2). To the best of our knowledge, the task of emotion transfer in text is formally proposed by Helbig et al. (2020). That is, given an input sequence with one emotion, the model should output another sequence with another emotion. Compared to other style transfer problems, this task is non-binary (i.e. there are more than two emotions) and concerns both style and content (Helbig et al., 2020).

2.3.2 Main Approaches

Due to the scarcity of parallel data, the majority of works in TST adopt unsupervised methods and exploit non-parallel, or mono-style, corpora. Three main approaches are (1) disentanglement at embedding level, (2) prototype editing at surface structure, and (3) pseudo-parallel corpus creation (Jin et al., 2020).

Disentanglement of style and content at embedding level: Earlier works on disentanglement have adopted methods originally developed for machine translation (MT). Prabhumoye et al. (2018) uses back-translation technique to learn a latent representation of the input word sequence, trying to retain the content while reducing its original stylistic characteristics. Then, separate decoders are utilized to generate style-specific text using the disentangled representation. Later methods have been developed to eliminate the needs for parallel stylized corpora (Shen et al., 2017), borrowing the Generative Adversarial Networks (GAN) from Goodfellow et al. (2014). In a purely unsupervised manner, one approach proposed by Dathathri et al. (2020) is to exploit a big pre-trained language model, combined with one or more attribute classifiers to guide the text generation process.

Prototype editing at surface structure: In contrast to an end-to-end framework in which the disentanglement of style from content is performed in a black-box system, prototype editing aims for a more transparent approach consisting of several steps (Li et al., 2018; Xu et al., 2018; Sudhakar et al., 2019; Helbig et al., 2020). These steps can be performed via either rule-based or machine learning systems. The first one is to identify and remove the mark-

ers of the original style in the input word sequence. One popular method is to utilize a style classifier with the attention mechanism to select the tokens with high attention weights as the attribute markers of an input sentence. This method works on the assumption that the higher the weights of the tokens are, the more relevant they are to the classifier, and thus, the more *style* information they carry with them. The second step is to retrieve the attribute markers of the target style. This can be achieved by finding the sentence of the target style which is most similar to the sentence of the source style and extracting the target attribute markers. Finally, the target sentence is generated in an autoregressive manner, conditioned on the retrieved markers and the input sentence while ensuring that the generated text is grammatical and fluent. The last step has been recently framed as a text-infilling task to make use of the learning objective of Masked Language Models (MLM) (e.g. BERT) (Wu et al., 2019; Malmi et al., 2020). The attribute markers are replaced by `[MASK]` tokens, which the MLM needs to fill in by predicting the most suitable words or phrases given the context and the target style. Similarly, Reid and Zhong (2021) use a masked sequence-to-sequence (seq2seq) model to fill in the phrases of the target style. However, they propose to identify the `[MASK]` tokens by training a token tagger to predict the Levenshtein editing operations (i.e. insert, keep, replace, and delete) to transform the source to the target text.

In addition to being more transparent and interpretable, it has been argued that performing style transfer in a pipeline fashion helps to preserve the original content better compared to an end-to-end black-box system (Xu et al., 2018; Reid and Zhong, 2021). Under this approach, as the selection of stylized words is of utmost importance, attempts have been made to improve it, for example, exploiting attention mechanism (Sudhakar et al., 2019). Yet, selection methods still leave much to be desired.

Pseudo-parallel data: The research community has focused mainly on purely unsupervised methods due to the lack of parallel data, leveraging huge mono-style corpora using GAN model architecture to learn the style attributes. However, it is not surprising that the models achieve best performance when being trained on parallel data (Lai et al., 2020). Therefore, efforts have been made to create pseudo-parallel data from large mono-style corpora via automatic methods.

Reformulating TST as a paraphrase generation task, Krishna et al. (2020) aims to express the original sentence in different ways via two paraphrase models. Their approach works on the assumption that the paraphrasing step produces a normalizing effect, stripping away significant indicators of original style. The pseudo-parallel data is derived by first neutralizing sentences of style s through a paraphrase model, forming a pseudo-parallel corpus between original sentences and their paraphrased non-style versions. The style paraphraser is obtained by training another paraphrase model to convert the paraphrased

sentences from the first step back to their original stylized sentences.

Additionally, paraphrasing has been proposed as an intermediate pre-training step to help improve semantic preservation (Bujnowski et al., 2020; Krishna et al., 2020; Lai et al., 2020). Bujnowski et al. (2020) cast the task as multilingual translation, considering each style as a new language. They follow a multi-task setup, training a seq2seq model to jointly learn how to paraphrase and how to paraphrase with a specific style. Their results show that the model performance is still acceptable despite a small training dataset of just 1k parallel sentences. Interestingly, the bigger the training data, the higher the Style score but the lower the Content score.

Recently, it has been argued that emotion attributes are lost during a back-translation process (Troiano et al., 2020). In other words, one can imagine a similar approach to obtain a pseudo-parallel neutral-emotional dataset. However, Lai et al. (2020) argue that such paraphrasing approach might be more helpful to tasks involving pure style pure (i.e. formality) rather than those having ambiguous boundary between style and content (i.e. sentiment). Therefore, in an effort to create pseudo-parallel data for sentiment transfer task, resources such as WordNet are exploited. Using sentiment lexicons, high-scoring sentiment words in the sentences are detected and then replaced with their antonyms. After that, an iterative back-translation strategy is adopted to augment more high-quality parallel data to fine-tune the style paraphraser.

However, their approach might not be ideal for the emotion transfer task. The reason being, emotion transfer is a non-binary task (Helbig et al., 2020), not a polarity one such as sentiment. Therefore, it is not clear-cut to determine an opposite emotion (i.e. what is the opposite emotion of *"happiness"*? *"Anger"*? *"Sadness"*? *"Disgust"*?). One can resort to adding the emotion *"neutral"* and creating neutral-emotional data. Li et al. (2018) and Xu et al. (2018) remove the markers of one style make the original sentences neutral. However, the neutralized sentences then become ungrammatical.

One closely related task to Emotion Style Transfer is called phrase insertion (Dryjanski et al., 2018): given a neutral sentence, it learns to infer where to insert which emotional phrase to maximize emotion in the text while ensuring its naturalness. Their model is trained to perform jointly binary classification on whether to insert at a certain position and multi-label classification on which phrase to insert.

2.3.3 Issues

TST approaches assume that there exists a clear separation between style and content and the key is to develop a method for such separation. However, questions have been raised about whether this is possible (Lample et al., 2019). An inverse correlation between transferring to a new style and preserving the

original content has been observed (Mir et al., 2019; Pang and Gimpel, 2019; Pang, 2019a; Bujnowski et al., 2020; Helbig et al., 2020), suggesting a trade-off between the two objectives of TST. As a result, some have called for a move from an ad-hoc, "operational" definition of *style* to explore further the "real-world" relationship between style and content (Pang, 2019b,a; Troiano et al., 2021).

Xu et al. (2018) have noticed that the majority of research in TST only succeed in transferring to another style while failing to preserve the original semantic content, causing the system to *"hallucinate"*. For example, with *"The food is delicious"* as input, the model generates *"What a bad movie"* as output. Our assumption that this hallucination tendency is caused by the model's failure to separate style from content. If the word *"food"* appears frequently in sentences associated with positive sentiment while *"movie"* shows up mostly in negative sentences, the model might possibly pick up that undesirable connection.

3 Task Definition and Goals

This section defines the task of text emotionalisation and establishes goals and research questions for the project.

3.1 Text Emotionalisation

Figure 1: Emotionalising the neutral texts into texts with the target emotions

We define *text emotionalisation* as the task to transform neutral utterances into emotional ones given target emotions as illustrated in Figure 1. This new task can be considered as a subtask of Emotion Style Transfer (EST) (Helbig et al., 2020), which aims to change the emotion of the input text into a target emotion. In EST, the first step is to identify the part of the text which contains the original emotion and replace it with phrases of another emotion. As for

our emotionalisation task, since our input texts are neutral, it mainly involves mainly the second step, searching for suitable emotion phrases and, if necessary, rewriting the input into a fluent text containing the target emotion. Under this definition, our task is similar to the works by Dryjanski et al. (2018), who emotionalise the texts via a pure insertion operation of emotion phrases.

Similar to the Style Transfer task, our evaluation will be performed on (1) transfer accuracy (i.e. whether the outputs contain the target emotion), (2) content preservation (i.e. whether the outputs preserve the original content), and (3) fluency (i.e. whether the outputs are natural-sounding and grammatical).

3.2 Goals

Following the separation between two communities, namely the speech and the text, each adopts a different approach to the emotionalisation of an utterance, namely altering its prosodic features and modifying its lexical units. The former aims to control *"how one speaks"* while the latter specifically targets at *"what one says"*. In speech synthesis, emotional style transfer has become a popular research area. On the other hand, on the text side, this task is relatively new. To the best of our knowledge, only a handful of works deal with modifying emotions in texts (Dryjanski et al., 2018; Smith et al., 2019; Helbig et al., 2020; Sharma et al., 2021). However, these two approaches are complementary to each other.

Humans, undoubtedly, partly express their emotions by controlling their prosody. Therefore, the same utterance can convey different emotions depending on how one says it. The difference between the expressed emotions is subtle, especially if they are compared with neutral sentences[2][3]. Emotion recognition has a low inter-annotator agreement, even higher under the condition of modality deprivation (Cavicchio and Poesio, 2008). This is due to the fact that conversations, by nature, are multimodal. Facial expressions, prosody, language, and even gestures are essential for the detection of emotions. When the cues from one modality are not reliable, one often resorts to other modalities to make correct predictions (Poria et al., 2019). Our belief is that what one says does contribute greatly to the expression of emotions.

The objective of this thesis is to investigate the possibility of modifying the input texts to incorporate emotions in advance before feeding them into TTS systems for the production of speech outputs.

Song et al. (2019) suggest two methods to put feelings into texts. One is to use strong emotional words in an explicit manner to express one's emotional states (e.g. saying *"I'm delighted that"* to express *happiness*). Another is

[2] `https://kunzhou9646.github.io/controllable-evc/`

[3] `https://ttslr.github.io/i-ETTS/`

to combine neutral words in distinct ways for an implicit expression of emotional experiences. Our project adopts the former approach to emotionalise the texts as we believe the latter requires a comprehensive analysis into language structure and meanings. By choosing the right emotion phrases, the listeners should be able to understand the expressed emotions of the speakers.

Such transformation proves to be challenging. First, one emotion phrase suitable for a context does not necessarily fits into another one. Exclamations such as *"wow"* for *happiness* and *"damn"* for *anger* seem to be able to fit well into any context. However, the phrase *"absolutely delicious"* goes with an utterance about food or drinks, but not for weather or persons, for which the phrase *"so lovely"* would be a better choice. Due to the non-existence of universal emotion expressions, the models would need to understand the topics which are being discussed. Additionally, they need to be aware of the target of the conversation to correctly add the phrase *"I'm happy for **you**"*, or *"for **her**"*, or *"for **him**"*. Exclamations such as *"yay"* and *"oh my good"* require no such understanding of semantics. However, if the models keep overusing them, the transformation would be monotonous. Second, the transformation of any neutral utterance to any emotion might not be achievable. Let us take the sentence *"you are late for five minutes"* as an example. Adding the phrase *"amazing!"* does not necessarily make it convey *happiness* emotion but make it sound more sarcastic. We believe that a sentence, though annotated as being neutral, can have underlying emotional readings, which might require deeper insight into its context. Lastly, language fluency and diversity should also be taken into account if the models wish to perform beyond pure insertion either at the beginning or the end of the utterances. Given the sentence *"I saw you at the party"*, the transformation to *"It was nice seeing you at the party"* requires knowledge of both syntax and semantics.

This thesis include the implementation of text emotionalisation models and an investigation into the possibility of emotionalising neutral sentences. Our main research questions include (1) *what is style and content in the Emotion Style Transfer task*, and (2) *is it possible to transfer the emotion of an utterance without changing their original meaning*.

4 Our Approach

This section provides readers with our definition and examples of emotion *"style"*, including the difference between emotional words and emotion-bearing events (Section 4.1). After that, we will briefly describe the process to create our parallel neutral-emotional data (Section 4.2) and our approach to the task of text emotionalisation (Section 4.3).

4.1 Defining Emotion "Style"

Section 2.3.3 raises the potential issues of having an "operational" definition of *style* by adopting attention mechanism and/or lexicons. We ask ourselves, *"What is Emotion Style?"* Our investigation into emotion lexicons such as WordNet-Affect (Strapparava and Valitutti, 2004) shows us two types of emotion-evoking words, namely emotional words and emotion-bearing words. The former is a direct expression of one's feelings (e.g. furious, excited, depressed) while the later refers to events which can cause some emotional reactions from the listeners. For example, the words *"birth"* and *"death"* are normally associated with the emotions *"happiness"* and *"sadness"* respectively. However, an unwanted *"birth"* can lead to *"sadness"*. A *"death"* can also be discussed in an *"happy"* manner if it is a peaceful and natural death after a long struggle from illness.

Our project defines style-related words as explicit emotional words and content-related words as the remaining words in the texts, including emotion-bearing events. As mentioned in Section 3, one of our motivations for the thesis is to explore the question *what should be considered as style and content in the Emotion Style Transfer task*. To reach this goal, we create a new parallel neutral-emotional corpus. Our belief is that the manual creation of a parallel corpus would allow us to examine closely the emotion style attributes and the possibility to transfer sentences to a certain emotion without changing its original content. We acknowledge that the boundary is hard to define. For exclamations such as *"wow"*, *"ugh"*, and *"damn"*, or comments such as *"i'm glad"*, *"this is awful"*, and *"it's sad"*, one can argue that they are emotion expressions about some event. Thus, the neutralisation process works by removing such expressions. However, what about the phrase *"I hate to tell him that..."*? How much of original content is lost when we discard *hate to*? And whether the transferred emotional sentence fits into context of a dialogue?

4.2 Creating Parallel Data

We collect 5 datasets annotated with emotion labels. The number of emotion labels for each dataset differs, ranging from 6 to 32. Our project uses 4 emotions, namely anger, happiness, sadness, and neutral. Therefore, the labels which are not needed for our project are either discarded or mapped to our existing emotions if such mapping is provided by the dataset creators. As briefly mentioned in the previous section, an utterance can be annotated with a certain emotion due to its emotion-bearing events (e.g. *"marriage"* and *"divorce"*). What we want to come up with is utterances with emotional words such as *"I'm glad"* and *"it saddens me"*, which explicitly express emotional states. To collect such phrases for each emotion, we select the **DailyDialog** corpus (Li et al., 2017) and start the annotation process. For each emotion, we

annotate the span of texts containing emotional words, which we call "emotion phrases". These phrases serve as initial "seed" patterns to determine the syntactic structure surrounding emotional words (e.g. *"kind"* as an adjective and not a noun). Using a pattern matching system, we can filter out undesirable utterances (e.g. *"What kind of person is she?"*).

Our rule-based pattern matching system is developed using the `Matcher` system of `spaCy`[4]. To minimize our efforts in writing the patterns, we cluster the emotion phrases based on syntax. The intuition is that many phrases have similar structure, only differing in one or two tokens. That means we only need one pattern which is flexible enough to cover similar phrases. For example, phrases *"i am amazed that"*, *"this is perfect!"*, *"which was really wonderful"* all have the same sequence of part-of-speech tags (i.e. *pronoun-be-adjective*) obtained by spaCy Tagger[5]. Therefore, there is no need to write one pattern for each phrase. One can cover all similar phrases and it should also specify several optional lemmas and/or part-of-speech to enable the recognition of, for example, *really* in the third phrase.

Using the matching system, we are able to get utterances containing emotional words from the remaining 4 datasets. The next step is to neutralise the emotional utterances to create parallel sentence pairs. The neutralisation process is done manually on a subset of collected emotional utterances. From this step, we obtain more emotion phrases and refine our patterns further for false positive and false negative cases. Our aim is to obtain roughly 1-2k training samples for each emotion. This number is based on the works by Bujnowski et al. (2020), who argue that the model performance appears to be fairly reasonable even if only 1k samples are used for the training. In other words, this step aims at obtaining a small but high-quality parallel corpus of neutral-emotional sentence pairs. Section 5 provides details about our data creation procedure.

Our corpus consists of corresponding neutral sentences to the original emotion-labelled sentences. The `Emotionalisation` model is fine-tuned on this corpus (Further details in the next Section). We hypothesise that it is theoretically possible to develop an additional `neutralisation` model using the same corpus. This model will enable us to perform an automatic neutralisation process on the remaining unprocessed emotional utterances. We discuss this possibility in details in Section 11.1. Our assumption is that low-quality results can be filtered out and only the high-quality ones are kept. This iterative process can go on until there is no further training data is obtained. Inspired by Lai et al. (2020), we believe that this data augmentation technique can be adopted to create more training data for the emotionalisation model if necessary.

[4] `https://spacy.io/api/matcher`.

[5] `https://spacy.io/api/tagger`

4.3 Paraphrasing For Style

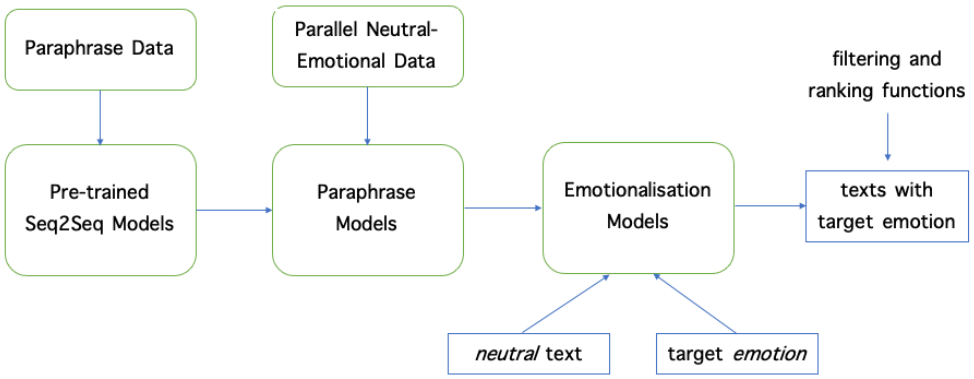


Figure 2: An overview of our training procedure from pre-trained models to emotionalisation models. The last stage shows the generation process.

