Traversing the Landscapes of Sentiment: Generalized Sentiment Transfer

Stanford CS224N Custom Project, **Mentor:** Anna Goldie

Maciej Kurzynski
Department of East Asian Languages and Cultures
Stanford University
makurz@stanford.edu

Abstract

Sentiment transfer is an important area of research in Natural Language Processing. Given a piece of text and its dominant sentiment (e.g. *happiness*, *sadness*, *anger*, *surprise* etc.), the goal of sentiment transfer is to express the original semantic content, but through a different sentiment. Existing solutions focus on binary transfers (e.g., from positive to negative), and/or sentiment intensity (e.g., a scalar value between 0 and 1). These approaches, however, neglect the complexity of human emotionality and how this emotionality is represented in texts. In our preliminary exploration, we leverage the potential of large transformer models and introduce a set of problems as well as a training technique called Generalized Sentiment Transfer, or GST. During training, we provide a generalizable target sentiment signal to the model so that it ultimately learns to transfer semantic information between a wider, potentially unlimited range of sentiments.

1 Introduction

"Style transfer" models facilitate many current NLP applications, such as automatic conversion of paper titles to news titles, poetry generation, transformation of plain English into Shakespearean English or modern Chinese into classical Chinese, reducing the language bias and toxicity, and paraphrasing (Fu et al. (2017)). Outside of NLP, one of the most popular applications is the style transfer in images (e.g., transforming photographs into paintings à la van Gogh).

The key challenge in the family of problems known as "style transfer" is to separate the content of the source sentence from other aspects, such as style. One major obstacle is the lack of supervised parallel data (corpora where each pair of sentences describes the same content while expressing different sentiment). As a consequence, it becomes non-trivial to design targets and metrics against which the model's outputs can be evaluated during training. For example, two phrases can use different vocabulary and still convey identical sentiment (e.g., *I feel wonderful today* and *I am in a great mood today*). On the other hand, the sentences *I am feeling confused right now* and *I am feeling wonderful right now* convey different sentiments, despite their semantic and grammatical similarity. Moreover, many currently available models have difficulties classifying sentences featuring negation (*I am not in a great mood today*) or phrases requiring sophisticated cultural background to be properly understood, such as rhetorical questions (*Do you think I am happy about it?*). Finally, the level of complexity grows substantially in the case of longer passages featuring ambiguous sentiments or phrases transitioning between sentiments (e.g. *I was feeling fabulous until this morning when I realized that the deadline is approaching*).

It is our conviction that to better understand the problem of sentiment transfer, it is necessary to broaden the scope of sentiments that language models can detect and translate between. Literature scholars such as Nussbaum (1990), Zunshine (2006), and Sklar (2013) have argued that by reading novels, humans refine their "mind-reading" skills and develop sensitivity to other people's feelings (empathy). We learn to interpret the slightest changes in other people's speech and behavior as indicators of their current emotional states. In order to allow language models to acquire comparable complexity and sensitivity, we need to train beyond the binary.

2 Related Work

Numerous solutions to sentiment transfer have been proposed so far (see Hu et al. (2020) for a comprehensive review). A popular approach adopted by many NLP researchers has been to train an LSTM-based model to map a sentence in one style to a style-independent vector, and then decode this vector to a new sentence featuring the same content but a different style (Shen et al. (2017)). Li et al. (2018) designed a "Delete, Retrieve, Generate" model which provided a baseline and inspiration for many scholars in recent years. Specifically, they delete phrases associated with the sentence's original attribute (style) value, retrieve new phrases associated with the target attribute from the dataset, and then use a neural model to fluently combine these into a final output. To tackle the problem of scarce parallel data, Xu et al. (2018) proposed a cycled reinforcement learning approach that consists of two parts: a neutralization module and an emotionalization module. The neutralization module is responsible for extracting non-emotional semantic information by explicitly filtering out emotional words. The emotionalization module, on the other hand, is responsible for adding sentiment to such neutralized semantic content in order to achieve the final sentiment-to-sentiment translation. Since there is no parallel data available to evaluate performance, the model learns through reinforcement, maximizing the BLEU score and sentiment transfer accuracy. A variation of this approach has been proposed by Luo et al. (2019), where the numeric sentiment intensity value is incorporated into the decoder so as to finely control the sentiment intensity of the output. Sudhakar et al. (2019) introduced the Generative Style Transformer (GST) to rewrite sentences to a target style, benefiting from the power of large unsupervised pre-trained language models and the Transformer architecture. Sun et al. (2020) further improved upon the reinforcement learning approach by using two self-attention layers (for semantics and sentiment, respectively).

