Text-Style-Transfer with CycleGAN: Decoder-only Models and Italian Datasets

Alessandro Arneodo s331361
Daniele Ercole s328755
Davide Fassio s323584
Alessia Manni s331377

Politecnico di Torino

Introduction

Definition of Text Style Transfer

TST involves modifying the stylistic attributes of text (e.g., formal vs. informal, poetic vs. prosaic) while preserving its semantic content.

Key Challenge

Achieving style transformation without parallel (aligned) datasets for training.

Approach

- Adapt CycleGAN, originally used for images, to TST
- Its self-supervised framework is ideal for situations where paired examples are not available.
- Model Structure: 2 generators and 2 discriminators work in a cycle-consistent manner to transform text styles and then recover the original content, with the help of a style classifier.

Introduction

Proposed extensions and experiments:

1. **Salesforce's CTRL**
   Employing a conditional language model to control and generate text in the desired style.

2. **GPT-2 Integration**
   Using GPT-2 (a decoder-only model) as the generator within the CycleGAN framework to potentially improve generation quality.

3. **Verse-to-Prose Transfer**
   Applying the method to convert verse (poetic form) into prose, with the "Divina Commedia" as a case study.

4. **Extension to Italian**
   Testing the original model on Italian datasets, including an Italian version of the GYAFC dataset from the XFORMAL collection.

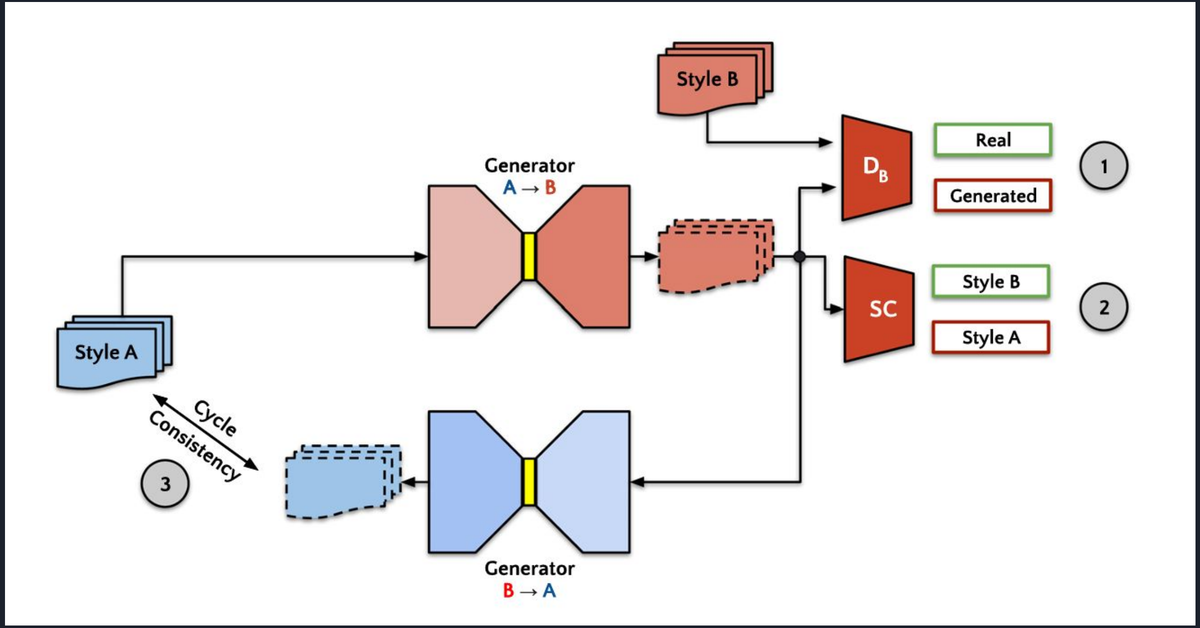
Methodology – CycleGAN for TST

Key Idea: use a cycle-consistent adversarial network to transform text from one style to another and back.

Components:

1. **Generators ($G_{A\to B}$, $G_{B\to A}$):** transform text from style A to B and vice versa.
2. **Discriminators (D_A, D_B):** distinguish between real text and text generated in each style.
3. **Style classifier:** evaluates and enforces the stylistic consistency between the generated text and the target style, contributing to the style loss component.

Methodology – CycleGAN for TST



Methodology – CycleGAN for TST

The **Total Loss** is a weighted sum of several losses to balance style transfer:

- **Adversarial Loss**
  Ensures the generated text is indistinguishable from real text in the target style.
- **Cycle-Consistency Loss**
  Ensures that translating a text to the opposite style and back recovers the original text (preserving content).
- **Style Loss**
  Encourages the generated text to align with the stylistic features of the target domain.
- **Discriminator Loss**
  Penalizes the discriminators when they fail to correctly identify real versus generated text.