For style transfer task, the generated texts, besides containing the target style, need to meet two further requirements. They should be (1) naturally sounding, and (2) semantically similar to the original content. The second requirement remains a challenge, especially for the tasks in which the boundary between style and content is unclear. Another issue arises from the fact that our parallel manually-created corpus would have at most 6k samples, meaning that our training is performed under low-resourced settings. To tackle this issue, paraphrasing is adopted as an intermediate pre-training step to improve semantic preservation. An empirical analysis by Bujnowski et al. (2020) shows that the model can still achieve reasonable performance even if the training dataset per style consists of 1k parallel samples only. In other words, style transfer can be considered as a special way of paraphrasing (Krishna et al., 2020; Lai et al., 2020).

Our implementation adopts Transformer-based sequence-to-sequence models. Thus, the texts are to be generated in an autoregressive manner. Such autoregressive generators are prone to "hallucinations", referring to the situation in which outputs are irrelevant to input texts. In style transfer tasks, we believe this issue is due to an ambiguous definition between style and content. Bujnowski et al. (2020) encounter similar issue despite having high-quality parallel data. However, the reason might lie in their low-quality paraphrase data instead. Therefore, we adopt several filtering methods to guarantee the quality of our paraphrase data (Further information can be found in Section 5.4). Paraphrases are sentence pairs which are similar in semantics but different in lexicons and syntax. As mentioned in Section **??**, paraphrasing is adopted as an intermediate training stage to improve content preservation.

An overview of our training procedure is shown in Figure 2. Large pre-trained sequence-to-sequence models are trained further on paraphrase data to ensure that the models know to re-write the original content. In the next stage,

the paraphrasers are fine-tuned on our parallel corpus to learn the connection between the target emotion word and its corresponding emotion phrases. Bujnowski et al. (2020) treats the *paraphrasing for style* task as a multi-task learning problem. Their model learns jointly how to generate paraphrases and transfer styles in a same framework. We, however, opt for the sequential transfer learning technique to learn first paraphrase generation and then style transfer as done by Krishna et al. (2020) and Lai et al. (2020). Though the two methods are closely related, multi-task learning is expensive for the entire training process (Ruder, 2019). On the other hand, sequential learning shifts the cost effectively to the paraphrase generation stage, enabling us to easily perform different experiments on the later fine-tuning step.

The emotionalisation models are obtained after fine-tuning the paraphrase models on our parallel corpus. The input to our emotionalisation models is a concatenation of the neutral utterance and the target emotion. During generation stage, we get a list of texts containing the target emotion. The next question is how to select the best candidate texts. Three scoring methods, corresponding to three most important requirements in style transfer task, are adopted for the scoring of the generated outputs. They include an emotion score (i.e. transferring to target style), a similarity score (i.e. preserving original content), and a fluency score (i.e. sounding natural). The candidates with scores lower than certain thresholds are filtered out. For the remaining texts, we rank them based on a combination of these scores. Further details can be found in Section 6.5.

5 Datasets

In this section, we introduce the list of corpora with gold emotion labels (Section 5.1). As an utterance can belong to an emotion category based on the context or emotion-bearing events, different filtering methods are performed to discard such utterances (Section 5.2). The remaining utterances should contain emotional words to enable the creation of the neutral utterances in our `Emotionalisation` dataset (Section 5.3).

5.1 Emotion Datasets

Instead of creating a new corpus from scratch, we adapt the publicly available datasets annotated with gold emotion labels accordingly to suit our needs. Below is the list of emotion datasets chosen for our project. The number of emotion labels is different for each dataset. Thus, for the labels which are out of scope of our project (e.g. fear, optimism, remorse), we either discard them completely or map them into our label list (i.e. anger, happiness, sadness) if such mapping is provided by the dataset creators.

DailyDialog (Li et al., 2017) is a multi-turn dialogue dataset consisting of conversations about daily life. Instead of collecting data from social media or movie subtitles, the authors crawled English learning websites to obtain the dialogues that are most useful to English learners to practice their language skills. They are human-written and focus on a certain real-life scenario, and thus, contain a natural flow of human communication. Furthermore, each utterance is manually labeled with an emotion. The dataset consists of 13,118 dialogues and has seven emotion categories: Anger, Disgust, Fear, Happiness, Sadness, Surprise, and Neutral.

MELD (Poria et al., 2019), an extension of **EmotionLines** (Hsu et al., 2018) which consists of dialogues from scripts of *Friends* episodes. The emotion annotation of **EmotionLines** was performed by having five workers from the Amazon Mechanical Turk (AMT) platform read the *textual* dialogues and decide on an emotion for each utterance. For **MELD**, the utterances were annotated by three workers both reading the scripts and watching the videos at the same time. Therefore, the inter-annotator agreement of **MELD** is higher than that of **EmotionLines**. The multimodal dataset contains about 13,000 utterances from 1,433 dialogues annotated with three sentiment classes (Negative, Neutral, and Positive) and seven emotion categories (Anger, Disgust, Fear, Happiness, Sadness, Surprise, and Neutral).

GoEmotions (Demszky et al., 2020) is a large-scale dataset developed for the task of Emotion Classification. The authors crawled Reddit comments, carefully filtering offensive and harmful comments, and masking person names and religions. Three annotators were assigned for each comment, and if no consensus was reached, two additional raters were assigned. The final dataset consists of 58k comments, labeled with a fine-grained list of 27 emotion categories plus Neutral.

EmpatheticDialogues (Rashkin et al., 2019) aims at facilitating the development of emotion-aware dialogue systems by creating conversations grounded in emotional situations. Each conversation in the dataset was created by first having a worker describe a situation based on a given emotion label (e.g. proud), and then letting him or her chat with another worker about that situation. There are in total 32 emotional labels, covering a wide range of positive and negative emotions. The dataset comprises 24,850 conversations, gathered from 810 different participants.

CARER (Saravia et al., 2018) is a collection of English tweets, constructed by using distant supervision for the annotation of emotions and not by human annotation process. A total of 339 hashtags serve as noisy labels. For example, the tweets containing the hashtags *#depressed* or *#grief* are considered to be of emotion *sadness* while those with the hashtags *#fun* or *#joy* belong to *happiness* category. The dataset has around 20,000 samples annotated with one of six emotions: Anger, Fear, Joy, Love, Sadness, and Surprise.

5.2 Data Filtering

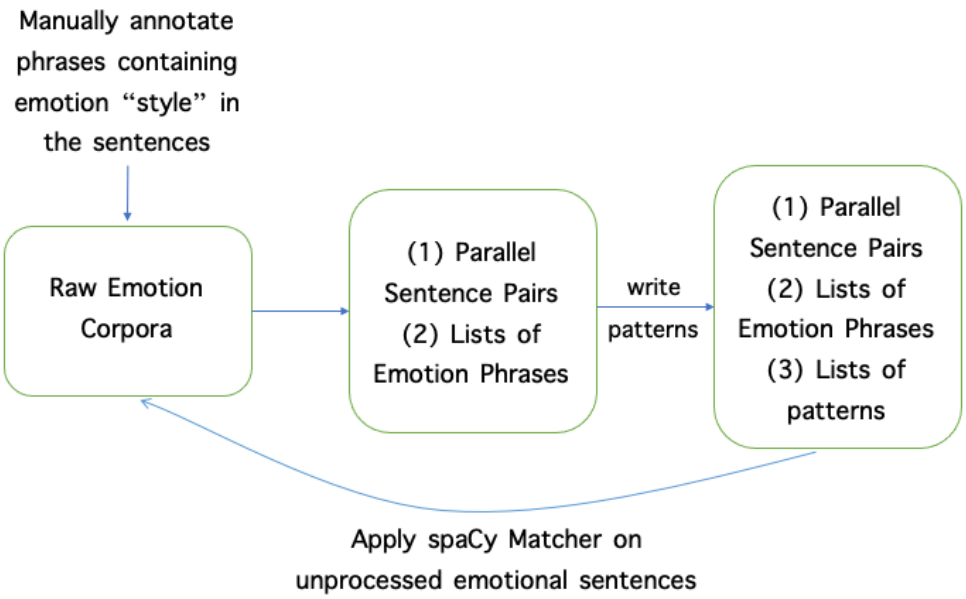


Figure 3: A broad overview of the steps to create a parallel neutral-emotional corpus. At first, we manually process a subset of the raw emotion corpora. Next, emotion patterns suitable for a rule-based pattern matching system are written using the initial lists of emotion phrases. Applying this rule-based system on the unprocessed sentences in the corpora, we automatically get sentences containing emotion "style" and not emotion "content". This iterative process of creation and refinement is repeated until we get enough parallel samples.

As mentioned in Section 3.1, an utterance can be of a particular emotion due to the context of previous utterances or emotion-bearing events. Our project, however, requires utterances with explicit emotional words/phrases. Previous works use emotion lexicons such as *WordNet-Affect* (Strapparava and Valitutti, 2004) and *NRC VAD Lexicon* (Mohammad, 2018) as the filtering step. However, upon close inspection, we discover that such resources have two characteristics unsuitable for our approach.

Firstly, they do not distinguish between emotion-bearing events and emotional words. For example, the *NRC VAD Lexicon* associates *marriage* with emotion *joy* and *divorce* with *sadness*. However, it is not entirely uncommon to talk about a *happy divorce* and a *sad marriage*. Such words are considered as emotion content and not emotion style. The second undesirable characteristic is that syntax can change the emotion association of the words in these lexicons. For example, the adjective *kind* denotes emotion *happiness*; however, in the sentence *"What kind of person is she?"*, *kind* as a noun evokes no such emotion, thus making the sentence a *neutral* question. Another example is the appearance of negation, which can easily change the emotion of the sentence

(e.g. *do not love*). As a result, words alone by themselves are not enough for filtering purposes. We need to know the syntactic structure of surrounding words to determine whether the utterance contains any emotional phrase. Therefore, we decide to build a pattern matching system, as further explained in Section 5.2.2, from a list of emotion phrases as initial seeds described in Section 5.2.1. Patterns can be understood as lists of token specifications (e.g. its part-of-speech, lemma).

5.2.1 Emotion Phrases as Pattern Seeds

We create one list of emotion phrases for each emotion *anger*, *happiness*, and *sadness* from the dataset **DailyDialog**. Table 1 shows the steps of obtaining emotion phrases from annotated emotional utterances. An initial parallel dataset of emotional and neutral emotions are created as a result. For example, for the utterance *"How dare you marry her?"*, we consider content as the fact that *"you marry her"* and *"how dare"* as style as the way the speaker expresses their anger. Thus, *"how dare"* will be added to our list of *anger* emotional phrases. We also collect one emotional-neutral sentence pair for our parallel corpus. Further details are described as follows.

annotation	parallel sentences
How dare you marry her?	How dare you marry her? (angry) You marry her. (neutral) → *anger* phrase: how dare
This is the **coolest** watch I've ever owned**!**	This is the coolest watch I've ever owned! (happy) This is the watch I've ever owned (auto) (neutral) I own a watch. (refined) (neutral) → *happiness* phrase: this is the coolest
It's a pity we can't afford a house.	It's a pity we can't afford a house. (sad) We can't afford a house. (neutral) → *sadness* phrase: it's a pity

Table 1: Examples of our annotation process to get emotion phrases and create parallel data from gold-labeled emotional sentences. Sentences are taken from **DailyDialog** corpus.

Annotation: For each utterance in the dataset labeled with one of the aforementioned emotions, we annotate the phrases in the utterance which express the emotion of the speaker, if such phrases exist. We try to keep our annotation strictly on the phrases that have little impact on the semantics of the utterances. However, as discussed in Section 4.1, we acknowledge that the boundary between emotion *style* and *content* is blurry, and such strict separation is difficult to keep throughout the annotation. For example, we consider

the phrase *This is highway robbery!* as emotion style if the conversation is about a man accusing another of ripping him off but as emotion content if it is about the actual robbery. Additionally, phrases such as *congratulations!* and *thanks!* are both counted as style though one can argue that they do convey the meaning of *showing one's happiness and/or gratitude to someone else*.

Initial Parallel Data Creation: From the annotated emotional utterances, the neutral utterances are first created in an automatic manner by removing the annotated phrases. This, however, leads to unnatural or ungrammatical neutral sentences (see example of *happy* in Table 1). In such cases, we try to rephrase the automatically-created sentences so that the neutralized utterances sound natural and grammatical. Another possible scenario is that the annotated phrase carries a large part of the sentence's content as in these examples *I'm glad for your child* and *Thanks for your concern*. Since it is almost impossible to rephrase the phrases *your child* and *your concern* into grammatical sentences without ignoring its original content, we simply remove them from our data.

Emotion Phrases Creation: With the parallel emotional-neutral data obtained from the previous step, we now proceed to get the lists of emotion phrases. These phrases are served as initial pattern seeds for the detection of emotional words in utterances. For each sentence pair, we use Levenshtein distance algorithm to get the edit operations to convert neutral sentences into emotional ones. Each emotion phrase of the sentence is a continuous sequence of tokens tagged with either *INSERT* or *REPLACE* operation. An example of the process is shown in Figure 4. In the step to create emotion patterns (Section 5.2.2), phrases containing no emotion words will be removed.

neutral						I	own	a	watch	.
emotion	this	is	the	coolest	watch	I	've	ever	owned	!
edit	[insert]	[insert]	[insert]	[insert]	[insert]	[keep]	[replace]	[replace]	[replace]	[replace]
phrases	(1) ***this is the coolest watch*** and (2) ***'ve ever owned***									

Figure 4: Obtaining emotion phrases from a pair of neutral-emotional sentences though Levenshtein operations.The second phrase is to be removed later since it evokes no emotion.

It is possible to get the list of emotion phrases directly from the annotation step. However, to err is human. There might exist some inconsistencies in our annotation. Furthermore, the annotations might fail to capture the patterns we need. In the second example in Table 1, having *coolest* as the emotion phrase is less helpful than *this is the coolest watch*. With the latter phrase, we can write a pattern such as ***pronoun–be–article–cool–noun*** to capture both the emotion word *cool* and its context. Otherwise, it is not very different from

directly using the word lists from the emotion lexicons. Table 2 lists several phrases for each emotion that are used to construct patterns described in the next section.

5.2.2 Emotion Pattern Matcher

```
pattern_1 = [{"POS": {"IN": ["PRON", "PROPN"]}},
             {"LEMMA": "be"},
             {"POS": "DET"},
             {"LEMMA": "cool",
             {"POS": "NOUN"}]

pattern_2 = [{"POS": {"IN": ["PRON", "PROPN"]}},
             {"LEMMA": "be"},
             {"POS": "DET"},
             {"LEMMA": {"IN": happiness_adjs},
             {"POS": "NOUN"}]

pattern_3 = [{"POS": {"IN": ["PRON", "PROPN"]}},
             {"LEMMA": "be"},
             {"POS": "DET",
             {"POS": "ADV", "OP": "?"},
             {"LEMMA": {"IN": happiness_adjs},
             {"POS": "NOUN"}]
```

Listing 1: Some possible patterns with *Matcher* system. Each pattern is written as a list of tokens. Each token is represented as a dictionary with token attributes as keys and token values as values. Token values can be of one value (e.g. one specific lemma or part-of-speech) or belong to a list of possible values (e.g. either pronouns (PRON) or proper names (PROPN)).

For the detection of emotion patterns, we use the *Matcher* system developed by spaCy[6] to match sequences of tokens, based on pattern rules. With *Matcher*, one can freely define the length of the patterns, and the attributes of each token in the sequence to suit their needs. Attributes can refer to the type (e.g., IS_PUNCT, IS_DIGIT) or part-of-speech tags of the tokens. Tokens can be matched with their lemma (e.g. be) or their exact spelling (e.g. am/is/are). Attributes can also specify that a token has to belong to a list of possible tokens or cannot be of a particular token. For example, with the phrase *this is the coolest watch*, one possibility is `pattern_2` in Listing 1 in which we require the appearance of one of the adjectives that evoke the emotion *happiness*. Or one can define pattern that demands that the token *not* should not stand before verbs of *happiness* emotion. Additionally, *Matcher* supports regular expressions to match different spellings of a word, as well as operators and quantifiers to define the number of appearance of a certain token (e.g. two adverbs).

[6] `https://spacy.io/api/matcher`

emotion	phrases
anger	it's disturbing that you moron i think i'll go crazy i am super mad that it creeps me out! no, for god's sakes, i can't stand it anymore! it drives me insane that for crying out loud! damn, it hurts! how dare you! to hell with that! that sucks!
happiness	i was so excited! that's fucking cool that i truly appreciate it. it is very generous of you isn't this lovely? wow, so cool. have the pleasure of it always amazes me that how wonderful it would be if you make my day! oh my god, how exciting! i couldn't be happier! thanks honey!
sadness	it's tough to see i feel so embarrassed that extremely miserable indeed i'm really sorry about and it's all my fault! oh, so bad. i'm afraid it pains me which makes me very disappointed what's so sad, is oh dear. well, sadly, how awful, poor you.

Table 2: Examples of phrases for each emotion.

Pattern Clusters: From Section 5.2.1, we obtain a list of phrases for each emotion. After inspecting the lists, we realize that there is no need to write a separate pattern for each phrase since some phrases would have exactly the same underlying grammar structures. The `pattern_2` Listing 1 can match *this is the coolest watch* but it can also match *that's great news*. Therefore, to reduce the number of patterns, all the phrases in a list are clustered based on the sequence of their part-of-speech tags.

Pattern Writing: During the writing process, when encountering a new cluster, we would prioritize refining an existing but similar pattern instead of writing a new one. For example, in Listing 1, the refined version of `pattern_2` would be `pattern_3` in which we add the possible appearance of adverbs before *happiness_adjectives*. Thus, `pattern_3` can match phrases such as *it is a truly amazing view* as well as phrases without adverbs. The list of *happiness_adjectives* is created based on the adjectives obtained from *happiness* phrases as well as the lexicons from **WordNet-Affect** (Strapparava and Valitutti, 2004). To avoid repetition in patterns, we would try to create as many lists as needed for words in the same position and/or part-of-speech tags. Some lists include *happiness_names* (e.g. honey, sweetheart, sweetie, darling) and *happiness_exclamations* (e.g. woah, horray, yay, cool). In total, we have 73 patterns for *anger* emotion, 59 for *happiness*, and 60 for *sadness*.

5.3 The `Emotionalisation` Dataset

As mentioned in Section 4, the manually-created parallel corpus `Emotionalisation` represents our effort to define what should be considered as *"emotion style"*. The four following emotions are included, namely neutral, anger, happiness, and sadness. The corpus consists of sentence pairs of neutral and one of the other three emotions.