3 Approach

Sudhakar et al. (2019) formulated the problem of style transfer in the following way. Given a dataset $D = \{(x_1, s_1), ..., (x_m, s_m)\}$ where x_i is a sentence and $s_i \in S$ is a specific style, the goal is to learn a conditional distribution $P(y|x, s^{tgt})$ where y is a sentence and $style(y) = s^{tgt}$, and where $style$ is determined by an oracle that can accurately determine the style of a given sentence. Following Xu et al. (2018), we model our task in two steps: **(1) neutralization** module which learns $P(c|x)$ such that c is the non-stylistic component of x and $style(c) \notin S$ (i.e., c does not have any particular style), and **(2) emotionalization** module which learns $P(y|c, s_x)$, where c is the neutralized content and s_x is the original sentiment of x.

In what follows, we propose the **Generalized Sentiment Transfer**, or **GST**: instead of indicating the target sentiment s^{tgt} as a discrete value (e.g., 0 for *negative*, 1 for *positive*) or a fraction between 0 and 1 (see Mousa and Schuller (2017); Xu et al. (2018); Luo et al. (2019)), we provide a generalizable sentiment signal to the model. We design two variations of the emotionalization module. In the first approach (**GST-k**), we provide one keyword as the only sentiment signal available to the model (e.g., *anger* or *love*); in the second approach (**GST-s**), we provide yet another sentence to indicate the desired target sentiment. In both cases, the model needs to remove the sentiment information from the input (semantic) sentence, extract the sentiment information from the sentiment signal ("mood"), and then produce a sentence containing the original semantic content but now interpreted from the perspective of the new sentiment. Our goal is to train a model in such a way that it learns to generalize to other possible sentiment keywords unseen during the training time (for **GST-k**), as well as other possible "mood" sentences (for **GST-s**).

3.1 Neutralization

We first construct a multi-label sentiment classifier to extract emotionally salient words. The classifier encodes each input (word) into a contextual representation through multi-head self-attention. These contextual representations are then averaged and passed through another linear layer to produce a context vector, different for each sentence, which is then projected onto the final *n* dimensional

