DEEP LEARNING FOR NLP: PROJECT REPORT

IntentCONANv2 Intent-Specific Counterspeech Generation

Kartik Bansiwal (2021EE30743)[1] and Siddhant Raj (2021MT10932)[1]

[1]Department of Electrical Engineering, IIT Delhi

May 10, 2025

Abstract

The growing challenge of online hate speech demands effective, scalable countermeasures that balance free expression and harm reduction. In this project, we tackle the task of intent-specific counterspeech generation using the IntentCONANv2 dataset. By fine-tuning T5 and OPT-1.3B models with LoRA adapters, we generate fluent and rhetorically controlled responses to hate speech. Our system conditions on both hate speech and a specific intent — informative, denouncing, questioning, or positive — to guide the response style. The resulting model achieves high performance across automatic metrics and demonstrates practical value for use in content moderation and civic discourse support tools.

Keywords: counterspeech, hate speech, LoRA, transformers, intent generation, NLP moderation

1 Problem Statement

The objective of this project is to build an intent-specific counterspeech generation model using the IntentCONANv2 dataset, which contains approximately 13,000 counterspeech examples. Each counterspeech instance in the dataset is associated with one of four specific intents: informative, denouncing, questioning, or positive. The task involves generating high-quality counterspeech responses that are not only relevant to the given hate speech input but also aligned with the specified counterspeech intent. This enables the creation of purpose-driven, context-aware counterspeech that enhances the effectiveness of online dialogue moderation systems.

2 Definitions

- **Hate Speech:** Language that attacks or demeans a group based on race, religion, ethnicity, gender, sexual orientation, or other identity markers.
- **Intent:** The rhetorical tone or communicative goal of a counterspeech, such as informing or denouncing.
- **Counterspeech:** A direct, constructive response to hate speech aiming to refute or de-escalate harmful content.
- **T5:** A text-to-text transformer that reformulates NLP tasks as generation problems.
- **OPT:** A causal decoder-only transformer model trained to generate language given a prompt.
- **LoRA:** Low-Rank Adaptation, an efficient fine-tuning method enabling training of large models with few parameters.
- **Instruction Tuning:** A technique where models are trained using natural language instructions to improve generalization and controllability in generation tasks.
- **Prompt Engineering:** The practice of carefully designing input prompts to guide language models toward producing desired outputs more effectively.

3 Dataset and Preprocessing

Dataset: We used the IntentCONANv2 dataset, which consists of approximately 13,000 entries. Each entry contains:

- Hate speech input

- Counterspeech output
- An associated intent label

Preprocessing Steps:

1. Concatenated the intent label with the hate speech as input.
2. Used the counterspeech as target output.
3. Tokenized using T5Tokenizer (T5) or BPE Tokenizer (OPT).
4. Applied truncation and padding to 512 tokens (input) and 128 tokens (target).
5. Loaded data using HuggingFace datasets from CSVs (train, validation, test).

4 Methodology

4.1 Problem Formulation

The objective is to generate intent-specific counterspeech conditioned on a given hate speech input and its associated rhetorical intent. The four supported intents are:

- Informative
- Denouncing
- Questioning
- Positive

Each sample is transformed into a prompt-based format, e.g., `<intent_question> Why do immigrants take our jobs?`. The model must generate a relevant, tone-aware counterspeech in response.

4.2 Model Architecture

We fine-tuned two base models:

- **T5-Large:** Encoder-decoder transformer that maps input to target using a text-to-text format
- **OPT-1.3B:** A decoder-only causal language model used for prompt-based generation.

Both models were fine-tuned using LoRA adapters. In T5, LoRA was applied to the query and value projection layers in the attention modules. In OPT, adapters were applied to `q_proj` and `v_proj` in decoder blocks.

Note: OPT-1.3B is a private model on Hugging Face and requires authentication via a Hugging Face API key. Users must accept the license agreement on the model card and use the API key to load it via `transformers`.