Our annotation work on the **DailyDialog** dataset resulted in a small parallel neutral-emotional corpus. The numbers of sentence pairs are 130, 555, and 156 for anger, happiness, and sadness emotions respectively. From that, for each emotion, we get a list of phrases that we believe can transform a neutral sentence into an emotional one. With these lists of emotion phrases, we next proceed to create corresponding lists of emotion patterns for the rule-based matching system *Matcher* developed by **spaCy** (See Section 5.2.2).

Data for emotion style transfer: Applying *Matcher* on the remaining emotion datasets listed in Section 5.1, we now can easily filter out the data which, despite being manually labeled with an emotion, contain no emotion-evoking phrases. The process of creating parallel data is the same as described in Section 1: the neutralized sentences are first obtained by an automatic removal of the corresponding emotion phrases. Then, we go through a subset of the data and manually correct the neutralized sentences to turn them into naturally-

sounding and grammatical sentences as much as we can. In parallel, sentences that prove to be tricky for the neutralization process are discarded. For example, with the utterance *plus corporations are universally* ***unethical***, what should be its content? Is it simply the existence of corporations? Additionally, if false-positive and false-negative sentences classified by the *Matcher* are discovered, we refine our emotion patterns and run the system again in an iterative manner.

Data for emotion classifier: Data with *neutral* emotion is needed for the training our emotion classifier. To ensure data quality, we run our *Matcher* system on the sentences gold-labelled with *neutral* emotion and accept only those containing no emotion phrases whatsoever. As for emotion data, we take all the sentences labeled with the emotion and containing emotion phrases founded by our *Matcher*. To be more specific, the *anger* data should have at least one *anger* pattern, regardless of whether or not they can be neutralized.

Table 3 shows statistics of our `Emotionalisation` dataset, which contains data to train both our emotion style transfer and emotion classification models.

		emotion				
task	split	anger	happiness	sadness	neutral	total
emotion style transfer	train	1107	1235	1894	n/a	4236
	dev	196	218	334	n/a	748
	test	230	256	394	n/a	880
emotion classifier	train	5987	10853	10890	6875	34605
	dev	1417	2634	2587	1726	8364
	test	1759	3224	3213	2171	10367

Table 3: Data distribution for the *Emotionalisation* dataset for the training of the text emotionalisation and the emotion classification models. The unit of emotionalisation data is sentence pairs while that of classification data is single sentences.

5.4 Data for Paraphrasing Step

Paraphrases are sentences which express the same semantics but use different lexicons and syntactic structures. For example, two following sentences *"I heard you went out with him."* and *"I heard you guys were on a date."* are paraphrases. Our approach (See Section 4.3) adopts paraphrasing as an intermediate step for the preservation of semantics, especially our `Emotionalisation` is small.

"Hallucination" phenomenon, which refers to situations when the input and the output texts are irrelevant to each other, occurs in neutral-to-cute task by

Bujnowski et al. (2020) despite their model being trained on a parallel corpus. We assume the reason could be due to the data used for the training of their paraphrase models. One of their datasets is ParaNMT-50M (Wieting and Gimpel, 2018), which does contain sentence pairs with low similarity score and big difference in length (Krishna et al., 2020). They appear to be not paraphrases of each other. Therefore, we take extra measures to ensure the quality of our paraphrase data.

For the training of the paraphrase models, we use two datasets. First is a filtered version of PARANMT-50M, a corpus of backtranslated text (Wieting and Gimpel, 2018). After discovering the benefits of promoting lexical and syntactic diversity of the paraphrase pairs, Krishna et al. (2020) applied three main filters to the original PARANMT-50M and reduced the dataset from 5 million to 75 thousand pairs. The first filter aims at maximizing lexical diversity by removing sentence pairs with more than 50% unigram and trigram overlap. To promote syntactic diversity, the second filter discards pairs with lower than 50% reordering of shared words. And the last filter is meant to obtain only high-quality paraphrases by removing pairs with low semantic similarity scores.

The second dataset used for training our paraphrase model is PARABANK-2 (Hu et al., 2019), a huge corpus of nearly 100 million pairs developed for the task of paraphrase generation and detection. Similar to Krishna et al. (2020), we apply several filters on the raw corpus. First, very short and very long sentences are discarded. We only keep the pairs that are within a length between 7 and 25 tokens excluding punctuation. Furthermore, the length difference between the paraphrase pairs cannot exceed 5 tokens. Finally, we calculate the overlap of unigrams of the pairs and if more than half of the tokens of one sentence can be found in the other sentence, we simply remove these pairs.

Table 4 shows the final dataset used to train our paraphrase model, which consists of sentence pairs filtered from two paraphrase corpora as described above.

split	size
train	594585
dev	10000
test	10000

Table 4: Data size for paraphrase model

6 Methods

In this section, we introduce the architecture of the sequence-to-sequence models used for our implementation (Section 6.1). After that, we describe the training and working of the emotion classification component (Section 6.2). Two sections, 6.3 and 6.4, illustrate the training and fine-tuning of paraphrase and emotionalisation models respectively. We will demonstrate how to generate texts containing target emotions in Section 6.5.

6.1 Pre-trained Sequence-to-Sequence Models

The sequence-to-sequence model (Sutskever et al., 2014) has become the dominant approach in generation tasks. In recent years, the *Transformer* architecture (Vaswani et al., 2017) with self-attention block (Cheng et al., 2016) is shown to achieve state-of-the-art performance in a wide range of NLP tasks. Furthermore, attempts have been made to combine the Transformer architecture with self-supervised objectives trained on a large amount of text, (Devlin et al., 2019). In the training stage, these masked language models have to reconstruct corrupted input texts in which random tokens are masked out. Due to their success, different noising schemes have been studied and explored for various tasks. Examples include the span prediction SpanBERT (Joshi et al., 2020), and the generalized text-to-text framework T5 (Raffel et al., 2020).

In our project, we experiment with two Transformer encoder-decoder models called BART and PEGASUS. BART (Lewis et al., 2020) aims to be a general sequence-to-sequence model that is applicable to both text generation and comprehension tasks. On the other hand, PEGASUS (Zhang et al., 2020a) is originally developed for text summarization.

6.1.1 BART

BART (Lewis et al., 2020) is a denoising autoencoder for pretraining sequence-to-sequence models, using a standard Transformer-based encoder-decoder architecture. It can be seen as a combination of BERT (Devlin et al., 2019) due to its bidirectional encoder and GPT (Radford and Narasimhan, 2018) with its autoregressive decoder. Its pre-training step consists of two stages. The first one is the corruption of the input texts with a noising function. Similar to SpanBERT (Joshi et al., 2020), BART masks spans of text instead of single tokens. In SpanBERT, however, one masked symbol *[MASK]* still represents one token, meaning that the number of masked symbols in a span equals to its original number of tokens. In contrast, each masked symbol in BART representing an arbitrary length of span, including zero length. The second training stage learns to reconstruct the original text from its corrupted transformation, using a left-to-right decoder. Figure 5 illustrates the training procedure. For

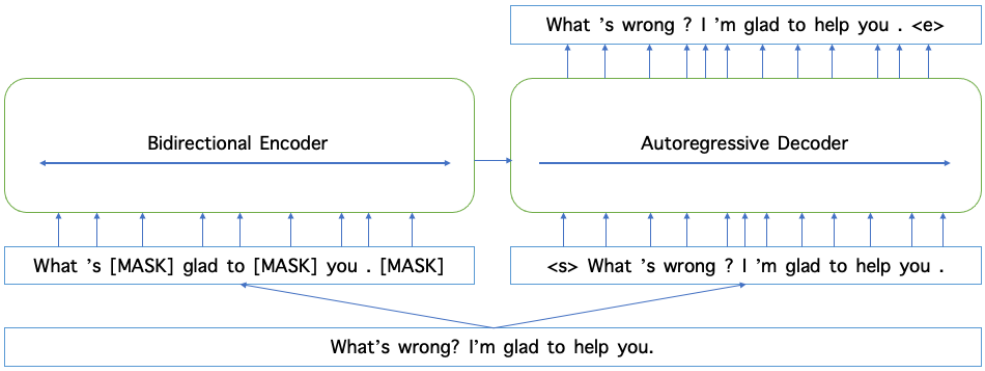


Figure 5: Training procedure of BART (Lewis et al., 2020). On the left, an arbitrary number (zero included) of spans of text is corrupted and replaced with *[MASK]* symbols before being fed into a bidirectional encoder. On the right, an autoregressive decoder tries to reconstruct the original uncorrupted text by calculating its likelihood from the encoded representation step-by-step from left to right.

fine-tuning, both the encoder and decoder are fed an uncorrupted text and representations from the final hidden state of the decoder is used for text generation.

BART is most suitable for generation tasks due to its decoding component. It also works well for comprehension tasks (e.g. classification, question answering) with a performance comparable to RoBERTa (Liu et al., 2019).

6.1.2 Pegasus

PEGASUS (Zhang et al., 2020a) is also a standard Transformer-based encoder-decoder model pre-trained with self-supervised objectives on a large amount of text. However, it is developed mainly for the task of abstractive summarization. For this purpose, Zhang et al. (2020a) experiment and propose a novel pre-training objective called Gap Sentences Generation (GSG). GSG resembles extractive summarization task in which the model learns to predict which sentences in the input text can be accepted as the summary of the text.

Similar to BART, PEGASUS is also inspired by the works of SpanBERT (Joshi et al., 2020) in which contiguous tokens are replaced with *[MASK]* symbols. PEGASUS, however, masks the entire sentences in the text, not just spans. Furthermore, the sentences are not masked randomly but chosen based on their importance to the text. Each sentence is scored independently and the top m sentences are selected. All the masked sentences are concatenated to create a pseudo-summary which the decoder is trained to generate from the encoded representations.

Figure 6 shows the training procedure of PEGASUS. In contrast to BART,

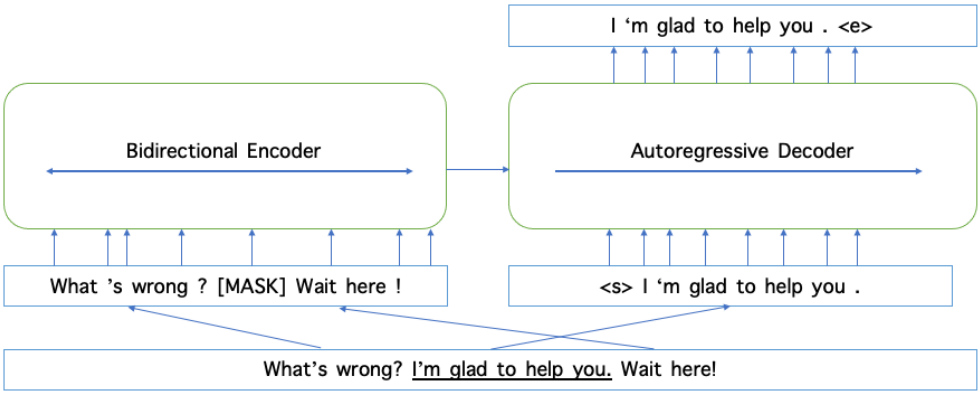


Figure 6: Training procedure of PEGASUS-large model (Zhang et al., 2020a). On the left, the input into the encoder is a corrupted text in which selected sentences are replaced with masked symbols. On the right, the decoder learns to generate the masked sentences from the encoded representations. The large model of PEGASUS excludes MLM training objective. The masked sentence is chosen based on its importance in the input text.

PEGASUS is trained to predict and generate the masked sentences, not the entire input text. For the base model, the encoder of PEGASUS is trained further with the Masked Language Model (MLM), similar to BERT. However, since MLM shows no gains on downstream tasks when training at a large number of steps, the large model uses only the GSG objective.

6.2 Emotion Classifier

Our emotion classification model is trained using the RoBERTa architecture. Developed by Liu et al. (2019), RoBERTa is an optimized model of BERT (Devlin et al., 2019) with these following modifications:

1. use dynamic masking strategy instead of static masking, meaning that the masking pattern is generated when the sequence is fed into the model, not during data processing step. Therefore, the same input is masked differently in each training epoch.
2. use *FULL-SENTENCES* training format and removing Next Sentence Prediction (NSP) loss. *FULL-SENTENCES* is a sampling method in which each input consists of contiguous full sentences from one or more documents.
3. train with a very large batch size: 8K sequences. BERT is trained with batch size of 256 sequences.
4. increase vocabulary size of Byte-Pair Encoding (BPE) word segmentation method (Sennrich et al., 2016) from 30K to 50K.

Our RoBERTa classifier is trained on the subset of the `Emotionalisation` dataset that is designed for the emotion classification task (See Section 5.3). Let N be the number of all samples in the dataset and M the number of labels. Additionally, y is the true label for a sample and p is the predicted probability of a sample for a label m. The cross-entropy loss function for multi-label classification is as follows:

$$L_{class} = -\sum_{n=1}^{N}\sum_{m=1}^{M} y_{mn} log(p_{mn}) \quad (1)$$

We use the emotion classifier to identify whether a generated output contains the target emotion. In the post-processing step after generation, the outputs without the target emotion are filtered out. To make sure that the generated texts *do* contain the desired emotion, we set the minimum score to 0.6 and discard all texts below this threshold. The remaining outputs are ranked using a scoring function of which the predicted emotion score is a component.

During the training of our emotionalisation model (See Section 6.4), the loss of the classifier is added to the total training loss to guide the decoding process towards generating the target emotion. For this purpose, our classifier accepts both the discrete tokens and their embedded representations.

6.3 Training for Content Preservation via Paraphrasing

Our `Emotionalisation` dataset is small: the number of each emotion style is fewer than 2k samples, and the total samples is roughly 4k (See Table 3 in Section 5.3). Therefore, we leverage transfer learning techniques to improve the model's ability to preserve the original meaning of the transferred sentences. We frame the content preservation problem as the task of paraphrase generation, and proceed to train the sequence-to-sequence models mentioned in Section 6.1 into paraphrase models first before fine-tuning them into style transfer models.

Paraphrase models aim to generate texts that convey the semantics of the input texts using different words and/or syntax (Zhou and Bhat, 2021). A bad paraphraser only copies or slightly modifies the input. Inspired by Krishna et al. (2020), to improve the quality of paraphrased outputs, we perform several stages of filtering sentences pairs from the `PARABANK-2` corpus which are too similar lexically and syntactically. Section 5.4 shows details about the creation of our paraphrase data.

Let $x = \{x_1, x_2, ..., x_n\}$ represent the input sentence of length *n* and $y = \{y_1, y_2, ..., y_m\}$ of length *m* the target paraphrase. The goal of the paraphraser is to generate y from x.

The paraphraser is optimized using cross-entropy loss L_{CE} of language mod-

eling as in Equation 2. The outputs are generated in an autoregressive manner using representations from the last hidden state of the decoder.

$$L_{lang} = \sum L_{CE}(x, y) \tag{2}$$

6.4 Final Fine-Tuning for Emotion Style Transfer

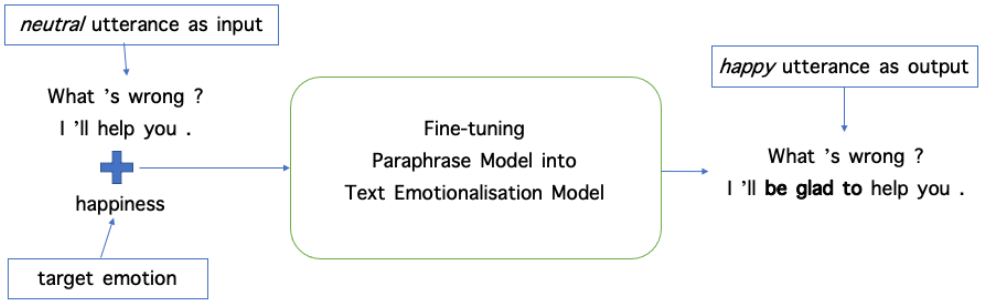


Figure 7: Fine-tuning procedure of our text emotionalisation model. The source input text is concatenated with the target emotion before being fed into the paraphrase model. The model is trained to generate the text with suitable emotion phrase that fit into the context.

We fine-tune the paraphrase models described in Section 6.3 to emotionalise neutral texts using our small parallel `Emotionalisation` dataset (See Section 5.3). Figure 7 illustrates the fine-tuning of our emotionalisation model. An overview of our training procedure from pre-trained models to text emotionalisation models is shown in Figure 2. We consider the transfer of the input text to each emotion as a separate task. The emotionalisation model handles the transformation to three emotions, meaning a multi-task learning approach for the training.

Let $x = \{x_1, x_2, ..., x_n\}$ represent the source input sentence and t_i be the target emotion with $t_i \in \{anger, happiness, sadness\}$. The concatenated input text $x + t_i$, separated by the separator token `[SEP]`, is fed into the paraphrase model with the goal to generate from left to right the sequence $y = \{y_1, y_2, ..., y_m\}$ containing t_i. Our hypothesis is that the model should learn the association between the emotion and its corresponding phrases. Therefore, being conditioned on the target emotion, it can infer the position in the source sentence to insert the correct emotion phrases and paraphrase the text if necessary to generate fluent outputs.

Similar to Prabhumoye et al. (2018) and Lai et al. (2020), our training makes use of style loss to shift the model to changing the input sentence into the target style. The generation loss (Equation 3) is a combination of the model's language modeling loss (Equation 2) and the emotion classifier loss (Equation 1). The calculation is made using the last hidden state from the decoder.

$$L_{gen} = L_{lang} + L_{class} \tag{3}$$

The loss from the emotion classifier is used to provide feedback and guide the generation process towards outputting the emotion phrases which are strongly associated with the target emotion. Since the tokenizer of RoBERTa is identical to that of BART[7] and the last hidden layer of BART decoder has the same size of the hidden layer of RoBERTa (i.e. 1024), we simply pass the last hidden state of BART decoder as embedded inputs to our trained RoBERTa classifier to get the emotion loss.