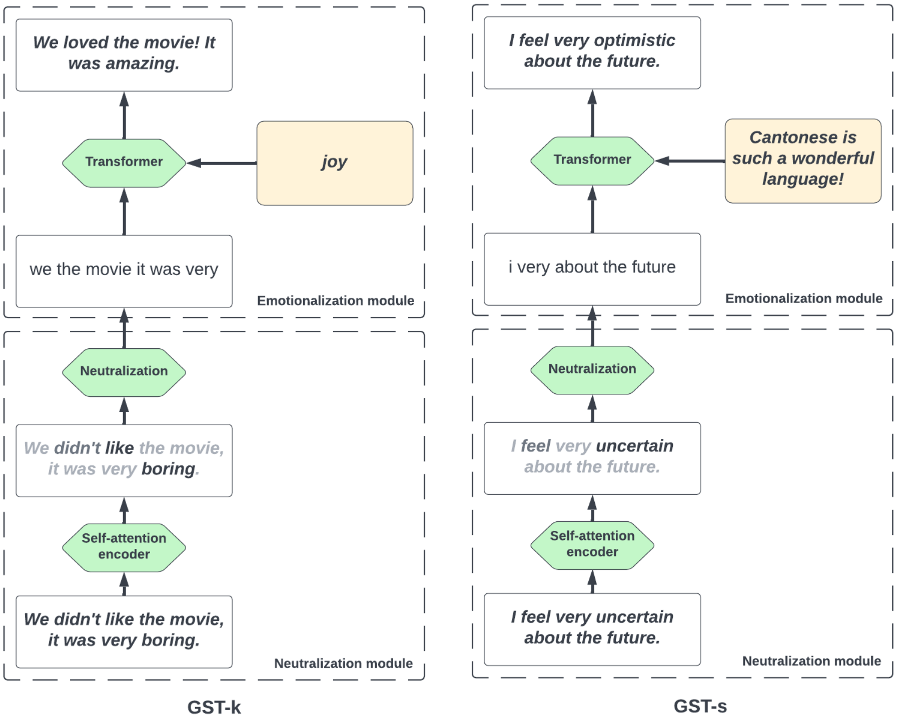
We loved the movie! It was amazing.
Transformer
joy
we the movie it was very
Emotionalization module
Neutralization
We didn't like the movie, it was very boring.
Self-attention encoder
We didn't like the movie, it was very boring.
Neutralization module
GST-k
I feel very optimistic about the future.
Transformer
Cantonese is such a wonderful language!
i very about the future
Emotionalization module
Neutralization
I feel very uncertain about the future.
Self-attention encoder
I feel very uncertain about the future.
Neutralization module
GST-s

Figure 1: Generalized Sentiment Transfer

Original sentence	**Neutralized sentence**	**Sentiment**
I plan to share my everyday life **stories**, traveling **adventures**, **inspirations**, and handmade creations with you, and **hope** you will also feel **inspired**.	i plan to share my everyday life traveling and handmade with you and you will also feel	*joy*
I am feeling **bitter** today, my mood has been **strange** the entire day, so I guess it's that.	i am feeling today my has been the entire day so i it is that	*anger*
I will adjust to it but for now it **feels** so **strange**.	i will adjust to it for now it so	*surprise*

Table 1: Examples of sentiment neutralization

vector, where *n* is the number of distinct sentiments. As the model learns to classify sentences from the training dataset, the self-attention module simultaneously learns to detect emotionally salient words and phrases.

To neutralize the corpus and extract the pure neutralized information, we remove the sentimentally-charged words that most strongly influence the classifier's predictions. However, since every head in the self-attention module encodes different aspects of semantic and linguistic structure and thus distributes attention differently, we need to find the head(s) that "specialize(s)" in detecting emotionally charged words. The attention score α for every word $w \in x$ for head h is calculated as follows:

$$\alpha_h(w) = softmax_{w \in x}(Q_h K_h^T) \tag{1}$$

where Q and K indicate the query and key vectors as used by Vaswani et al. (2017) in the Transformer architecture:

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{2}$$

Inspired by Sudhakar et al. (2019), we use a brute force method: for every head, we remove the words with above-the-average attention scores from each sentence, and then run the classifier again on the whole corpus of such reduced sentences (e.g., for 16 heads, we end up with 16 different corpora). The corpus that is now most difficult to classify indicates the head that is performing the most accurate distribution of attention, since it correctly detected features serving as markers of sentiment. In the emotionalization module (section 3.2) we combine the attention scores from the two best performing heads to remove the sentiment information from the corpus.

3.2 Emotionalization

Having neutralized our inputs, we now fine-tune a pre-trained GPT2 model. GPT-family models are fine-tuned using a causal language modeling (CLM) loss – at each iteration, the model predicts the next token given a sequence of tokens; the prediction is then compared with the target token. We then replace the predicted word with the actual word and move on to predict the next token in the sequence etc, which is a method called "teacher forcing." However, since we do not have parallel corpora to measure accuracy of sentiment transfer for each sentence rendered in different sentiments, during training we measure only the reconstruction loss. Specifically, we provide prompts to the model containing the sentiment signal, neutralized sentence, and the original sentence (reconstruction target). For the **GST-k**, the input looks like the following:

```
<|sentiment|>
<|sem|> [neutralized sentence]
<|orig|> [original sentence]
```

For the **GST-s**:

```
<|sem|> [neutralized sentence]
<|sen|> [sentiment sentence]
<|orig|> [original sentence]
```

Tokens `<|sem|>`, `<|sen|>`, and `<|orig|>` serve as special tokens for the GPT tokenizer, and the sentiment sentence (in **GST-s**) is randomly sampled from the sentences belonging to the target sentiment. We calculate the cross-entropy loss only for the words generated after the `<|orig|>` token.