4.3 Training Strategy

Training was performed using HuggingFace's `Trainer` and `Seq2SeqTrainer`. Key aspects include:

- **Loss Function:** Cross-entropy loss with optional label smoothing ($\epsilon = 0.1$)
- **Optimizer:** AdamW with warm-up and linear decay
- **Decoding:** Beam search with 4 beams and early stopping
- **Evaluation Metrics:** BLEU (fluency), ROUGE (n-gram overlap), BERTScore (semantic similarity)

4.4 Instruction Tuning and Prompt Engineering

We incorporated instruction tuning to guide the model using structured prompts. Two distinct formats were explored:

- **Intent Token Format:** e.g., `<intent_informative> Why do immigrants take our jobs?`
- **Natural Language Instructions:** e.g.,

```
Intent: Informative
Hate: Why do immigrants take our jobs?
Response: ...
```

We observed that T5 performed better with the special token format, while OPT responded better to natural language prompts. This instruction tuning helped the models align their outputs more closely with the expected rhetorical tone.

4.5 Optimization Enhancements

We adopted LoRA (Low-Rank Adaptation) to improve training efficiency. Instead of full model updates, LoRA introduces trainable low-rank matrices in attention layers, reducing parameter count and memory usage. This allowed faster training, lower risk of overfitting, and more scalable experiments. Inference was further optimized via:

- Batched generation with GPU acceleration
- Beam size and length penalty tuning
- Prompt formatting tailored to model architecture

4.6 Implementation Summary

The training pipeline was modularized using object-oriented wrappers:

- `LoRAOPT`: Encapsulates OPT with LoRA-based attention adaptation
- `CSTrainer`: Custom trainer using HuggingFace `Trainer` APIs with logging, checkpointing, and gradient accumulation
- `CSEvaluator`: Generates and evaluates responses using HuggingFace pipelines
- `IntentTokenizer` and `PatchedData`: Handled tokenization, prompt alignment, padding, and caching

This setup enabled clean experimentation with different model types, prompt formats, and training configurations, facilitating repeatable and scalable NLP research on counterspeech generation.

5 Training Configuration

T5-Large Fine-Tuning Details

- **Model:** T5-Large (770M parameters)
- **LoRA Adapters:** Applied to query and value projections ($r = 8$, $\alpha = 32$)
- **Dataset:** Training set ($\approx$10K samples)
- **Batch Size:** 8 (with gradient accumulation)
- **Epochs:** 3 base + 5 LoRA epochs
- **Learning Rate:** 5e-5 (base), 3e-5 (LoRA)
- **Loss:** Cross-entropy with label smoothing $\epsilon = 0.1$
- **Optimizer:** AdamW with linear warm-up
- **Decoding:** Beam search with 4 beams (max length 128)
- **Device:** Google Colab (NVIDIA T4)
- **Training Time:** 30 minutes (LoRA phase)

OPT-1.3B Fine-Tuning Details

- **Model:** OPT-1.3B (1.3B parameters) — private model requiring Hugging Face API key
- **LoRA Adapters:** Applied to q_proj and v_proj ($r = 8$, $\alpha = 16$)
- **Dataset:** Training set
- **Batch Size:** 2 (gradient accumulation = 2 $\rightarrow$ effective size = 4)
- **Epochs:** 3
- **Learning Rate:** 2e-4
- **Input Length:** Max 256 tokens
- **Practices:** Gradient clipping (norm=1.0), early stopping, mixed-precision

6 Evaluation

- **BLEU-:** Measures n-gram overlap.
- **ROUGE:** Longest common subsequence similarity.
- **BERTScore:** Cosine similarity of contextual embeddings.

Metric	T5-Large	OPT-1.3B
BLEU	21.6%	23.4%
ROUGE	8.1%	8.7%
BERTScore (F1)	0.8702	0.8810

Table 1: Evaluation results on IntentCONANv2 test set.

7 Results and Discussion

OPT-1.3B consistently outperformed T5-Large across all evaluation metrics. The decoder-only architecture of OPT is inherently better suited for continuation-based generation tasks, particularly when the prompt includes structured prefixes like intent and hate speech. This structure allows OPT to condition on the input with greater fluency and coherence.