6.5 Generation with Specific Emotion

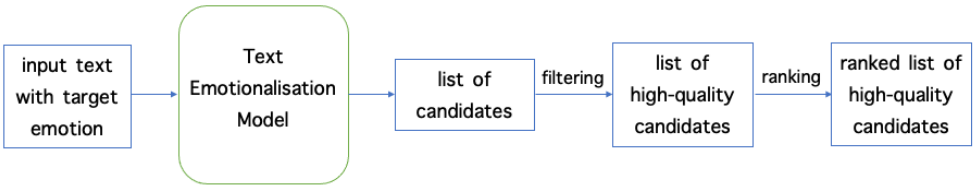


Figure 8: Generation with our emotionalisation model. Using a large beam size for candidate search, we obtain a list of candidate texts. After the filtering stage, low-quality predictions are discarded, including the texts containing the same emotion phrases. After that, the remaining texts are ranked based on an aggregation of three scores, namely emotion, similarity, and fluency scores.

In the fine-tuning stage on development set, the minimum number of beam size of the emotionalisation model is set to 8. Only the best hypothesis is returned. During inference, we set the beam size to 50 and get a list of 30 outputs. It is possible to adapt the generation process by changing the parameters of the function `generate()`[8] shared by all the `huggingface` generators. After that, different scoring methods are applied to obtain a ranked list of high-quality predictions.

- **emotion score:** for the measurement of the strength of the emotion "style" in the generated texts, we use the confidence score of our emotion classifier (See Section 6.2) after softmax function.
- **similarity score:** we adopt Sentence-BERT[9] (Reimers and Gurevych, 2019), a modified BERT model that uses Siamese networks for semantic similarity scoring between the input and generated texts. For each

[7] `https://huggingface.co/docs/transformers/model_doc/bart#transformers.BartTokenizer`

[8] `https://huggingface.co/docs/transformers/v4.16.2/en/main_classes/model#transformers.generation_utils.GenerationMixin.generate`

[9] `https://github.com/UKPLab/sentence-transformers`

pair of input and output, we measure their similarity by taking the cosine of their sentence embeddings.

- **fluency score:** following Krishna et al. (2020), we use their RoBERTa-large classifier trained on CoLA corpus (Warstadt et al., 2019) to score the naturalness of the generated outputs. Ranging from 0 to 1, the score reflects the grammatical acceptability judgement on a sentence.

Figure 8 illustrates our generation process during inference. In the filtering stage, we adopt three heuristics to maximize the output diversity. First, predictions that are shorter than the input text are discarded. Short texts do not suffer from this issue. However, for long input texts which consist of multiple sentences, the models occasionally fail to produce the entire original texts. Secondly, we aim to get a set of predictions with distinct emotion phrases. Predictions are considered duplicates if they have the same set of tokens, excluding punctuation. In other words, the following texts *"damn, he sold all out for that kick"*, *"he sold all out for that kick. Damn."*, and *"damn! He sold all out for that kick"* are duplicates, but not *"damn him he sold all out for that kick"*. Our final heuristics ensures that the sentences are scored higher than certain thresholds to be considered for the ranking stage. The minimum emotion, similarity, and fluency scores are set to 0.5, 0.4[10] and 0.5 respectively. If none of the predicted texts meet the scoring requirements after filtering stage, we back-off, removing the requirements one by one until the predicted set is not empty.

In the ranking stage, emotion, similarity, and fluency scores are combined as in Equation 4 for each prediction o and input text i. We set $\alpha = \gamma = 1$ and $\beta = 0$, obtaining the 10 highest-scoring predictions.

$$score_o = \alpha \times emo_o + \beta \times sim_{o,i} + \gamma \times flu_o \tag{4}$$

We decide not to include the similarity score due to our observation that more interesting predictions tend to occur when the similarity score is not too high. For example, given the input text *"there are some people out there in the world"* and the target emotion *"anger"*, the prediction *"there are some **jerks** out there in this **stupid** world"* scores only 0.6 on similarity while *"there are some **fucking** people out there in this world"* obtains 0.8. The former output involves some paraphrasing, removing one word and replacing it with another more *angry* word. The latter, however, simply inserts an emotion phrase into the input text for the transformation. Furthermore, we believe both predictions are equally satisfactory and thus choose not to let the similarity score

[10]We calculate the cosine similarity between all the neutral and emotional sentences in our dataset. The distribution shows that scores range from -1.2 to 1.0, with the majority being over 0.5. We, however, choose 0.4 as minimum semantic similarity score to relax the restrictions.

influence the ranking but instead reply on the prediction's emotion intensity and fluency.

7 Automatic Evaluation

This section explains the automatic metrics for evaluating our text emotionalisation models. We will first introduce three most popular metrics commonly used in Style Transfer literature in Section 7.1, and move to define the aggregation methods of these scores in Section 7.2. The last part of this section, Section 7.3, discusses our approach to measure the diversity of the emotion phrases generated by each system.

7.1 Popular Metrics on Style Transfer

For the Style Transfer task, popular evaluation methods aim at measuring (1) how successful the models are in transferring input sentences to the target style, (2) how well the original meaning is preserved in the transformed sentences, and (3) how fluent and natural the transformed sentences are. At sentence level, these requirements correspond to *emotion score*, *similarity score*, and *fluency score* respectively as described in Section 6.5. However, for an evaluation of the performance of the models on the test set, we switch over to the terms *transfer accuracy*, *content preservation*, and *fluency* as they are more commonly used in style transfer literature. During inference (Section 6.5), we obtain not only the best prediction but a ranked list of predictions. For evaluation purposes, therefore, only the highest-scoring output is used.

Transfer Accuracy (`EMO`) is used to measure the *style strength* of the transferred texts. Given a generated output o and its target style s, a pre-trained emotion classifier described in Section 6.2 is adopted to predict the emotion label of the generated output o. We report the accuracy of the predictions in the test set.

A minimum emotion score of 0.5 is used for filtering purposes (Section 6.5), which might result in an empty list if none of the generated texts satisfy our requirements. In this case, we simply treat the predicted emotion label for the transferred text as incorrect.

Content Preservation (`SIM`) measures the degree of *original content* preserved in the transferred texts. As pointed out by Xu et al. (2018), style transfer models can have *"hallucinations"*, generating texts with the target style but changing its semantics. Common evaluation methods adopt BLEU metrics (Papineni et al., 2002) and its variants (e.g. self-BLEU). In essence, BLEU variants calculate the percentage of matching tokens of two or more sentences, either using single tokens (i.e. unigram), or multiple tokens (i.e.

bigram, trigram).

However, strict *n-gram* overlap suffers from several shortcomings: (1) no correlation between BLUE scores and human evaluation (Callison-Burch et al., 2006), (2) it ignores the fact that there is more than one strategy to transfer a sentence, and (3) it does not account for words having similar semantic meaning.

In recent years, methods to calculate semantic similarity based on distance between embeddings have been proposed such as sub-word embeddings (Wieting et al., 2019) and token embeddings, e.g. BERTScore (Zhang et al., 2020b). Another possibility is to calculate the cosine score between the true reference and the transferred output as their semantic similarity, similar to what has been done in Section 6.5. However, we have decided to adopt BERTScore (Zhang et al., 2020b) as it has been shown to have the strongest correlation with human evaluation (Bujnowski et al., 2020). The semantic similarity of two sentences is computed as the sum of cosine similarities between their tokens' embeddings based on BERT contextual embeddings (Devlin et al., 2019). A similarity score is computed for each token in the reference sentence and each token in the candidate sentence. The final score is the sum of the highest scores between the token pairs.

Fluency (`FLU`) measures the *naturalness* of the transferred texts. The most common method is to calculate the perplexity of the text using a language model. Following Helbig et al. (2020), we adopt GPT (Radford and Narasimhan, 2018), an autoregressive Transformer-based language model, for perplexity calculation and then normalize the score to the range $[0, 1]$. However, after a manual inspection, we observe that the normalized perplexity scores do not fully reflect the fluency of the texts: a text of 0.7 (e.g. *"I am so mad that he sold all out for that kick"*) can be as natural as one of 1.0 (e.g. *"I can't believe he sold all his shit for that kick"*). Raw unnormalized perplexity scores, on the other hand, are not completely straightforward to be aggregated into one number for all predictions on the test set.

Therefore, we adopt the approach proposed by Krishna et al. (2020), which adopts an classifier trained on the CoLA corpus (Warstadt et al., 2019) for grammar calculation. The CoLA corpus is a collection of roughly 10k English sentences from linguistics literature, each labelled as grammatical or ungrammatical. The classifier aims to judge the grammatical acceptability of sentences. We acknowledge that *fluency* and *grammatical correctness* do not necessarily refer to the same concept but we decide to adopt the term *fluency* due to its popularity in the literature.

7.2 Aggregation of Metrics

It is useful to aggregate all three aforementioned scores into a single number for model selection and comparison purposes across different models. Commonly, it is done in an macro scheme, meaning that the number is combined using averaged scores of the metrics on the test set. However, this aggregation scheme is argued to be problematic since the final score is unable to reflect the metrics the systems fail to perform well (Pang and Gimpel, 2019; Krishna et al., 2020). For example, assuming that there exists a naive model which only copies random words from the input text to the output. It can score almost perfectly on similarity and transfer accuracy metrics, but not fluency. If the final score is obtained by macro averaging, the model seemingly performs quite reasonably while it is not.

Therefore, we follow the micro aggregation scheme proposed by Krishna et al. (2020) in which a combined score of three metrics is first calculated for each sentence. The final score is an average of the scores of all the sentences in the test set. Let x be a sentence from the test set X, the final score is aggregated as:

$$micro = \sum_{x \in X} \frac{EMO(x).SIM(x).FLU(x)}{|X|} \tag{5}$$

Both *transfer accuracy* and *fluency* are treated as binary judgement. The intuition is that low-quality outputs (i.e. containing no target emotion or being ungrammatical) are automatically assigned a score of 0. Under a micro evaluation scheme, these zero-scoring outputs will lower the final score.

In addition, for comparison purposes, we compute the macro aggregation scheme as follow.

$$macro = \frac{EMO(X) + SIM(X) + FLU(X)}{3} \tag{6}$$

7.3 Diversity Score

It is possible that the emotionalisation models overuse a small set of emotion phrases for its generation strategy. In fact, as mentioned in Section 8.1, it is possible to obtain high scores on transfer accuracy and content preservation metrics simply by implementing some heuristics which randomly pick some emotional exclamation and insert it either at the start or the end of the input sentence. Therefore, we decide to perform another method to measure the diversity of the emotion phrases generated in the outputs.

We examine the survey on diversity evaluation in NLG by Tevet and Berant (2021), which discusses methods for measuring diverse *form* and diverse *con-*

tent. The two terms *form* and *content* refers to the diversity of *tokens* and *meaning* respectively. For example, two phrases *"I am glad"* and *"I was happy"* should score high on diverse *form* but low on diverse *content*. Therefore, we adopt *distinct n-grams* metrics (Li et al., 2016) for the measurement of diverse form. We briefly consider using also content diversity evaluation for our project. However, as one of the requirements for style transfer task is content preservation, having diverse content is simply opposite of what we aim to achieve.

Different from three methods in Section 7.1, our diversity method consider not just the highest scoring output but five highest scoring outputs for each input text. We adopt the same approach to obtain the emotion phrases from the emotional sentences as shown in Table **??** in Section 5.2.1. Using Levenshtein distance method on each output, we get a list of word sequences that are necessary for the transformation of the input text to the predicted output.

Given a list of emotion phrases from all the generated outputs, *distinct n-grams* metrics (Li et al., 2016) computes the ratio of unique n-grams to all n-grams of all the phrases in the list. Following common choices, we report diversity scores of unigrams and bigrams of tokens. The higher the scores are, the more diverse the phrases generated by a model will be.

8 Experimental Settings

We discuss the baseline of our experiments in Section 8.1. Our main experiments focus on the text emotionalisation models using BART and PEGASUS architectures and the generation strategy. However, Section 8.2 also provides details of our hyper-parameter choices for all models in Section 6, in addition to the performance of the emotion classifier and the paraphrase models.

8.1 Naive Method as Baseline

As briefly mentioned in Section 7.2 and Section 7.3, high scores can be obtained by using a naive model which picks a random phrase from a list of emotion phrases and inserts it either in the beginning or the end of the sentences. This model is expected to achieve almost perfect scores on *transfer accuracy* due to the guaranteed presence of an emotion phrase. Its performance on *semantic similarity* scores is, possibly, quite good due to the high overlap between the input and the generated output. It, however, should perform poorly on *fluency* scores since the emotion phrase is chosen at random at inference time from our lists of emotion phrases. Therefore, we hypothesise that the resulted outputs should be ungrammatical and unnatural.

8.2 Experimental Settings

Our experiments use the Pytorch framework (Paszke et al., 2019) and the `transformers` library[11] (Wolf et al., 2019).

Emotion Classifier: We fine-tune the `RoBERTa-large` model as described in Section 6.2. The learning rate is set to 2×10^{-5}. Our mini batch size is 8 with gradient accumulating for 2 steps, thus the model parameters are updated for every 16 training samples. We use early stopping and train the model until it converges. The evaluation on the validation set is performed after every 1000 training steps. The training is finished if no improvement on the accuracy metric is found for 5 consecutive evaluation times. For optimization, we use AdamW (Loshchilov and Hutter, 2019). Our emotion classifier achieves an accuracy score of 0.9 on both validation and test sets.

Fluency Classifier: We do not train our own classifier but instead use directly the RoBERTa-large classifier by Krishna et al. (2020), which is trained on CoLA corpus (Warstadt et al., 2019). The authors, however, did not release any information about the performance of their model in their paper.

Paraphrase Models: A BART `bart-large`[12] model is trained on its default denoising objectives specified in `transformers` library. We set the learning rate to 1×10^{-5} and train the paraphraser for 10 epochs with early stopping after no improvement found on validation set for 2 epochs. The training is optimized on AdamW (Loshchilov and Hutter, 2019) with a weight decay of 0.1, together with a linear learning rate scheduler. Our batch size is 6 and the maximum sequence length is set to 60. If the input text is shorter than 60, the padding is done on the right to maximum length. For the evaluation, we adopt BERTScore (Zhang et al., 2020b) to calculate the semantic similarity between the candidate and the reference. On both validation and test sets, our BART paraphraser achieves scores of 0.92 for all precision, recall, and f1 metrics.

As for PEGASUS, we use directly the pre-trained model from the `huggingface` database[13] which is fine-tuned for the paraphrase task using the `pegasus-large` model.

Emotionalisation Models: For our main experiments, we fine-tune two paraphrase models, BART and PEGASUS, with the same hyperparameters except for learning rates, which are 1×10^{-5} and 1×10^{-4} respectively. We did experimentally train BART with 1×10^{-4}, which, however, resulted in a very poor performance. Again, the optimizer is AdamW (Loshchilov and Hutter, 2019) with no weight decay and we use a linear learning rate scheduler. The batch size is 6 with 2 steps of gradient accumulation, resulting in an update

[11] `https://huggingface.co/docs/transformers/`

[12] `https://huggingface.co/facebook/bart-large`

[13] `https://huggingface.co/tuner007/pegasus_paraphrase`

of every 12 samples. Similar to the BART paraphraser, the maximum length of the input to the emotion style transfer models is 60 and padding is done on the right to maximum length. The model is trained until convergence on the validation set, after no improvement is found after 3 epochs. During training, the evaluation is performed using the transfer accuracy metric only. We decide not to include the semantic similarity metric BERTScore (Zhang et al., 2020b) in the evaluation loop after observing a long training time in the first trials. Therefore, BERTScore calculation is deactivated in our main experiments to speed up the training process. One reminder is that this evaluation during training is for fine-tuning purposes. It is, thus, different from our main evaluation which is performed after the training finishes.

Generation Strategy: During inference of text emotionalisation models on the test set, we use 50 beam searches and obtain a list of 30 best outputs from the models. After that, several filtering methods are adopted. The remaining outputs are ranked based on the sum of their emotion, and fluency scores as described in Section 6.5.

9 Experimental Results

In this section, we report the results of our main experiments (Section 9.1 and 9.2) and ablation studies on different architecture design (Section 9.3) using text emotionalisation models during inference.

model	EMO	SIM	FLU	micro	macro	diversity	
						d-1	d-2
inputs	0.106	0.961	0.802	0.090	0.623	0.000	0.000
references	0.876	**0.999**	0.838	0.736	0.904	**0.209**	**0.626**
naive	**0.997**	0.913	0.963	0.878	0.958	0.045	0.141
bart	0.987	0.951	0.953	0.897	0.964	0.055	0.195
pegasus	0.993	0.945	**0.971**	**0.914**	**0.970**	0.058	0.214

Table 5: Results obtained on the test set uisng automatic evaluation. EMO, SIM, and FLU refers to *transfer accuracy*, *content preservation*, and *grammar acceptability* respectively. *Micro* is the aggregation of three metrics described in Section 7.2. Additionally, we calculate the geometric mean of three scores and report it under *macro* heading. Diversity is reported on distinct unigrams and bigrams (See Section 7.3).

9.1 Results on Automatic Metrics

The results obtained on the test split of our `Emotionalisation` dataset are shown in Table 5. In addition to the naive method and two sequence-to-sequence models, the evaluation is performed on the neutral input texts and the references. We reported accuracy for `EMO`, which is obtained using the emotion classifier as explained in Section 6.2. As for `FLU`, the score should be understood as the percentage of grammatical outputs generated by the models. `SIM` uses BERTScore for the calculation of token embedding matching, resulting in precision, recall, and f1 scores. We adopt precision for both micro and macro aggregation calculations, thus `SIM` score reported in Table 5 is the precision of BERTScore.

The evaluation carried out on the neutral input texts shows a huge gap between micro and macro aggregation methods. Even with an `EMO` score of 0.106, the macro score still stands at 0.623. However, adopting micro calculation causes a sharp drop to just 0.09. This proves that on micro evaluation scheme, a harsher punishment is given for low-quality candidates. Surprisingly, references do not always deliver the best performance. Both of their `EMO` and `FLU` scores are lower than the other three models. Inspecting the emotion scores on the references, we discover that only 634 out of 880 references achieve perfect emotion score of 1.0 while 109 have lower scores than 0.5 and 24 scores 0.0.