4 Experiments

4.1 Data

The bulk of our dataset comes from the *Emotions* dataset provided by Saravia et al. (2018), containing 16,000 curated tweets labeled with six different classes (*sadness*, *joy*, *love*, *anger*, *fear*, and *surprise*). Since the labels are not distributed evenly (e.g., 5,362 sentences labeled as *joy*, but only 572 as *surprise*), we further add sentences from underrepresented categories from the *GoEmotions* dataset (Demszky et al. (2020)), featuring reddit posts labeled with 27 sentiments. Our final dataset features 18,334 sentences labeled into six categories. In addition, in the training loop we oversample minority classes so that each class is represented with ca. 5,300 examples. Oversampling turns out to be a crucial step with regard to the fine-tuning phase; otherwise, the model learns to transfer all input sentences into the majority sentiment. Our validation and test datasets contain 2,000 examples each.

4.2 Evaluation method

The goals of style transfer system are 1) preservation of the nonstylistic parts of the source sentence, 2) transfer strength of the stylistic features to the target style, and 3) naturalness (fluency and correct grammar) of the generated sentence (Mir et al. (2019)). Style transfer models should be compared across all three aspects to properly characterize differences. For instance, a model preserves content

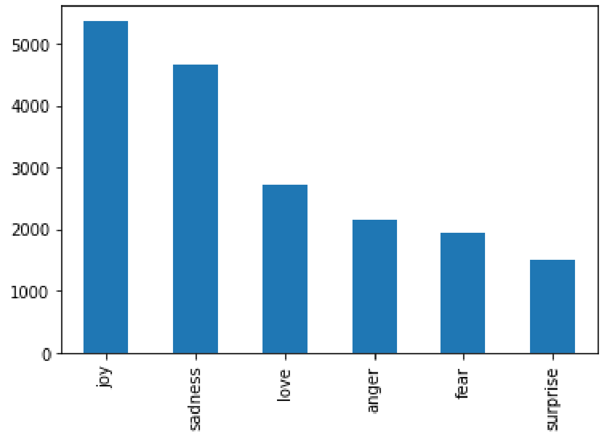

Figure 2: Training dataset statistics before oversampling

poorly if it alters content such as place names or verbs, even though it renders the final sentence in the correct sentiment. Given the preliminary nature of this exploration and the fact that our approach is essentially different from existing solutions, we do not compare our method with currently available binary transfer models. For example, a transfer from *negative* to *positive* sentiment is not the same as a transfer from *sadness* to *joy*, as the **GST** model is sensitized to different kinds of inputs. *Surprise* is an emotional category that can be both *positive* and *negative*, etc. However, to evaluate GST's performance, we compare the methods outlined above with each other.

To estimate the target style strength, we use the sentiment classifier trained on the original training-dev-test split. For every output sentence, we calculate the cosine similarity between its context vector representation (produced by the classifier) and the context vector representation of the target sentiment. The latter is an average of context vector representations of all sentences belonging to the target sentiment in the dataset. The higher the similarity, the better the sentiment has been preserved. To measure semantic content preservation, we use a publicly available Sentence-BERT transformer which encodes each sentence into a context vector. We then calculate the cosine similarity between the vector representation of the original sentence and the vector representation of the generated sentence. To measure fluency, we calculate the GLEU and BLEU scores between the generated and source sentences. We use these metrics to evaluate the two approaches outlined above as well as an additional baseline, which consists in a simple swapping of emotionally salient words (detected by the self-attention classifier) between the source and target sentences.