Extension 1 – TST with Salesforce CTRL

Objective: evaluate Salesforce's CTRL model for zero-shot text style transfer on the Yelp dataset.

What is CTRL and how it works

- Is a conditional language model that enables controlled text generation.
- Uses predefined control codes to condition the output, guiding the model in generating text with the desired style.
- To transform sentiment, we use specific prompts:

Opinion Negative: *<input sentence>* Positive:

Opinion Positive: *<input sentence>* Negative:

Extension 1 – TST with Salesforce CTRL

Generation Constraints and Custom Token

- **Output Length Constraint**
  Limit the generated text to a maximum of **150%** of the input sentence length.
- **Custom End-of-Sequence Token**
  Introduce a special token (ID **246533**) to allow early termination of the output, ensuring concise generation.

Yelp Dataset

- **Source:** restaurant reviews.
- **Test Set:** contains 4 human-generated references per sentence.

Extension 1 – Results

CTRL significantly underperforms compared to CycleGAN-based models.

- **Fluency vs. Style Control:** CTRL struggles to balance fluency with effective style control.
- **Critical Issue:** empty output generated in 59% of instances.
- **Implication:** these low scores suggest major challenges in producing coherent and stylistically appropriate text.

RESULTS ON THE YELP DATASET

	ref-B avg	acc_{BERT}	GM	HM
CycleGAN BART (base)	55.7	**78.8**	**66.3**	**65.3**
CycleGAN BART (large)	**56.5**	75.1	65.1	64.5
CycleGAN T5 (small)	53.0	78.0	64.3	63.1
CycleGAN T5 (base)	54.2	76.6	64.4	63.5
CycleGAN T5 (large)	55.3	72.9	63.5	62.9
CTRL (Ours)	0.308	54.8	4.11	0.612

Extension 1 – Results

EXAMPLES FROM YELP TEST SET – NEGATIVE TO POSITIVE – SUCCESS CASES

Source	you are not the only place in town with decent pizza .
Generated	* You are the only place to get a good pizza
References	decent pizza they had here it is the only place with excellent pizza you are the only place in town with decent pizza . this is the best place in town for pizza
Source	when she did n't answer him quickly enough , he hung up on her .
Generated	when she did answer him , he did n't hang up on her .
References	when she did n't answer him quickly enough , he patiently waited on her . when she answered him quickly , he did n't hung up on her . she answered quickly when she did n't answer him quickly enough , he hung up on her .

Extension 1 – Results

EXAMPLES FROM YELP TEST SET – POSITIVE TO NEGATIVE – FAILURE CASES

Source	this resort has been in scottsdale forever and it truly knows service .
Generated	* the location is so close to the ocean that you can feel the waves crashing against the shore .
References	this resort has been in scottsdale forever and it truly does n't know service . this resort has been in scottsdale forever and it truly does not knows service this resort has been in scottsdale forever and it does not knows service at all . despite this resort having been in scottsdale for forever they do n't know a single thing about service
Source	try anything on the menu and you will be glad you came in !
Generated	- no choice of wine
References	i would not try anything on their menu try anything on the menu and you will be sad you came in try anything here and you will regret coming here try anything on the menu and you will be glad you never came in !

Extension 2 – Replacing the Generator with GPT-2

Objective: replace the original encoder-decoder generator with a GPT-2 Instruct model (vicgalle/gpt2-open-instruct-v1), a decoder-only architecture.

Model Replacement

- Replace *AutoModelForSeq2SeqLM* with *AutoModelForCausalLM* for causal language modeling.

Tokenizer Configuration

- Set *padding_side = "left"* for proper padding.
- Use the end-of-sequence (EOS) token if no pad token is available.
- Different EOS Tokens:
  a. *openai-community/gpt2:* Uses **<|endoftext|>** (ID 50256)
  b. *vicgalle/gpt2-open-instruct-v1*: Uses **"### End"** (ID 50257)

Extension 2 – Replacing the Generator with GPT-2

Specialized Prompt Template

Each input is embedded within a prompt with the following structure:

> Below is an instruction that describes a task. Write a response that appropriately completes the request.
>
> **### Instruction:** Transform the following sentence from informal style to formal style:
>
> *<input sentence>*
>
> **### Response:**

Extension 2 – Replacing the Generator with GPT-2

Teacher Forcing & Loss Computation during training

- **Prompt Tokenization**
  Tokenize the prompt without truncation to capture its full length.
- **Target Sentence Tokenization**
  Tokenize with truncation to fit within *max_seq_length - prompt_length.*
- **Loss Computation**
  - Concatenate prompt and target tokens.
  - Mask out prompt tokens (set to -100) so that only target tokens contribute to the loss.