The references having zero emotion scores are all due to containing phrases of another emotion for three reasons. First, the speaker is being sarcastic: In the reference *"oh wow, talk about bad timing"*, though being *angry*, the speaker expresses it via the phrase *"oh wow"*, normally used to show if one is surprised or impressed. Second, the reference contains mixed emotions: *"I feel both thrilled and shy. It's both unsettling and exciting to see myself in this way."* is annotated with *happiness* emotion. However, due to the words *"shy"* and *"unsettling"*, the classifier predicts it to be of *sadness*. The last reason is because of a strong association between the phrases and emotions in our corpus. For example, phrases *"it sucks"* and *"pissed off"* are defined as *anger* style and therefore, would get a very low, if not zero score, on *sadness*. We consider this problem to be the same as using emotion lexicons for emotion recognition, which is high accuracy but low recall. One solution could be to expand our collection of emotion phrases to be more inclusive.

On the micro aggregation metric, PEGASUS outperforms the other two models, achieving the highest `FLU` score and second highest scores for two remaining metrics. On diversity metrics, PEGASUS consistently scores higher, which we attribute to its ability to perform some paraphrasing instead of simple insertion. However, compared to the references, its diversity scores are still threefold lower. We speculate this issue could be addressed by manipulating different parameters of the generators for a more diverse outputs. Additionally, we can design some heuristics to get a outputs with varying emotion

scores. As mentioned earlier, some phrases are strongly associated with an emotion, resulting in a higher ranking of the predictions containing them. For example, we observe a tendency of over-generating phrases with the word *"sorry"* in both BART and PEGASUS when it comes to *sadness* emotion. That means the diversity scores can be improved by allowing predictions with lower total scores.

input	oh wow, talk about timing.
target emotion	anger
reference	oh wow, talk about bad timing.
naive	**that is utterly incorrect!!** oh wow, talk about timing oh wow, talk about timing! **drive me crazy!** oh wow, talk about timing, **you racist nazi.**
bart	oh wow, talk about **fucking** timing. oh wow, talk about **stupid** timing. oh wow, talk about **fucked up** timing.
pegasus	oh wow, talk about **fucking** timing. oh wow, **how dare you** talk about timing. oh wow, talk about timing **bullshit**.

Table 6: Examples of output texts of *anger* emotion.

input	the new football season starts soon.
target emotion	happiness
reference	I'm looking forward to the new football season starting!
naive	**I agree with you!!** the new football season starts soon. **this sounds intriguing!!** the new football season starts soon. **and i am so excited!** the new football season starts soon.
bart	**i can't wait for** the new football season. **i am so excited for** the new football season. **congrats!** the new football season starts soon.
pegasus	**i am excited about** the new football season starting soon. **i am excited for** the new football season to start soon! **i am looking forward to** the new football season starting soon!

Table 7: Examples of output texts of *happiness* emotion.

Tables 6, 7, and 8 show examples of outputs generated by the naive method, BART, and PEGASUS models on three emotions, namely anger, happiness, and sadness respectively. For each model, we select the three highest scoring predictions. The reference in Table 6 has a zero emotion score on *anger*.

input	yeah, i don't mean it.
target emotion	sadness
reference	yeah sorry about that i don't mean it.
naive	yeah, i don't mean it! **i feel left out...** yeah, i don't mean it. **that 's too bad.** yeah, i don't mean it. **I am very sorry.**
bart	yeah, **i'm sorry but** i don't mean it. yeah, **sorry,** i don't mean it. **yeahsorry.** yeah, i don't mean it.
pegasus	yeah, **i apologize, but** i don't mean **to offend**. yeah, **sorry, but** i don't mean it. yeah, **sorry,** i don't mean **to offend**.

Table 8: Examples of output texts of *sadness* emotion.

For a successful transformation, both BART and PEGASUS replace the word *"bad"* with words with stronger associations with *anger* such as *"fucking"*, *"stupid"*, and *"bullshit"*. Though their main strategy is pure insertion of emotion phrases, in Table 7, they appear to have the ability to paraphrase the output texts when necessary for fluency and grammar. Occasionally, BART appears to generate duplicate words and ungrammatical texts. In Table 8, PEGASUS is shown to add also the word *offense*.

9.2 Interpretation of the Naive Method

Our naive method outperforms both BART and PEGASUS on the transfer accuracy (EMO), which we believe is due to the fact that the inserted emotion phrase, though randomly chosen, is a strong signal for the emotion. Thus, its presence results in a high transfer accuracy. However, its fluency score is unexpectedly high, even higher than that of the BART model. When randomly inspecting the generated outputs by the naive model, we discover that though the inserted phrases might not fit very well into the context, the resulted texts are perfectly grammatical. Some of them include *"I have a roof over my head ! **I really enjoyed it!**"*, *"Why were they closed? **You are such a liar!!**"*, and *"The woman types like she isn't a native speaker of English! **My bad!**"*.

Furthermore, in the generation stage, the number of predictions is set to 30 and we obtain the best 10, ranked on emotion and grammar scores. It is perfectly possible that among all the randomly selected phrases, there exists at least one that results in a perfectly grammatical output. For the calculation of fluency score FLU, only the highest scoring prediction is taken into consideration, which proves to score high in grammar metric. However, considering that not

all the emotion phrases are complete sentences as can be seen from Table 2, it is surprising that the overall FLU score is quite high. Moreover, since the phrases are selected randomly, we have our doubts over the coherence of the outputs, i.e. whether the combination of the emotion phrase and the input text makes sense semantically. A method for evaluating coherence, unfortunately, is out of scope of our project. To simplify the problem, we decide to test our hypothesis using perplexity metrics. We adopt the implementation of the language model GPT-2 (Radford et al., 2019) by huggingface[14] for the calculation of perplexity scores of the five highest-scoring predictions for each input text of all three models. Table 9 shows the mean and median of perplexity scores of our models. Of the three, the naive method has lowest-quality predictions by a large margin. Scores of BART and PEGASUS are, on the other hand, quite close.

model	mean	median
naive	258.9	155.7
bart	199.7	**83.6**
pegasus	**182.9**	86.3

Table 9: Mean and median of perplexity scores by GPT-2. The lower the scores, the more natural the predictions.

Additionally, Figure 9 shows the perplexity distribution. The maximum score is set to be 1000, resulting in the removal of 113, 62, and 71 outliers on the naive method, BART, and PEGASUS respectively. In the plot, the naive line is observably more skewed to the right compared to the other two models.

9.3 Impact of Architecture Design

Table 10 shows the results of BART and PEGASUS when we experiment further with using no emotion loss (nel) during training, and no paraphrasing stage (nps) before the fine-tuning into the emotionalisation models. Our standard experiment (std) uses both emotion loss and paraphrasing stage. Both on the micro and macro evaluations, the best performance is obtained on PEGASUS trained with emotion loss. Though the scores vary slightly during inference, we observe a greater difference during training. During inference, the models generate a list of 20 predictions, on which we apply filtering and ranking methods to obtain 10 highest scoring candidates (See Section 6.5). During training, to accelerate training process, in addition to adopting only style accuracy metrics, evaluation performed on both development and test sets use only one best candidate from the models.

[14] https://huggingface.co/docs/transformers/v4.16.2/en/model_doc/gpt2

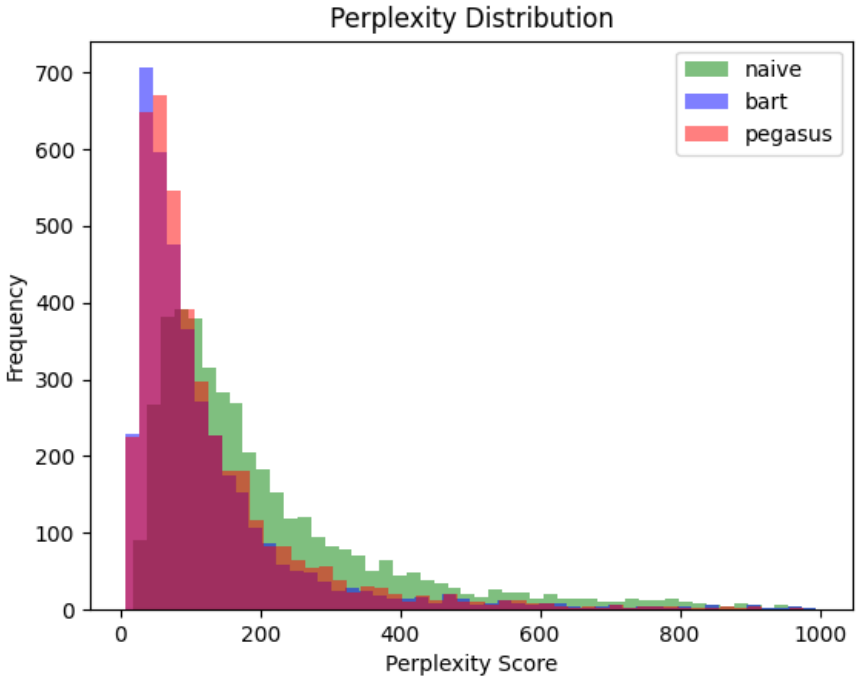


Figure 9: Perplexity distribution of three models.

The generation processes performed during training and inference are different. The former produces only one output while the latter generates 30 candidates. Therefore, we investigate whether this difference occurs during the training. Table 11 shows the emotion style accuracy scores during training. Standard BART trained with emotion style loss and paraphrase stage produces best evaluation results. BART trained without paraphrase stage takes the longest to converge but appears to obtain best performance on both the micro and macro aggregation of three scores. On PEGASUS, the results are mixed: best style accuracy scores are obtained on the development set when training without style loss, and on the test set when training with style loss. Overall, PEGASUS trained with style loss still delivers the best performance. Nevertheless, any difference among model configurations is settled during inference due to the filtering and ranking stages.

When manually inspecting the outputs, we notice that different systems can sometime generate the same texts. To understand more this phenomenon, for each input text, we first select 10 highest scoring outputs which meet our minimum scores of emotion and grammar (i.e. higher than 0.5). With these lists, we calculate the overlap between different configurations. It can be observed from Figure 10 that though fewer texts are generated with BART compared to PEGASUS, three BART systems have approximately 20% of identical outputs while PEGASUS has just around 14%.

model	EMO	SIM	FLU	micro	macro	diversity	
						d-1	d-2
bart std	**0.987**	**0.951**	0.953	0.897	0.964	0.055	0.195
bart nel	0.986	0.949	0.964	0.905	0.966	0.057	0.198
bart nps	0.986	0.950	**0.969**	**0.910**	**0.969**	**0.069**	**0.243**
pegasus std	0.993	**0.945**	**0.971**	**0.914**	**0.970**	0.058	0.214
pegasus nel	**0.999**	0.944	0.962	0.910	0.969	**0.062**	**0.253**
pegasus nps	0.993	**0.945**	0.968	0.912	0.969	0.056	0.196

Table 10: Results on ablation studies. The acronyms *"std"*, *"nel"*, and *"np"* are shortened for *"standard"*, *"no emotion loss"*, and *"no paraphrase stage"* respectively.

model	dev.	test	epoch
bart std	0.802	0.786	5
bart nel	0.743	0.642	5
bart nps	0.703	0.688	15
pegasus std	0.808	0.803	9
pegasus nel	0.826	0.797	7
pegasus nps	0.806	0.801	9

Table 11: Evaluations performed on development and test sets during training, in addition to the epoch when the models converge. The acronyms *"std"*, *"nel"*, and *"np"* are shortened for *"standard"*, *"no emotion loss"*, and *"no paraphrase stage"* respectively. Reported scores are the transfer accuracy EMO.

The emotion style loss and the content preservation are meant for improving transfer accuracy and content preservation respectively. However, our pilot ablation studies show no noticeable difference among all three configurations on both models. We hypothesise the filtering and ranking stages after generation process contribute to narrowing the gap between style transfer accuracy scores during training and inference. Nevertheless, as briefly mentioned in the previous section, the current metrics might not accurately measure what we want it to measure. Therefore, new metrics on emotion style transfer task need to be developed before proper ablation studies on architecture design can be done. We present further discussion in the next Section.

(a) BART configurations (b) PEGASUS configurations

Figure 10: The number of outputs generated by different configurations of BART and PEGASUS, and their overlaps.

9.4 Interpretable Scores for Output Selection

The high scores obtained using the automatic metrics on the naive method pose the difficult question of how much we should trust these scores. To be more specific, this leads to us the problem of validity, i.e. their accuracy in measuring what we want them to measure. When manually inspecting the list of ranked predictions, interestingly, we discover that a lower score on any of the three metrics do not necessarily imply lower quality because our inspection reveals that even the references can achieve poor scores occasionally.

On emotion metrics: The candidate *"I feel bad for these guys, I'd like to be able to help ex-mos in trouble."* achieves an emotion score of 0.531 on sadness and is ranked eighth in the list. It, however, turns out to be the same as the reference. The other higher ranked candidates include phrases such as *"sorry for"* and *"helpless"*, which are extremely strong signal for *sadness*. Thus, they achieve a higher score on emotion metrics. This raises the question about whether it should be interpreted for emotion intensity purposes rather than to indicate the existence of an emotion.

On fluency metrics: This judges the grammatical acceptability of sentences. However, both spoken language and language used on social media are less formal than written language. We do observe that even the references (e.g. *"awesome! is your family going with you?"*) can get a score of 0.6 only. Scores of both grammatical acceptability and perplexity tend to be lower on predictions from the `GoEmotions` dataset. It consists of Reddit comments, which explains their low scores on models trained on more formal texts[15]. In other words, the `FLU` scores simply measure formality. Furthermore, just as Noam Chomsky has famously stated *"colorless green ideas sleep furiously"*, a grammatically well-formed sentence can sound semantically nonsensical.

On similarity metrics: Similarly, when the models change the word *"peo-*

[15]The CoLA corpus (Warstadt et al., 2019), used for the training of grammatical acceptability model, consists of sentences from linguistic publications.

		emo.	sim.	flu.
input	I see you liked it.			
target emo.	happiness			
reference	I'm glad you really enjoyed it!			
output 1	I see you liked it **very much**.	1.0	0.870	0.964
output 2	**wow,** I see you liked it.	1.0	0.821	0.966
output 3	**I am glad** you liked it.	1.0	0.634	0.977
output 4	**I'm happy to** see you liked it.	1.0	0.619	0.979

Table 12: Examples of outputs ranked by emotion, similarity, and fluency scores. Output 3 and 4 have lower similarity scores and are closer to the reference.

ple" to *"jerks"* or add the adjective *"stupid"* to produce *angry* utterances, semantic similarity scores between the input text and each these utterances drop to around 0.5 and 0.6. In Table 12, the outputs 1 and 2 are ranked highest on similarity metrics thanks to having most word overlap with the input. On the other hand, the output 3, though most similar to the reference, is ranked the third. During inference, since it is not possible to get access to the reference, the input is the only value we can rely on for the measurement of content preservation. However, our analysis reveals that the commonly used similarity metrics tend to favour the candidates with most overlap with the input, irregardless of what *content* is preserved. From Table 12, one can also argue that the output 1, though having the best similarity score, is, in fact, *changes* the original meaning of the input when it adds the emotion phrase. Originally, the speaker only says that *"you liked it"*, not *"liked it very much"*. In emotion transfer task, existing works show that it might not be possible to transfer the emotions without altering the semantic content (Helbig et al., 2020; Troiano et al., 2021). Our work raises another question, namely which should be the lower bound and the upper bound of the degree of original content preserved in the transferred utterances.

Trade-off between scores in the ranking stage: Our ranking stage adopts Equation 4, which raises the question about the selection of the weights for each score. Previous works have shown that the emotion, semantic similarity, and fluency scores have an inverse correlation with each other (Mir et al., 2019; Pang and Gimpel, 2019; Pang, 2019a; Helbig et al., 2020). To be more precise, the higher the style scores are, the lower the content similarity scores are. An increase in similarity leads to a reduction in fluency. Lastly, when fluency increases, style accuracy decreases.

Pang and Gimpel (2019) ask humans to annotate their preferences on a subset of transferred sentences and propose a model to learn the weights from this dataset. On our text emotionalisation task, one possibility is to consider the

scores not as indicators of quality but as the degrees of presence of an aspect. For example, one can treat emotion scores as the intensity of emotions in the sentences. As for content preservation, all sentences are accepted as long as the cosine similarity is within reasonable range. Moreover, additional methods can be adopted to further verify whether the content that needed to be preserved is indeed preserved. We leave the exploration of weight choices for the ranking and selection of predictions to future work.

10 Human Evaluation

Besides automatic evaluation methods, we perform human evaluation for verification purposes on three key metrics for the Style Transfer task mentioned in Section 7.1. Our human evaluation includes one additional question about whether a generated utterance fits into the context of the dialogue. Since our current generators are not designed to consider context, the added question is not meant to verify but rather to find out the current fitness of the system outputs to the dialogue.

10.1 Evaluation Settings

Evaluation platform: The evaluation is carried out on Amazon Mechanical Turk (AMT) platform. Each worker is paid $\$0.25$ per task, and we hire 3 workers per task.

Rating method: *Direct rating*, which is also called *single-item scales* in different literature, refers to settings in which the ratings are assigned in isolation on one single system output. In contrast, *relative rating*, also called *multiple-item scales*, refers to settings in which the ratings are assigned in comparison between two or more system outputs. In Style Transfer task, the most popular evaluation framework is direct ratings on a 5-point Likert scale (Briakou et al., 2021). However, since relative rating has been shown to be more reliable (Mir et al., 2019; van der Lee et al., 2021), we adopt the relative rating method on a 5-point Likert scale, yielding an absolute score from 1 to 5 for each evaluated text.

Task content: Karpinska et al. (2021) recommend references to be evaluated together with generated outputs when using AMT workers to improve rating calibration. Our generated outputs come from 4 systems, namely the naive method, PEGASUS trained with emotion style loss, PEGASUS trained without emotion style loss, and PEGASUS trained without paraphrase stage. This results in 5 system outputs for each input text to be evaluated, namely, reference (ref), pegasus standard (std), pegasus with no emotion loss (nel), pegasus with no paraphrasing step (nps), and naive (nai). For simplification, we exclude outputs from BART systems in our human evaluation framework.