4.3 Experimental details

For the neutralization module, we use the learning rate of 1e-3, dropout probability of 0.2, and the hidden size of 300. The keys and values in the self-attention module are 768-dimensional vectors. To accelerate the training of the classifier, we use pre-trained 300-dimensional GloVe embeddings. For the emotionalization module, we use a multi-layer 'decoder-only' Transformer based on the Generative Pre-trained Transformer (GPT) of Radford et al. (2018). We choose the publicly available pre-trained OpenAI GPT-2 English model with 12 decoder layers, embedding size of 768 and 12 self-attention heads, with a total of 117M parameters. Our initial learning rate is 2e-5 and we use a cosine training schedule which decreases the learning rate following the values of the cosine function between the initial learning rate of the AdamW optimizer and 0. We continue the training until the validation loss starts growing, which indicates overfitting of the model to the training dataset. At test time, we beam search using softmax probabilities with a beam width of 5. The output beam (out of the top 5 final beams) that obtains the highest target-style match according to the sentiment classifier is chosen as the output sentence.

Model	**BLEU**	**GLEU**	**sem**	**sen**	**All**
swap	0.84	0.84	0.83	0.90	0.85
GST-keyword	0.84	0.84	0.85	**0.91**	0.86
GST-sentence	**0.85**	**0.86**	**0.87**	0.89	**0.87**

Table 2: Models evaluation. *sem* - semantic similarity, *sen* - sentiment similarity

4.4 Results

The differences in performance between the three methods are minimal. The relatively high BLEU and GLEU scores (above 0.8) indicate that the reconstruction is accurate across the board. The swapping method which simply replaces the words in the original sentence with the words from the sentiment sentence achieves a remarkable high accuracy, which is mainly due to the fact that much of the original sentence remains intact, and the model does not need to generate any new words. Moreover, since at test time the two sentences between which the words are swapped belong to the same sentiment category, the sentiment context is also well preserved. On the other hand, while offering similar and often slightly higher accuracy, the GST approach provides us with something that the simple swapping mechanism cannot achieve - the generalized sentiment transfer between potentially unlimited kinds of sentiment.

5 Analysis

One of the main challenges in training the models for sentiment transfer consists in accurate neutralization of sentences. If too many words are removed, the model might not be able to fully reconstruct the semantic content of the original sentence. On the other hand, if too few words are removed and some of the sentiment information remains in the input sentence, we can end up with contradictory (but not ambiguous) results (e.g., *The food was terrible and the restaurant was surprisingly clean*).

The learning process is very informative. The model first learns to simply repeat the sentence that contains any emotional words, for example:

```
<|sem|> the food was and the restaurant was
<|sen|> i feel great today
<|orig|> i feel great today
```

Then, the model learns to merge the two sentences:

```
<|sem|> the food was and the restaurant was
<|sen|> i feel great today
<|orig|> i was feeling very generous and the food was delicious
```

Eventually, the model learns to generate a sentence focused on the semantic content only:

```
<|sem|> the food was and the restaurant was
<|sen|> i feel great today
<|orig|> the food was delicious and the restaurant was very
welcoming
```

Another interesting ability of the model is to generate emotionally ambiguous sentences. If the neutralized sentence still "suggests" certain sentiment (e.g., *joy*), but we provide a contrary sentiment signal (e.g., *sadness*), the model generates an emotionally ambiguous sentence, for example:

```
<|sadness|>
<|sem|> this was a movie and i want to watch it again
<|orig|> i want to watch this again and again but i was feeling
 a little homesick
```

Probably the most important finding is that the fine-tuned model is not limited to the six sentiments that we started the training with. Thus, if we provide `<|confusion|>` as the sentiment keyword in **GST-k**, which is a signal the model has never seen before, the model produces a "confused" sentence:

```
<|confusion|>
<|sem|> i thought i knew what is going now but now i feel very
and
<|orig|> i thought i knew what is going now but now i feel very
 confused and doubtful
```

A common problem of the transformer models is the repetition of content during text generation. Our model is not an exception, and the model frequently produces repetitive sentences:

```
<|orig|> i feel very satisfied and satisfied
```

Such unwanted results can be mitigated through repetition penalties and by widening the beam search.