Extension 2 – Replacing the Generator with GPT-2

Inference and Generation Process

- **Inference Steps**
  - Tokenize (and pad) only the prompt.
  - Pass the prompt to the model for generation.
- **Post-Processing**
  Remove the initial prompt from the output to extract the final response.

The **Discriminator** remains a transformer-based classifier.

❌ Due to limited computational resources, we were unable to complete the training of the GPT-2 Instruct generator variant.

Extension 3 – Verse-to-Prose Transfer

Objective: transform verse from Dante's *Divina Commedia* into prose using the CycleGAN framework.

Dataset Composition

- **Source Text:** tercets from Dante's Divina Commedia (original verse).
- **Target Text:** corresponding prose interpretations
- **Data Splits:** Train (75%), Validation (15%) and Test (10%)
- **References:** only one reference is available for both modern Italian and Dantean styles due to the difficulty of obtaining multiple alternative versions.

Extension 3 – Verse-to-Prose Transfer

Challenge: original pretrained models were designed for English.

Solution: substitute with Italian-pretrained models to handle the language-specific nuances.

Two Main Phases:

1. **Style Classifier Training**
   Fine-tuned Italian BERT model: *dbmdz/bert-base-italian-cased*

2. **CycleGAN Model Training**
   - **Generator:** utilizes the *morenolq/bart-it* generator, adapted for Italian.
   - **Discriminator:** uses the same fine-tuned Italian BERT model (*dbmdz/bert-base-italian-cased*) as a style discriminator.

Extension 3 – Results

Evaluation metrics indicate that the CycleGAN model has not learned to modify the input as intended.

Key Metrics

- **Self-BLEU & Self-ROUGE:** extremely high scores, implying outputs are nearly identical to inputs.
- **BERTScore & Reference-based BLEU/ROUGE:** minimal differences between the input and generated text.

KEY EVALUATION METRICS FOR STYLE TRANSFER WITH *Divina commedia* DATASET

Metric	**A → B**	**B → A**
BERTScore	0.7415	0.7493
g-BLEU	22.5463	24.8201
ref-BLEU	6.1775	7.3397
ref-ROUGE-1	0.3474	0.3679
ref-ROUGE-2	0.0982	0.1128
ref-ROUGE-L	0.3033	0.3233
self-BLEU	82.2886	83.9319
self-ROUGE-1	0.9976	0.9958
self-ROUGE-2	0.9937	0.9872
self-ROUGE-L	0.9976	0.9958
style F1 score	0.00429	
style accuracy	0.00431	
style precision	0.00427	
style recall	0.00431	

Extension 3 – Results

Identity Mapping

- In both transfer directions, outputs are virtually identical to the inputs.
- This suggests that the model is performing an identity mapping rather than a true style transformation.

Challenges in the Task

- **Complexity beyond simple Style Transfer:** the task resembles a translation task due to the need for paraphrasing old Italian text.
- **Additional demands:** requires careful rephrasing of archaic expressions without distorting the original meaning.

Extension 4 – XFORMAL Dataset

Objective: extend the work by incorporating the XFORMAL dataset, a multilingual version of the GYAFC dataset.

XFORMAL Dataset

- Provides corresponding translations of the GYAFC dataset in several languages (e.g., French, Portuguese, Italian).
- Same train/validation/test splits as in the original dataset.
- Contains 4 human-generated references per sentence.
- We focused our experiment on the Italian translations from the Family & Relationships category.

Extension 4 – Results

- The model failed to perform effective style transfer.
- Instead of altering the input, the system reproduced it almost unchanged.
- The model's performance remains insufficient, even with a simpler, language-specific dataset. A simpler approach using only Italian data did not resolve the style transfer issue.

KEY EVALUATION METRICS FOR STYLE TRANSFER WITH *XFORMAL* DATASET

Metric	**A → B**	**B → A**
BERTScore	0.8699	0.8315
g-BLEU	9.4268	8.9585
ref-BLEU	1.3297	1.2173
ref-ROUGE-1	0.7734	0.6068
ref-ROUGE-2	0.6299	0.4150
ref-ROUGE-L	0.7589	0.5755
self-BLEU	66.8277	65.9283
self-ROUGE-1	0.9944	0.9977
self-ROUGE-2	0.9933	0.9975
self-ROUGE-L	0.9944	0.9977
style F1 score	0.2588	
style accuracy	0.2702	
style precision	0.2594	
style recall	0.2583	

Final Takeaways & Future Directions

Challenges in Generalization

- While CycleGAN has shown promise in prior research, its effectiveness has not been proven to be easily adaptable to new architectures, datasets, and complex style shifts.

Insights for Future Research

- This study provides insights into modern NLP architectures for style transfer.
- Future improvements should focus on refining CycleGAN for multilingual and complex style adaptation.

THANK YOU FOR THE ATTENTION!

Politecnico di Torino