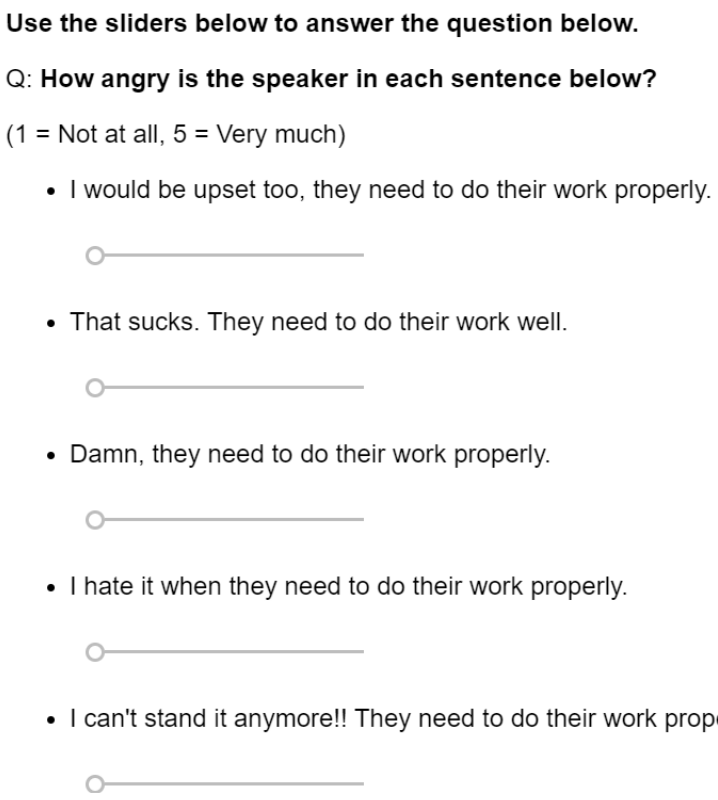


Figure 11: AMT interface for evaluating emotion transfer strength of five system outputs for one input text.

Except for the reference, for each remaining system, we pick the output having the highest sum of emotion, similarity, and fluency scores obtained in the ranking stage (See Section 6.5). To be chosen for the human evaluation, the texts have to meet four criteria. First, the number of tokens is limited from 5 to 20 to remove too long or too short utterances. Second, the five system outputs cannot be duplicates of each other. In other words, they have to use different emotion phrases. For example, *"I'm glad"* and *"I am glad"* are considered duplicates. Third, the input texts are from a dialogue, i.e. having previous utterances. Lastly, the input texts are completely neutral, and semantically meaningful. Our evaluation is performed on 21 input texts. There are 7 texts per emotion. These are randomly selected from the test set from a list of input texts that satisfy our four aforementioned requirements. Each input text is evaluated with 4 questions for 4 aspects, namely transfer strength, content preservation, fluency, and context coherence. Appendix B lists all the selected 21 input texts and system outputs used for human evaluation.

Transfer strength (EMO): How angry/happy/sad is the speaker in each sentence?

Content preservation (SIM): **show input text** How similar is the meaning of each sentence to the input text?

Fluency (FLU): How likely is each sentence written by humans?

Context coherence (COH): **show previous utterance** How well does each sentence fit with the previous utterance like in a dialogue?

We provide workers with the input text and its previous utterance (i.e. context). For each input text, they are asked to rate 5 system outputs on a 5-point Likert scale. Point 1 indicates "not at all" and point 5 "very much". Figure 11 shows one sample task for emotion transfer strength evaluation to be completed by AMT workers. See Appendix A to see the AMT interface of all four evaluated aspects.

10.2 Results

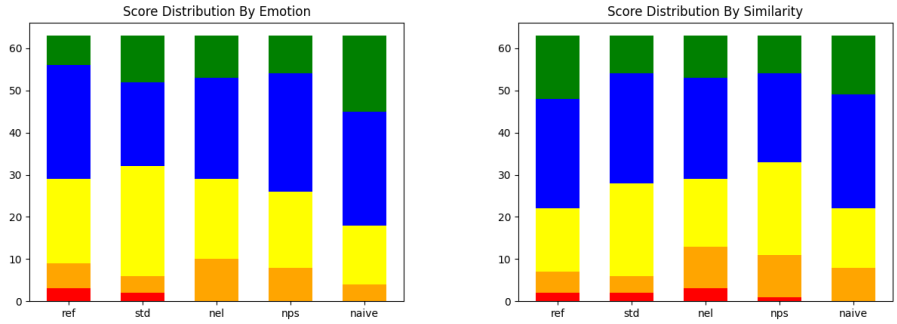
	EMO	SIM	FLU	COH
ref.	3.46 ± 0.98	$\mathbf{3.75} \pm 1.01$	$\mathbf{3.67} \pm 1.02$	$\mathbf{3.92} \pm 0.88$
std	3.54 ± 0.96	3.57 ± 0.93	3.52 ± 0.95	3.59 ± 1.01
nel	3.54 ± 0.95	3.44 ± 1.09	$\mathbf{3.67} \pm 0.86$	3.49 ± 0.91
nps	3.60 ± 0.89	3.43 ± 0.98	3.44 ± 1.09	3.35 ± 0.92
nai.	$\mathbf{3.94} \pm 0.88$	$\mathbf{3.75} \pm 0.95$	3.24 ± 1.06	3.64 ± 0.94

Table 13: Mean and standard deviation of scores given by AMT workers for each evaluation aspect of the outputs of five systems. From top to bottom, five systems are "references", "pegasus standard", "pegasus trained with no emotion loss", "pegasus trained with no paraphrasing step", and "naive" respectively.

EMO	SIM	FLU
0.116	0.186	0.039

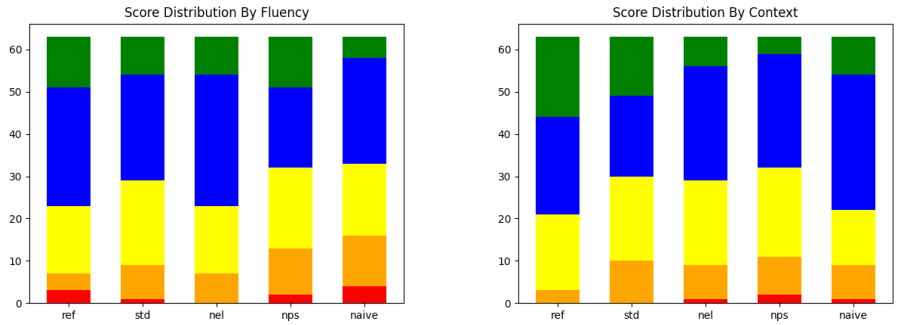
Table 14: Spearman correlation between automatic evaluation and human judgement on three aspects, namely transfer strength, content preservation, and fluency.

With 3 workers per aspect, and 4 aspects per input text, there are in total 252 human judgements made for 21 input texts. Table 13 shows the means and standard deviations for these judgements. Spearman correlation between results obtained on automatic evaluation and human judgement can be seen in Table 14. The scores ranges from -1 (perfect negative correlation) to 1 (perfect positive correlation). Our scores imply there is a very weak correlation between automatic and human evaluations. Figure 12 illustrates the stacked bar chart of scores given by AMT workers to all five systems for four evaluation aspects. Each color represents a score with the red one on the bottom being score 1 and the green one on the top being score 5.



(a) transfer strength

(b) content preservation



(c) fluency

(d) context coherence

Figure 12: Stacked scores for each evaluation aspect given by AMT workers. From the bottom to the top, the colors "red", "orange", "yellow", "blue", and "green" indicate the scores "1", "2", "3", "4", and "5" respectively. From the left to the right, the columns represent the five systems "references", "pegasus standard", "pegasus trained with no emotion loss", "pegasus trained with no paraphrasing step", and "naive" respectively.

On Table 13, on transfer strength aspect (EMO), the naive system and the reference achieves the highest (3.94) and the lowest (3.46) scores respectively. This outcome is similar to the results obtained using automatic evaluation in Table 5. The naive system has the best score also on the content preservation aspect (SIM) but the worst score on the fluency aspect (FLU). In Section 9.2, though the naive system has the second highest FLU score on grammatical correctness, we hypothesise that its scores should be lower on sentence coherence metrics. The results obtained on human evaluation seemingly imply that our intuition is correct. On the Figure 12c, the naive system gets the highest number of the lowest scores. One of such outputs is *"Greatly! They recognized my contributions for all these years."* The references achieve the best scores on three aspects, namely SIM, FLU, and COH. The last aspect is meant to measure the fitness of the outputs for the dialogue. Therefore, the best score is expected to be obtained on the references.

Among the three training configurations (i.e. standard, no emotion loss, and no paraphrasing step), the standard system has the best overall performance. However, its results are not too different from each other by a large margin. When taking the median (i.e. the middle value) and the mode (i.e. the most frequently occurring value), all systems are found to get either a score of 4 or 3, which can also be observed in Figure 12. This initial human evaluation, in addition to the observation on the overlap between these systems from Figure 10, might suggest that using our dataset, a simpler model, with neither the emotion loss nor paraphrasing step, nor both, can still perform emotionalisation with reasonable results. Yet, we believe a more extensive and comprehensive human evaluation should be conducted to find out whether this is true.

As mentioned earlier, the question about "context coherence" (`COH`) is meant to explore the fitness of the generated outputs to the dialogue if the generators use no information about the context (i.e. the previous utterances). A manual inspection of the scores leads us to believe that context should be taken into consideration for real-world applications of the text emotionalisation models. It is not entirely unexpected to find generated outputs which are scored high on emotion, similarity, and fluency metrics but do not fit in the dialogue. This is especially true when the generated outputs contain remarks to the previous utterances such as *"that's a shame"*, *"this is stupid"*, and *"great job"*. Furthermore, we find several cases in which the context is not clear enough for the assignment of `COH` scores. One of the examples is the input text number 4 (See Appendix B) which has the following sentence as the previous utterance: *"well, y'know, monogamy can be a, uh, tricky concept. I mean, anthropologically speaking-"*. Five system outputs for the evaluation are as follows.

1. reference: *"**fine. fine,** alright, now you'll never know."*
2. standard: *"alright, now you'll never know, **you stupid bastard.**"*
3. no emotion loss: *"alright, now you'll never know **how ridiculously stupid you are.**"*
4. no paraphrasing step: *"alright, now you'll never know. **how dare you.**"*
5. naive: *"alright, now you'll never know. **for god's sake!!**"*

Except for the 4th output, assigning scores to the remaining outputs is no easy task. The outputs are supposed to be of "anger" emotion. Still, for example, should the context is angry enough to justify calling the other person *"ridiculously stupid"* or *"stupid bastard"*? It seems that providing just one previous utterance as the context might not be enough. If possible, two or even three previous utterances in the dialogue should be included.

Lastly, we observe the scores given to the same output evaluated on the same aspect can be different by a large margin (e.g. 1, 5, 3). To improve the quality of human judgement, Karpinska et al. (2021) offer three suggestions. First, crowd workers need to have at least 90% approval rate and at least 1000 approved tasks (HITs) as suggested for best results. They also have to come from English-speaking countries, namely the USA, Canada, the UK, Ireland, Australia, New Zealand, and Singapore. During the verification process, the AMT workers who give the same score for all system outputs are rejected. After a further investigation, we observe that an additional requirement on minimum working time should be applied, i.e. working time should be longer than 30 seconds. Since the workers need to read the instruction and make judgement on 5 texts, anyone who takes less than that amount of time possibly makes a wild guess. Setting a maximum working time, on the other hand, is not straightforward since we have no information about the amount of time the workers open the tab without doing the task.

11 Further Discussions

In this section, we want to briefly discuss (1) the potential use of neutralisation models and (2) the issue of style and content in the Text Style Transfer task.

11.1 Neutralisation Models

We hypothesise our `Emotionalisation` dataset (Section 5.3) can be used to train neutralisation models in which the inputs are emotional texts and the outputs are neutral texts. These models would serve two purposes: (1) a data augmentation method for our text emotionalisation task, and (2) a bridge between different emotions for the task of Emotion Style Transfer, similar to the works done by Helbig et al. (2020).

Data augmentation: Our approach to creating parallel neutral-emotional data results in a small-scale `Emotionalisation` corpus. However, due to limited human resources, we are able to work only on a subset of sentences from the datasets mentioned in Section 5.1. That means a substantial number of emotional sentences are left unprocessed. We believe extending the corpus in an automatic fashion is theoretically possible via neutralisation models.

Using the same training procedure of the text emotionalisation models in Section 8.2, a PEGASUS paraphrase model is fed with emotional texts as inputs and is trained to generate neutral texts as outputs. Unlike the training of emotionalisation models, no explicit target emotion is used because the outputs is engineered towards one emotion only (i.e. neutral), not multiple emotions (i.e. anger, happiness, sadness) in which the emotionalisation models need more signal to know which emotion to generate.

A simple experiment is conducted to test our hypothesis. We train a PEGASUS neutralisation model as explained above and use it to make predictions on the unprocessed emotional sentences. The model achieves a score of 0.745 on emotion transfer accuracy metric. After the filtering step, out of 66k emotional input texts, 17k neutral texts remain. Manually inspecting the outputs, we observe that the model generally detects the emotion phrases and successfully re-writes the emotional texts into the grammatical neutral texts as illustrated in Table 15.

input:	**oh no! glad** your dad was able to help. how did it happen?
output:	so your dad was able to help? how did it happen?
input:	**it's good to appreciate** people who have helped you in the past.
output:	there are people who have helped you in the past.
input:	**oh my gosh, that is terrible!** I hope you were able to salvage it.
output:	i hope you were able to salvage it.
input:	**I can't wait for** our next reunion.
output:	our next reunion is soon.
input:	I am **happy** of this achievement and my family is **very proud**.
output:	my family and I have accomplished this.

Table 15: Examples of neutral texts generated by a PEGASUS neutralisation model.

However, it remains an unanswerable question whether the original content of the input texts are well preserved in cases that require paraphrasing such as the last example in Table 15. Furthermore, for multiple-sentence texts, the model frequently fails to generate the full content, missing the last sentence in the outputs.

We believe neutralisation models have the potential as a data augmentation method. The low-quality candidates can be filtered out, and high-quality outputs can be kept to train the model in an iterative manner. Nevertheless, the criteria of desired outputs should be precisely defined. For example, the syntax of the generated outputs should be different from the inputs to discourage samples with strict phrase deletion. Another criterion is to obtain only the samples containing the emotion phrases that are currently under-represented in our corpus.

Emotion style transfer: In Section 3.1, we define our text emotionalisation task as a subtask of the Emotion Style Transfer task by Helbig et al. (2020). It aims to transform an utterance of one emotion to another, using Ekman's 6 basic emotions (i.e. anger, disgust, fear, happiness, sadness, surprise). Our belief is that "neutral" emotion can be adopted as a bridge for the transformation between different emotions. Texts of one emotion can be fed into the

neutralisation models, and the neutralized outputs can be fed next into the emotionalisation models to obtain the texts of the desired emotion.

This indirect approach to Emotion Transfer is appealing compared to a direct method. It lifts the restriction on one-to-one position mapping between emotion words or phrases. As mentioned in Section 2.3.2, prototype editing in Style Transfer works on an assumption that once the part of the text which contains the original emotion can be identified, there exists a similar text candidate of a target emotion at that position in the sentence for the replacement to take place. Even if the generation is performed using neutral network models, the transformation still happens at around the identified positions (Sudhakar et al., 2019; Xu et al., 2018)[16]. We hypothesise an indirect approach should not impose such local location restrictions on the transferred sentences. It is because the knowledge about the location of the text with the original emotion is not passed to the emotionalisation models.

11.2 Style Versus Content

Recently, data-driven definition of "style" in the field of style transfer in texts (Tikhonov and Yamshchikov, 2018; Pang, 2019b,a). Additionally, the current approaches to the task (Section 2.3.2) make an incorrect presupposition that "style" is somehow fixed across different domains. As a result, one method is not applicable to all possible styles (Jafaritazehjani et al., 2021; Troiano et al., 2021). It is argued that content-, and style-related words should be defined according to the task, or even domain (Pang, 2019a). For example, in the sentiment transfer task, the word *"romantic"* is typically classified as positive style and thus is changed when being transferred to negative style. However, that assumption is flawed when the domain belongs to movie reviews. *"Romantic"* is a movie genre and thus should remain unchanged in the transfer (Pang, 2019a). In their hierarchy of styles, Troiano et al. (2021) classify "emotions" as being on the border between *style* and *content*. When this text of *happiness* emotion *"I am **happy** for you."* is modified for the transformation to *sadness* emotion *"I am **sad** for you."*, how much of original content can be preserved?

We face the same challenge in our work to create a parallel corpus in Section 5. Our effort to separate these two concepts for our project leads to an definition of *"emotion style"* as *explicitly expressing the emotion states of the speakers*, and *"emotion content"* as *the remaining words in the texts*. Yet, a clear boundary seems to be an elusive goal. For the sentence *"I am happy to see you here."*, it is possible to separate the emotion state *"I am happy"* from the content that *"I see you here."*. But what about *"I am happy for you."*?

[16]One example from the paper by Sudhakar et al. (2019) for negative-to-positive transfer is from *"the store is **dumpy** looking and management **needs to change**."* to *"the store is looking **great** and management to **perfection**."*

What should be the content after the part *"I am happy"* is taken away? We argue that any attempt to connect *"I"* and *"you"* with any verb might leads to a complete change in original meaning. Lastly, for the text *"My family is happy with my achievements."*, one possible transformation to the neutral text is *"My family knows about my achievements."*. Yet, arguably, the text can be understood as only having the content about the attitude of the speaker's family towards his/her achievements, and thus remains unchanged.

Emotion modification in texts involves both style and content (Helbig et al., 2020; Troiano et al., 2021). However, the focus should be shifted from the question of *the extent* of the original content can be preserved to the challenge of *the aspects* of the original content should be preserved. Emotionalising the neutral text *"I see some **people** out there."* into *"I see some **jerks** out there."*, we acknowledge the change of the original semantic meaning since **jerks** refers to *"stupid, annoying"* people and not simply *people*. However, our belief is this change does reflect the attitude of the speaker towards an entity while still keeping the underlying meaning of *people*. There is no definitive answer to the challenge of defining style and content. Rather, it would depend on the task, the domain, and what one believes should and should not be changed. Once a clear definition is establish, it will enable the selection and development of reliable evaluation metrics for the task.

12 Conclusions and Future Work

This thesis aims at exploring the task of emotionalising the neutral texts. Instead of adopting unsupervised methods for unparalleled corpora, we have tackled the challenge by defining *"emotion style"* as the parts of the texts that explicitly express the feelings of the speakers. A small parallel neutral-emotional corpus was created with such definition in mind. The implementation of text emotionalisation models is achieved using sequence-to-sequence models with paraphrasing as an intermediate pre-training stage for better content preservation. Our experiments demonstrate that the models are able to perform emotion transfer despite being trained on a small-scale corpus with 1-2k sentence pairs for each emotion transfer direction.