T5-Large, in contrast, provides stronger performance in more structured generation settings, and its encoder-decoder architecture is advantageous when tasks require deeper understanding and transformation of input semantics. However, T5 required more careful hyperparameter tuning and intent-token alignment to maintain rhetorical consistency in generation.

Instruction Tuning and Prompt Engineering: Both models benefitted significantly from intent-conditioned prompts, but their architectural nature influenced how they interpreted instructions.

This prompt format enabled OPT to treat intent and hate speech as part of a structured dialogue input, making its generation more fluent and responsive to context. For T5, we used special tokens like `<intent_informative>` prepended to the input. T5 required cleaner formatting and tighter control to generate stylistically aligned outputs.

Fine-tuning Behavior and LoRA Efficacy: LoRA played a crucial role in enabling scalable training. T5 without LoRA exhibited signs of early overfitting, especially on smaller intent categories like "questioning." With LoRA, training was more stable, memory-efficient, and generalizable. For OPT, full fine-tuning was infeasible due to memory constraints, and LoRA was essential for adapting the large model to task-specific nuances.

Key Observations:

- **Fluency:** OPT-1.3B generates smoother, more naturally flowing counterspeech. Its strong pretraining on web-scale corpora gives it stylistic diversity and rich lexical representations.
- **Semantic Control:** T5 performs better in generating precise, intent-controlled counterspeech. It handles smaller domains and structured response types more effectively.
- **Overfitting:** Without LoRA, T5 tended to memorize intent-response pairs, especially when the training set was imbalanced. LoRA fine-tuning introduced regularization benefits.
- **Prompt Sensitivity:** OPT was more sensitive to subtle changes in instruction phrasing (e.g., line breaks, colon placement), which required multiple rounds of prompt refinement.
- **Inference Speed:** OPT had faster generation in batch mode using the HF pipeline API. However, it also consumed more memory.
- **Deployment Feasibility:** T5 is open, lighter, and easy to run on academic GPUs. OPT requires authenticated access to HuggingFace (if using 1.3B or higher) and more compute.

Model Selection Insights:

- Choose **OPT-1.3B** if the focus is on high-quality, fluent language generation in deployment environments with access to powerful GPUs (e.g., A100).
- Prefer **T5-Large** when interpretability, reproducibility, and lower resource usage are priorities, especially in research or education contexts.
- LoRA should be used by default when adapting large models, regardless of architecture. It offers a favorable trade-off between speed, memory, and performance.

Future Enhancements: In future iterations, instruction-tuned models like FLAN-T5 or OpenChat variants could be evaluated. Moreover, integrating emotion conditioning, real-time toxicity filtering, or multi-intent generation could further improve the system's versatility. We also envision a human-in-the-loop setup where generated counterspeech is reviewed or voted on, blending automation with social responsibility.

8 Applications

- **Online Moderation:** Real-time counterspeech in forums or comment sections.
- **Civic Education:** Teaching respectful rebuttal in schools or training.
- **Chatbots:** Intent-aware dialogue agents for de-escalation.
- **Policy Support:** Assisting moderators with automatic response suggestions.

9 Conclusion

We built and evaluated intent-specific counterspeech generation models using T5 and OPT. OPT-1.3B yielded better fluency and semantic alignment, while T5 ensured tighter stylistic control. LoRA adapters allowed training at scale without full model updates.

Our approach can be used in real-world moderation tools and civic discourse agents. Future work could explore emotion-conditioning, adversarial robustness, and user feedback-based refinement.

Acknowledgements

We thank Prof. Tanmoy Chakraborty for guidance, and HuggingFace, Google Colab for compute support.

References

[1] Raffel, C. et al. "Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer." 2020.

[2] Hu, E. J. et al. "LoRA: Low-Rank Adaptation of Large Language Models." arXiv:2106.09685, 2021.

[3] Zhang et al. "OPT: Open Pre-trained Transformer Language Models." Meta AI, 2022.