6 Conclusion

In this paper, we introduce a new set of problems called Generalized Sentiment Trasnfer and explore a simple training technique that allows us to transfer semantic content between potentially unlimited number of sentiments. The results are promising, since we used one of the smallest pre-trained GPT2 models available; models with a larger number of parameters should further improve the outputs. An area awaiting further research is the neutralization module, as the removal of words is way too simple a technique to completely eliminate emotive potential from human language. Certain purely "semantic" turns of phrase, when combined together, still contain traces of sentiment and suggest emotive complements; a clear-cut separation between semantics and sentiment might not be possible.

References

D. Demszky, D. Movshovitz-Attias, J. Ko, A. S. Cowen, G. Nemade, and S. Ravi. Goemotions: A dataset of fine-grained emotions. *CoRR*, abs/2005.00547, 2020. URL https://arxiv.org/abs/2005.00547.

Z. Fu, X. Tan, N. Peng, D. Zhao, and R. Yan. Style transfer in text: Exploration and evaluation, 2017.

Z. Hu, R. K.-W. Lee, C. C. Aggarwal, and A. Zhang. Text style transfer: A review and experimental evaluation. *arXiv preprint arXiv:2010.12742*, 2020.

J. Li, R. Jia, H. He, and P. Liang. 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, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1169. URL https://aclanthology.org/N18-1169.

F. Luo, P. Li, P. Yang, J. Zhou, Y. Tan, B. Chang, Z. Sui, and X. Sun. Towards fine-grained text sentiment transfer. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2013–2022, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1194. URL https://aclanthology.org/P19-1194.

R. Mir, B. Felbo, N. Obradovich, and I. Rahwan. Evaluating style transfer for text. *ArXiv*, abs/1904.02295, 2019.

A. Mousa and B. Schuller. Contextual bidirectional long short-term memory recurrent neural network language models: A generative approach to sentiment analysis. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 1023–1032, Valencia, Spain, Apr. 2017. Association for Computational Linguistics. URL https://aclanthology.org/E17-1096.

M. Nussbaum. *Love's Knowledge: Essays on Philosophy and Literature*. Oxford paperbacks. Oxford University Press, USA, 1990. ISBN 9780195074857. URL https://books.google.com/books?id=oq3POR8FhtgC.

A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever. Improving language understanding by generative pre-training. 2018.

E. Saravia, H.-C. T. Liu, Y.-H. Huang, J. Wu, and Y.-S. Chen. 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, Oct.-Nov. 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1404. URL `https://aclanthology.org/D18-1404`.

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

H. Sklar. *The Art of Sympathy in Fiction: Forms of Ethical and Emotional Persuasion*. Linguistic Approaches to Literature. John Benjamins Publishing Company, 2013. ISBN 9789027233509. URL `https://books.google.com/books?id=vbFvnET3jJkC`.

A. Sudhakar, B. Upadhyay, and A. Maheswaran. Transforming delete, retrieve, generate approach for controlled text style transfer. *arXiv preprint arXiv:1908.09368*, 2019.

H. Sun, Y. Huang, and S. Lu. Improving fine-grained text sentiment transfer for diverse review generation. In *2020 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA)*, pages 261–266, 2020. doi: 10.1109/ICAICA50127.2020.9182678.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

J. Xu, X. Sun, Q. Zeng, X. Ren, X. Zhang, H. Wang, and W. Li. Unpaired sentiment-to-sentiment translation: A cycled reinforcement learning approach, 2018.

L. Zunshine and O. S. U. Press. *Why We Read Fiction: Theory of Mind and the Novel*. Theory and interpretation of narrative series. Ohio State University Press, 2006. ISBN 9780814210284. URL `https://books.google.com/books?id=BtdB2CcXazEC`.