The emotion style loss and the paraphrasing stage are meant to improve further the transfer strength and the content preservation. Results on both automatic evaluation and human judgement, however, seem to suggest otherwise. A more comprehensive and well-designed human evaluation is needed to investigate this issue further. On the other hand, the naive method (i.e. randomly selecting a emotion phrase) performs better than our expectation, except on the sentence coherence metric (or "fluency" in our experiments). This raises a hypothetical question whether well-designed lists of emotion phrases and reliable ranking methods are all needed for a system that requires speed in

training and inference. However, we believe only deep learning models can perform more complete sentence transformations such as paraphrasing and adding emotion phrases in the middle of the text.

Our further analysis on experimental results and ablation studies show that future works should focus on the interpretation of evaluation metrics for the development of reliable methods to select our desired predictions. We believe the important prerequisite is having a deep understanding of the task domain, and thus, a clear definition of two concepts, namely "style" and "content". Furthermore, the emotion intensity should be considered as an additional parameter for the generation process.

Another potential area for exploration is the corpus itself. It would be interesting to have an analysis into its statistics for a better understanding on the diversity of emotion phrases and their frequency. Then, using data augmentation methods, one can create more training samples containing less frequent emotion phrases until the corpus is balanced. This might help increase the diversity score of the outputs. Another method is to influence the token distributions during decoding process in a way that more diverse tokens are sampled. For this purpose, one can experiment with different parameter choices of the generators.

In order to exploit context as an additional parameter, there are two possible approaches. The first approach makes a hypothetical assumption that the correct outputs exist in the decoding space. Thus, our job is to search for them using different methods including filtering, ranking techniques, or retrieval-based algorithms as they are more commonly adopted in dialogue generation task. The second approach is to explicitly model the context in the encoding step. We can concatenate the context with the neutral utterance into a long text, and use it as the input into the model. Additionally, the text emotionalisation models can be fed the explicit phrases which fit into the context dialogue.

In other words, the models can be adapted further to force them to generate the desired emotion phrases. Instead of having the target emotion as one of the inputs, we can replace it with the phrase. Thus, the models' job would become generating outputs containing the target phrase while ensuring grammar and preserving original content. However, this approach needs an additional model to select a suitable emotion phrase conditioning on the context of both the input sentence and the dialogue. The overall framework would become more complicated with the new model. However, if contexts are adopted for the generation, rather than concatenating the previous utterance and the input text into one long text, adding a classifier for emotion phrases is more feasible.

A AMT Interface for Human Evaluation

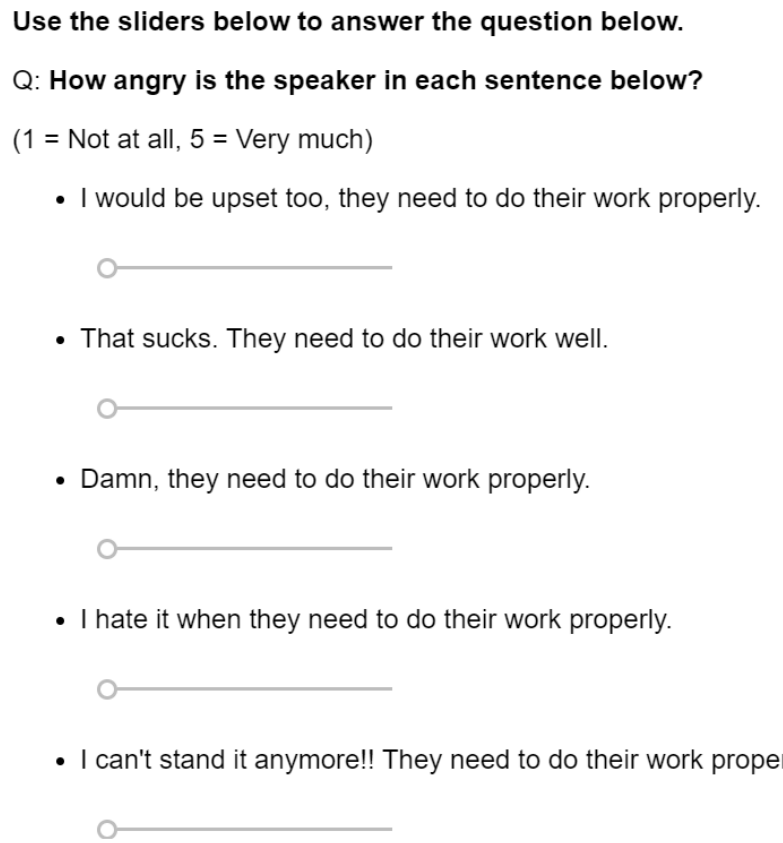


Figure 13: AMT interface for evaluating emotion transfer strength of five system outputs for one input text.

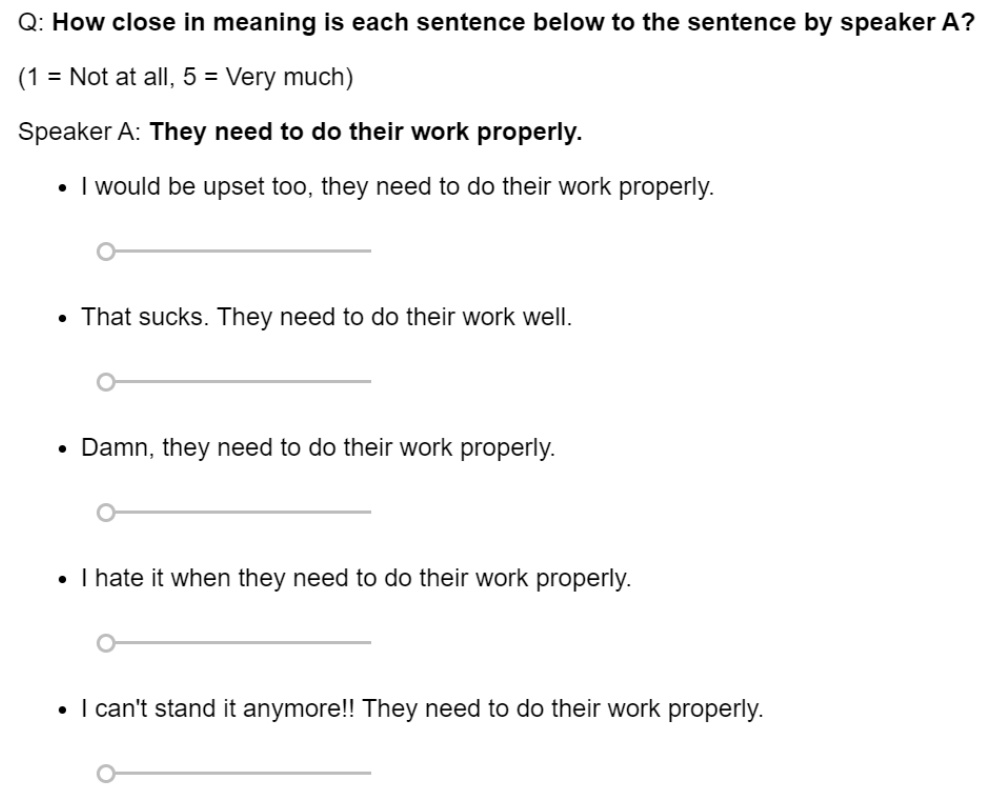


Figure 14: AMT interface for evaluating content preservation of five system outputs for one input text.

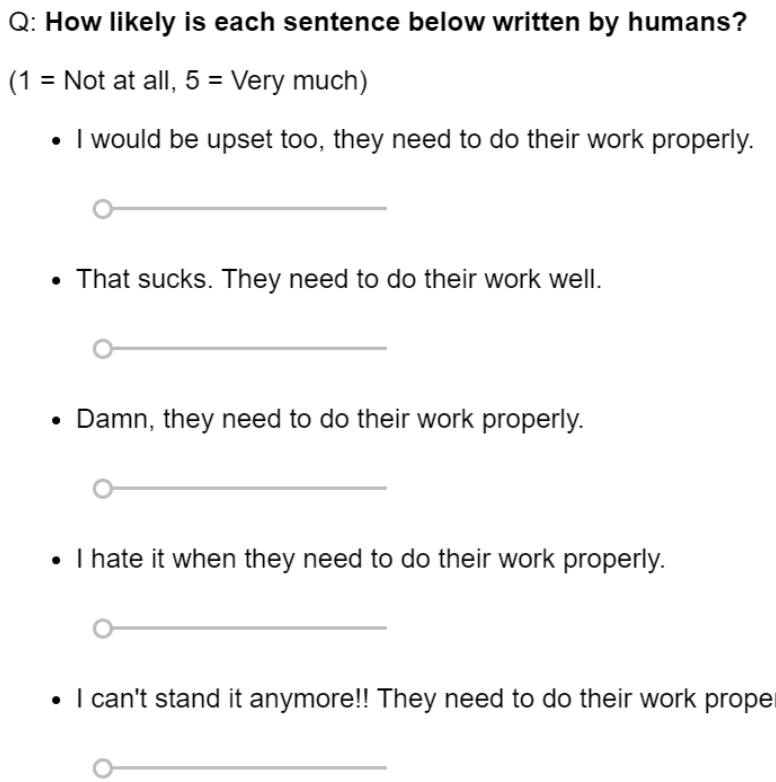


Figure 15: AMT interface for evaluating fluency of five system outputs for one input text.

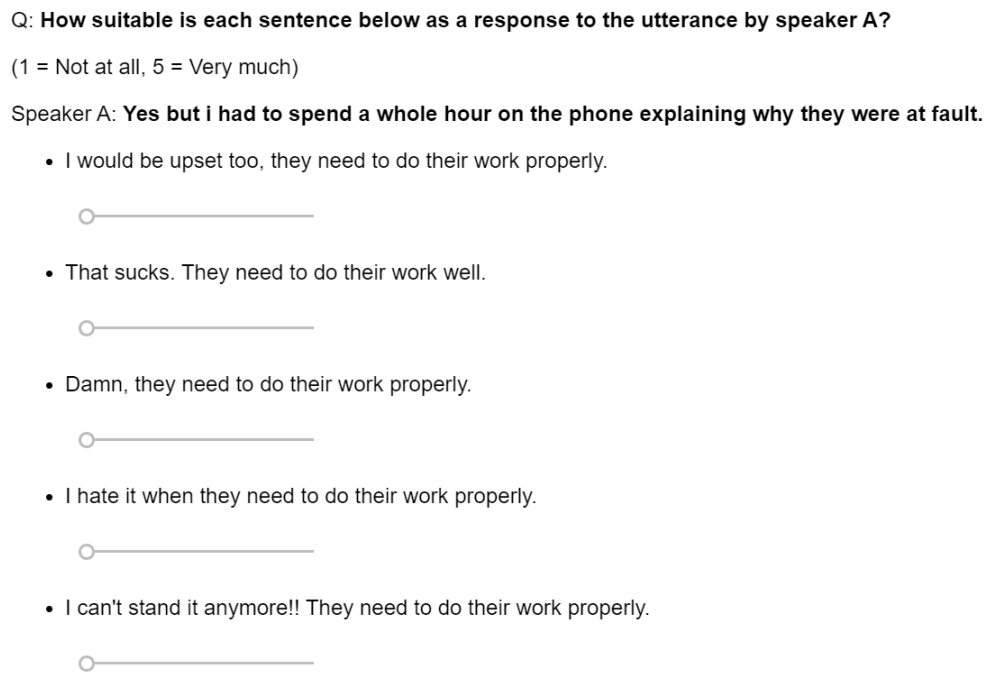


Figure 16: AMT interface for evaluating context coherence of five system outputs for one input text.

B Input Texts Used for Human Evaluation

Input text number 1
Previous utterance: My Kids Wouldn't Clean Their Room Yesterday. I Was Quite Displeased.
Target emotion: anger
Input text: I Hope You Pardon Them.
Reference: Oh No, I Hope You Pardon Them.
Pegasus standard: I Hope You Pardon Them, You Bastard.
Pegasus with nel: I Hope You Pardon Them. This Is Stupid.
Pegasus with nps: I Hope You Pardon Them. They Are Scumbags.
Naive system: This Is Highway Robbery. I Hope You Pardon Them.

Input text number 2
Previous utterance: Someone Keyed My Truck In A Restaraunt Parking Lot Last Night
Target emotion: anger
Input text: Did You Ask The Restaurant If They Have Video?
Reference: Oh No, Did You Ask The Restaurant If They Have Video?
Pegasus standard: How Dare You Ask The Restaurant If They Have Video?
Pegasus with nel: I Cannot Believe It! Did You Ask The Restaurant If They Have Video?
Pegasus with nps: I Hate When People Ask The Restaurant If They Have Video.
Naive system: I've Had Enough Of This!! Did You Ask The Restaurant If They Have Video?

Input text number 3
Previous utterance: I Swear My Friend Is Always Using Me
Target emotion: anger
Input text: Is She Your Friend Then?
Reference: That Sucks. Is She Really Your Friend Then?
Pegasus standard: Oh Damn! Is She Your Friend Then?
Pegasus with nel: Is She Your Friend Then? Fuck Off.
Pegasus with nps: Oh No! Is She Your Stupid Friend Then?
Naive system: You Are Such A Cheater!! Is She Your Friend Then?

Input text number 4
Previous utterance: Well, Y'Know, Monogamy Can Be A, Uh, Tricky Concept. I Mean, Anthropologically Speaking-
Target emotion: anger
Input text: Alright, Now You'll Never Know.
Reference: Fine. Fine, Alright, Now You'll Never Know.

Pegasus standard: Alright, Now You'll Never Know, You Stupid Bastard.
Pegasus with nel: Alright, Now You'll Never Know How Ridiculously Stupid You Are.
Pegasus with nps: Alright, Now You'll Never Know. How Dare You.
Naive system: Alright, Now You'll Never Know. For God's Sake!!

Input text number 5
Previous utterance: How Nice Seeing The Sun Back Again! The Air Is So Fresh.
Target emotion: happiness
Input text: I Hope It Stays Like This For A Long Period Of Time.
Reference: I Hope It Stays Nice Like This For A Long Period Of Time.
Pegasus standard: Wow! I Hope It Stays Like This For A Long Period Of Time.
Pegasus with nel: I Hope It Stays Like This For A Long Period Of Time. Great Job!
Pegasus with nps: Oh Wow! I Hope It Stays Like This For A Long Period Of Time.
Naive system: I Love It!! I Hope It Stays Like This For A Long Period Of Time.

Input text number 6
Previous utterance: It Is Beautiful! Look, There Are So Many Birds Around It.
Target emotion: happiness
Input text: It Is A Place For A Vacation.
Reference: It Is A Great Place For A Relaxing Vacation.
Pegasus standard: It Is A Great Place To Have A Vacation.
Pegasus with nel: It Is A Wonderful Place To Take A Vacation.
Pegasus with nps: It Is A Perfect Place For A Great Vacation.
Naive system: Whooo Hooo!!!! It Is A Place For A Vacation.

Input text number 7
Previous utterance: I Want Croatia To Win Because They Are The Underdogs.
Target emotion: happiness
Input text: I See An Underdog Win.
Reference: It's Always Good To See An Underdog Win!
Pegasus standard: I'm Glad To See An Underdog Win.
Pegasus with nel: I Am Thrilled To See An Underdog Win.
Pegasus with nps: I Am Excited To See An Underdog Win.
Naive system: I See An Underdog Win. Oh, So Wonderful.

Input text number 8
Previous utterance: So Honey, This Morning Was Fun, Huh? Me Hopping In On You In The Shower There.
Target emotion: happiness
Input text: And Maybe Someday We Could Get A Place With Two Bathrooms.
Reference: Yeah! And Maybe Someday We Could Get A Place With Two Bathrooms.
Pegasus standard: And Maybe Someday We Could Get A Place With Two Bathrooms. Hahaha
Pegasus with nel: And Maybe Someday We Could Get A Place With Two Bathrooms. Wow!
Pegasus with nps: And Maybe Someday We Could Get A Place With Two Bathrooms. Lol
Naive system: And Maybe Someday We Could Get A Place With Two Bathrooms. I Really Appreciate It.

Input text number 9
Previous utterance: What's The Matter, Dear ? **Target emotion:** sadness
Input text: Something Happened. We Went To The Portabello Road, And Someone Took My Purse.
Reference: Something Awful Happened. We Went To The Portobello Road, And Someone Stole My Handbag.
Pegasus standard: Something Terrible Happened. We Went To The Portabello Road And Someone Stole My Purse. I Was So Scared.
Pegasus with nel: Something Terrible Happened. We Went To The Portabello Road And Someone Took My Purse.
Pegasus with nps: Something Really Bad Happened. We Went To The Portabello Road, And Someone Took My Purse.
Naive system: I Am Worried About It! Something Happened. We Went To The Portabello Road, And Someone Took My Purse.

Input text number 10
Previous utterance: Look! They've Taken Your Fur Coat Too.
Target emotion: sadness
Input text: I Know, I Bought It.
Reference: I Know, I Am Sorry I Ever Bought It. **Pegasus standard:** I Know, It Was Embarrassing. I Bought It.
Pegasus with nel: I Know, That Was My Mistake. I Bought It.
Pegasus with nps: I Know, I Bought It. I Am So Sorry.
Naive system: C'mon, Don't Blame Me!! I Know, I Bought It.

Input text number 11

Previous utterance: It Felt Bad To Finish Last In A Race Last Week.
Target emotion: sadness
Input text: ... Did You Come Last??
Reference: Really Bad... Did You Come Last??
Pegasus standard: Oh Dear. Did You Come Last??
Pegasus with nel: I'm Sorry, Did You Come Last??
Pegasus with nps: Oh No! Did You Come Last??
Naive system: ... Did You Come Last?? How Horrible!!

Input text number 12
Previous utterance: My Dog Is Missing! She Has Been Gone For Almost A Week.
Target emotion: sadness
Input text: Have You Posted Everywhere And Checked Shelters?
Reference: Oh My God! Have You Posted Everywhere And Checked Shelters?
Pegasus standard: Oh No! Have You Posted Everywhere And Checked Shelters?
Pegasus with nel: Oh Gosh! Have You Posted Everywhere And Checked Shelters?
Pegasus with nps: Shame On You. Have You Posted Everywhere And Checked Shelters?
Naive system: I Am Still Embarrassed! Have You Posted Everywhere And Checked Shelters?

Input text number 13
Previous utterance: My Dog Pooped In My Wig Today. I Was So Mad!
Target emotion: anger
Input text: How Are You Going To Clean It Out?
Reference: Oh No! How Are You Going To Clean It Out?
Pegasus standard: How Are You Gonna Clean It Out? It Sucks.
Pegasus with nel: Damn, How Are You Going To Clean It Out?
Pegasus with nps: Oh No! How Are You Going To Clean That Shit Out?
Naive system: How Are You Going To Clean It Out? Damnit!

Input text number 14
Previous utterance: I Think Someone Stole My Headphones Recently.
Target emotion: anger
Input text: Do You Have A Suspect?
Reference: Oh No! Do You Have A Suspect?
Pegasus standard: Oh No! Do You Have An Actual Suspect?
Pegasus with nel: Oh My God! Do You Have A Suspect?
Pegasus with nps: Oh No! Do You Have A Bloody Suspect?

Naive system: I Can't Stand It Anymore!! Do You Have A Suspect?

Input text number 15
Previous utterance: Yes But I Had To Spend A Whole Hour On The Phone Explaining Why They Were At Fault.
Target emotion: anger
Input text: They Need To Do Their Work Properly.
Reference: I Would Be Upset Too, They Need To Do Their Work Properly.
Pegasus standard: That Sucks. They Need To Do Their Work Well.
Pegasus with nel: Damn, They Need To Do Their Work Properly.
Pegasus with nps: I Hate It When They Need To Do Their Work Properly.
Naive system: I Can't Stand It Anymore!! They Need To Do Their Work Properly.

Input text number 16
Previous utterance: School Starts Next Week And I Am All Ready To Go!
Target emotion: happiness
Input text: Are You Taking Any Classes?
Reference: Oh Nice! Are You Taking Any Cool Classes?
Pegasus standard: Oh Wow! Are You Taking Any Classes?
Pegasus with nel: Oh Great! Are You Taking Any Classes?
Pegasus with nps: Congratulations! Are You Taking Any Classes?
Naive system: Are You Taking Any Classes? I'll Be Extremely Glad!

Input text number 17
Previous utterance: That's What I'm Talking About! That Sounds So Nice. I'm A Little Jealous. Sounds Like You Deserve It, Though.
Target emotion: happiness
Input text: They Recognized My Contributions For All These Years.
Reference: I Was Thankful That They Recognized My Contributions For All These Years.
Pegasus standard: They Recognized My Contributions For All These Years. It Was Great!
Pegasus with nel: I Am So Happy That They Recognized My Contributions For All These Years.
Pegasus with nps: I Was Delighted When They Recognized My Contributions For All These Years.
Naive system: Greatly! They Recognized My Contributions For All These Years.

Input text number 18
Previous utterance: I'm Feeling Blue Because My Days And Nights Are Mixed Up

Target emotion: sadness
Input text: Have You Tried Taking Some Melatonin?
Reference: That Really Sucks. Have You Tried Taking Some Melatonin?
Pegasus standard: That's Not Good. Have You Tried Taking Some Melatonin?
Pegasus with nel: Oh Dear, Have You Tried Taking Some Melatonin?
Pegasus with nps: I'm Sorry But Have You Tried Taking Some Melatonin?
Naive system: Have You Tried Taking Some Melatonin? What A Let Down.

Input text number 19
Previous utterance: I Was Trapped In A Cave With My Mates For 15 Days
Target emotion: sadness
Input text: It's Impossible To Stay For 15 Days **Reference:** I Feel So Worried About This It's Impossible To Stay For 15 Days
Pegasus standard: I'm Sorry But It's Impossible To Stay For 15 Days
Pegasus with nel: It's Very Difficult To Stay For 15 Days
Pegasus with nps: It's A Shame It's Impossible To Stay For 15 Days
Naive system: It's Impossible To Stay For 15 Days ! It's Embarrassing!

Input text number 20
Previous utterance: My Brother Left The Gate Open And My Dog Got Out, Needless To Say My Brother Got Beatdown.
Target emotion: sadness
Input text: I Hope You Found Your Dog.
Reference: That Sucks, I Hope You Found Your Dog.
Pegasus standard: I'm So Sorry You Found Your Dog.
Pegasus with nel: That's A Shame. I Hope You Found Your Dog.
Pegasus with nps: That Is So Sad. I Hope You Found Your Dog.
Naive system: I Hope You Found Your Dog. What Bad Luck !!

Input text number 21
Previous utterance: I Could Easily Give You Some Time Tomorrow Night.
Target emotion: anger
Input text: Do You Meet Me At The Coffee House Next Door ?
Reference: Would You Like To Meet Me At The Coffee House Next Door?
Pegasus standard: Are You Excited To Meet Me At The Coffee House Next Door?
Pegasus with nel: I Am Looking Forward To Meeting You At The Coffee House Next Door?
Pegasus with nps: I'd Love To Meet You At The Coffee House Next Door?
Naive system: That Sounds Fantastic. Do You Meet Me At The Coffee House Next Door?

References

Babak Joze Abbaschian, Daniel Sierra-Sosa, and Adel Said Elmaghraby. 2021. Deep learning techniques for speech emotion recognition, from databases to models. *Sensors (Basel, Switzerland)*, 21.

Francisca Adoma Acheampong, Henry Nunoo-Mensah, and Wenyu Chen. 2021. Transformer models for text-based emotion detection: a review of bert-based approaches. *Artificial Intelligence Review*.

Ameeta Agrawal, Aijun An, and Manos Papagelis. 2018. Learning emotion-enriched word representations. In *COLING*.

Cecilia Ovesdotter Alm, Dan Roth, and Richard Sproat. 2005. Emotions from text: Machine learning for text-based emotion prediction. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 579–586, Vancouver, British Columbia, Canada. Association for Computational Linguistics.

Nabiha Asghar, P. Poupart, J. Hoey, Xin Jiang, and Lili Mou. 2018. Affective neural response generation. *European Conference on Information Retrieval*, pages 154–166.

Eleftheria Briakou, Sweta Agrawal, Ke Zhang, Joel Tetreault, and Marine Carpuat. 2021. 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, Online. Association for Computational Linguistics.

Pawel Bujnowski, Kseniia Ryzhova, Hyungtak Choi, Katarzyna Witkowska, Jaroslaw Piersa, Tymoteusz Krumholc, and Katarzyna Beksa. 2020. An empirical study on multi-task learning for text style transfer and paraphrase generation. In *Proceedings of the 28th International Conference on Computational Linguistics: Industry Track*, pages 50–63, Online. International Committee on Computational Linguistics.

Chris Callison-Burch, Miles Osborne, and Philipp Koehn. 2006. Re-evaluating the role of Bleu in machine translation research. In *11th Conference of the European Chapter of the Association for Computational Linguistics*, pages 249–256, Trento, Italy. Association for Computational Linguistics.

Federica Cavicchio and Massimo Poesio. 2008. Annotation of emotion in dialogue: The emotion in cooperation project. In *Perception in Multimodal Dialogue Systems. PIT*, volume 5078. Springer.

Jianpeng Cheng, Li Dong, and Mirella Lapata. 2016. Long short-term memory-networks for machine reading. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 551–561, Austin, Texas. Association for Computational Linguistics.

Pierre Colombo, Wojciech Witon, Ashutosh Modi, James Kennedy, and Mubbasir Kapadia. 2019. Affect-driven dialog generation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3734–3743, Minneapolis, Minnesota. Association for Computational Linguistics.

Sumanth Dathathri, Andrea Madotto, Janice Lan, Jane Hung, Eric Frank, Piero Molino, J. Yosinski, and Rosanne Liu. 2020. Plug and play language models: A simple approach to controlled text generation. *ICLR*.

Dorottya Demszky, Dana Movshovitz-Attias, Jeongwoo Ko, Alan Cowen, Gaurav Nemade, and Sujith Ravi. 2020. GoEmotions: A dataset of fine-grained emotions. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4040–4054, Online. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Tomasz Dryjanski, Pawel Bujnowski, Hyungtak Choi, Katarzyna Podlaska, Kamil Michalski, Katarzyna Beksa, and Pawel Kubik. 2018. Affective natural language generation by phrase insertion. *2018 IEEE International Conference on Big Data (Big Data)*, pages 4876–4882.

Paul Ekman. 1992. An argument for basic emotions. *Cognition & Emotion*, 6:169–200.

Albert Gatt and E. Krahmer. 2018. Survey of the state of the art in natural language generation: Core tasks, applications and evaluation. *Journal of Artificial Intelligence Research*, 61:65–170.

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

Tushar Goswamy, Ishika Singh, Ahsan Barkati, and Ashutosh Modi. 2020. Adapting a language model for controlled affective text generation. In *Pro-*

ceedings of the 28th International Conference on Computational Linguistics, pages 2787–2801, Barcelona, Spain (Online). International Committee on Computational Linguistics.

David Helbig, Enrica Troiano, and Roman Klinger. 2020. Challenges in emotion style transfer: An exploration with a lexical substitution pipeline. In *Proceedings of the Eighth International Workshop on Natural Language Processing for Social Media*, pages 41–50, Online. Association for Computational Linguistics.

Chao-Chun Hsu, Sheng-Yeh Chen, Chuan-Chun Kuo, Ting-Hao Huang, and Lun-Wei Ku. 2018. EmotionLines: An emotion corpus of multi-party conversations. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

J. Edward Hu, Abhinav Singh, Nils Holzenberger, Matt Post, and Benjamin Van Durme. 2019. Large-scale, diverse, paraphrastic bitexts via sampling and clustering. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 44–54, Hong Kong, China. Association for Computational Linguistics.

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

Somayeh Jafaritazehjani, Gwénolé Lecorvé, Damien Lolive, and John D. Kelleher. 2021. Style as sentiment versus style as formality: The same or different? In *Artificial Neural Networks and Machine Learning – ICANN 2021*, pages 487–499, Cham. Springer International Publishing.

Di Jin, Zhijing Jin, Zhiting Hu, Olga Vechtomova, and Rada Mihalcea. 2020. Deep learning for text style transfer: A survey. *ArXiv*, abs/2011.00416.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. 2020. SpanBERT: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 8:64–77.

Marzena Karpinska, Nader Akoury, and Mohit Iyyer. 2021. The perils of using Mechanical Turk to evaluate open-ended text generation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1265–1285, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Kalpesh Krishna, John Wieting, and Mohit Iyyer. 2020. Reformulating unsupervised style transfer as paraphrase generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*

(EMNLP), pages 737–762, Online. Association for Computational Linguistics.

Huiyuan Lai, Antonio Toral, and Malvina Nissim. 2020. Generic resources are what you need: Style transfer tasks without task-specific parallel training data.

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

Chris van der Lee, Albert Gatt, Emiel van Miltenburg, and Emiel J. Krahmer. 2021. Human evaluation of automatically generated text: Current trends and best practice guidelines. *Computer Speech Language*, 67:101151.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016. A diversity-promoting objective function for neural conversation models. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 110–119, San Diego, California. Association for Computational Linguistics.

Juncen Li, 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, New Orleans, Louisiana. Association for Computational Linguistics.

Yanran Li, Hui Su, Xiaoyu Shen, Wenjie Li, Ziqiang Cao, and Shuzi Niu. 2017. DailyDialog: A manually labelled multi-turn dialogue dataset. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 986–995, Taipei, Taiwan. Asian Federation of Natural Language Processing.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *ArXiv*, abs/1907.11692.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *ICLR*.

Yukun Ma, Khanh Linh Nguyen, Frank Z. Xing, and E. Cambria. 2020. A survey on empathetic dialogue systems. *Information Fusion*, 64:50–70.

Eric Malmi, 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)*, pages 8671–8680, Online. Association for Computational Linguistics.

M. McTear, Z. Callejas, and D. Griol. 2016. The conversational interface: Talking to smart devices. volume 6.

Remi Mir, 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, Minneapolis, Minnesota. Association for Computational Linguistics.

Saif Mohammad. 2018. Obtaining reliable human ratings of valence, arousal, and dominance for 20,000 English words. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 174–184, Melbourne, Australia. Association for Computational Linguistics.

Saif M. Mohammad and Peter D. Turney. 2010. Emotions evoked by common words and phrases: Using mechanical turk to create an emotion lexicon. In *HLT-NAACL 2010*.

Charles E. Osgood. 1952. The nature and measurement of meaning. *Psychological bulletin*, 49 3:197–237.

Charles E. Osgood, G. Suci, and P. Tannenbaum. 1957. The measurement of meaning. University of Illinois Press.

Richard Yuanzhe Pang. 2019a. The daunting task of real-world textual style transfer auto-evaluation. *ArXiv*, abs/1910.03747.

Richard Yuanzhe Pang. 2019b. Towards actual (not operational) textual style transfer auto-evaluation. In *Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019)*, pages 444–445, Hong Kong, China. Association for Computational Linguistics.

Richard Yuanzhe Pang 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*, pages 138–147, Hong Kong. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In *NeurIPS*.

Robert Plutchik. 1980. A general psychoevolutionary theory of emotion.

Soujanya Poria, Devamanyu Hazarika, Navonil Majumder, Gautam Naik, Erik Cambria, and Rada Mihalcea. 2019. MELD: A multimodal multi-party dataset for emotion recognition in conversations. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 527–536, Florence, Italy. Association for Computational Linguistics.

Soujanya Poria, Navonil Majumder, Devamanyu Hazarika, Deepanway Ghosal, Rishabh Bhardwaj, Samson Yu, Romila Ghosh, Niyati Chhaya, Alexander F. Gelbukh, and Rada Mihalcea. 2021. Recognizing emotion cause in conversations. *Cognitive Computation*.

Shrimai Prabhumoye, Yulia Tsvetkov, Ruslan Salakhutdinov, and Alan W Black. 2018. Style transfer through back-translation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 866–876, Melbourne, Australia. Association for Computational Linguistics.

Alec Radford and Karthik Narasimhan. 2018. Improving language understanding by generative pre-training.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Colin Raffel, Noam M. 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. *ArXiv*, abs/1910.10683.

Hannah Rashkin, Eric Michael Smith, Margaret Li, and Y-Lan Boureau. 2019. Towards empathetic open-domain conversation models: A new benchmark and dataset. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5370–5381, Florence, Italy. Association for Computational Linguistics.

Machel Reid and Victor Zhong. 2021. LEWIS: Levenshtein editing for unsupervised text style transfer. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3932–3944, Online. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. 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 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Fiorella De Rosis and Floriana Grasso. 2000. Affective natural language generation. In *Affective interactions*, pages 204–218. Springer.

Sebastian Ruder. 2019. *Neural Transfer Learning for Natural Language Processing*. Ph.D. thesis, National University of Ireland, Galway.

Elvis Saravia, Hsien-Chi Toby Liu, Yen-Hao Huang, Junlin Wu, and Yi-Shin Chen. 2018. CARER: Contextualized affect representations for emotion recognition. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3687–3697, Brussels, Belgium. Association for Computational Linguistics.

Klaus Scherer. 2005. What are emotions? and how can they be measured? *Social Science Information*, 44:695 – 729.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Ashish Sharma, Inna Wanyin Lin, Adam S. Miner, David C. Atkins, and Tim Althoff. 2021. Towards facilitating empathic conversations in online mental health support: A reinforcement learning approach. *Proceedings of the Web Conference 2021*.

T. Shen, Tao Lei, R. Barzilay, and T. Jaakkola. 2017. Style transfer from non-parallel text by cross-alignment. In *NIPS*.

Eric Michael Smith, Diana Gonzalez-Rico, Emily Dinan, and Y-Lan Boureau. 2019. Zero-shot fine-grained style transfer: Leveraging distributed continuous style representations to transfer to unseen styles. *ArXiv*, abs/1911.03914.

Zhenqiao Song, Xiaoqing Zheng, Lu Liu, Mu Xu, and Xuanjing Huang. 2019. Generating responses with a specific emotion in dialog. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3685–3695, Florence, Italy. Association for Computational Linguistics.

Carlo Strapparava and Alessandro Valitutti. 2004. Wordnet affect: an affective extension of wordnet. In *LREC*.

Akhilesh Sudhakar, Bhargav Upadhyay, and Arjun Maheswaran. 2019. "transforming" delete, retrieve, generate approach for controlled text style transfer. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3269–3279, Hong Kong, China. Association for Computational Linguistics.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems*, volume 2, pages 3104—-3112. MIT Press.

Guy Tevet and Jonathan Berant. 2021. Evaluating the evaluation of diversity in natural language generation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 326–346, Online. Association for Computational Linguistics.

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

Enrica Troiano, Roman Klinger, and Sebastian Padó. 2020. Lost in back-translation: Emotion preservation in neural machine translation. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4340–4354, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Enrica Troiano, Aswathy Velutharambath, and Roman Klinger. 2021. From theories on styles to their transfer in text: Bridging the gap with a hierarchical survey.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is

all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Xiangyu Wang and Chengqing Zong. 2021. Distributed representations of emotion categories in emotion space. 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 2364–2375, Online. Association for Computational Linguistics.

Alex Warstadt, Amanpreet Singh, and Samuel R. Bowman. 2019. Neural network acceptability judgments. *Transactions of the Association for Computational Linguistics*, 7:625–641.

John Wieting, Taylor Berg-Kirkpatrick, Kevin Gimpel, and Graham Neubig. 2019. Beyond BLEU:training neural machine translation with semantic similarity. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4344–4355, Florence, Italy. Association for Computational Linguistics.

John Wieting and Kevin Gimpel. 2018. ParaNMT-50M: Pushing the limits of paraphrastic sentence embeddings with millions of machine translations. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 451–462, Melbourne, Australia. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv*, abs/1910.03771.

Xing Wu, Tao Zhang, Liangjun Zang, Jizhong Han, and Songlin Hu. 2019. Mask and infill: Applying masked language model for sentiment transfer. In *IJCAI*.

Jingjing Xu, 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, Melbourne, Australia. Association for Computational Linguistics.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J. Liu. 2020a. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. *International Conference on Machine Learning*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020b. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Hao Zhou, Minlie Huang, Tianyang Zhang, Xiaoyan Zhu, and Bing Liu. 2018. Emotional chatting machine: Emotional conversation generation with internal and external memory. In *AAAI*.

Jianing Zhou and Suma Bhat. 2021. Paraphrase generation: A survey of the state of the art. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5075–5086, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.