RSSN at Multilingual Counterspeech Generation: Leveraging Lightweight Transformers for Efficient and Context-Aware Counter-Narrative Generation

Ravindran V
Department of Computer Science and Engineering
Sri Sivasubramaniya Nadar College of Engineering
ravindran2213003@ssn.edu.in

Abstract

This paper presents our system for generating counter-speech (CN) in response to hate speech (HS), developed for the COLING 2025 shared task. We employ lightweight transformer-based models, DistilBART and T5-small, optimized for computational efficiency while maintaining competitive performance. Through comprehensive dataset analysis, we identify linguistic patterns, explore challenges, and propose enhancements. Our findings demonstrate the viability of lightweight models and highlight error patterns that can guide future research directions. We provide a detailed evaluation of results across multiple metrics, including BLEU, ROUGE, and BERTScore, and discuss strategies for enhancing contextual relevance in CNs.

1 Introduction

The pervasive spread of hate speech (HS) across online platforms poses a significant challenge to fostering respectful and inclusive digital discourse. Counter-speech (CN) has emerged as a proactive and constructive strategy to address hate speech, offering alternative narratives that challenge biases while promoting empathy and inclusivity. However, the generation of effective counter-speech demands solutions that balance contextual relevance, linguistic coherence, and adaptability to diverse scenarios, making it a complex yet critical task.

This paper presents our system developed for the COLING 2025 shared task, designed to generate contextually appropriate counter-narratives efficiently. Our approach leverages lightweight transformer-based architectures, specifically **DistilBART** and **T5-small**, to achieve a balance between computational efficiency and performance. By utilizing these compact models, we aim to demonstrate that high-quality counter-narrative generation can be achieved without the computational overhead typically associated with larger architectures.

To inform and optimize the generation process, we conducted a thorough analysis of the multilingual dataset provided for the shared task, with a specific focus on the English subset. This choice was motivated by the need to maintain consistency across training, validation, and evaluation phases while ensuring robust and interpretable results. Our methodology incorporates structured preprocessing, integrating key components such as the target group (TARGET), hate speech instance (HS), and background knowledge (KN) to create inputs that preserve contextual richness.

This study underscores the viability of lightweight transformer models in generating high-quality counter-narratives, offering a scalable solution for addressing hate speech in resource-constrained scenarios. Through detailed evaluation and contextualized comparisons, we demonstrate the potential of these models for impactful and efficient counter-speech generation

2 Related Work

The development of counter-narratives (CNs) as a strategy to combat hate speech has been the focus of several research efforts. Early work highlights the challenges in evaluating CNs, as traditional metrics like BLEU and ROUGE often fail to align with human judgment. To address this, frameworks leveraging large language models (Jones et al., 2024) (LLMs) for multi-aspect evaluation have emerged, providing interpretable and human-aligned assessments based on guidelines from counter-narrative specialized organizations. Knowledge-grounded CN generation (Chung et al., 2021) has also gained attention, with approaches

Proceedings of the First Workshop on Multilingual Counterspeech Generation (MCG), pages 13–18
January 19th, 2025.

integrating external repositories to produce contextually rich and factually accurate CNs, addressing issues of generic or repetitive outputs. Additionally, comparative studies of pre-trained language models (Tekiroglu et al., 2022) have identified autoregressive models with stochastic decoding as particularly effective for CN generation. These studies emphasize the importance of task-specific training data and the use of post-editing pipelines to enhance quality and adaptability, particularly in addressing unseen hate targets. Together, these advancements contribute significantly to the development of robust and context-aware CN generation systems.

3 Dataset Analysis

The dataset, LanD-FBK ML_MTCONAN_KN, comprises training (1,584 samples), validation (400 samples), and test (400 samples) splits across four languages: English (EN), Spanish (ES), Italian (IT), and Basque (EU). Each entry includes a hate speech instance (HS), corresponding counter-narrative (CN), background knowledge sentences (KN), target group (TARGET), and language (LANG). The dataset covers multiple targets of hate: Jews, LGBT+, Migrants, Muslims, People of Color (POC), and Women.

3.1 Language and Target Group Distribution

The dataset is balanced across the four languages (English, Spanish, Italian, and Basque), ensuring fair representation for multilingual evaluation. The target groups include Women, Migrants, LGBT+, Jews, and POC, with Women being the most frequently targeted group.

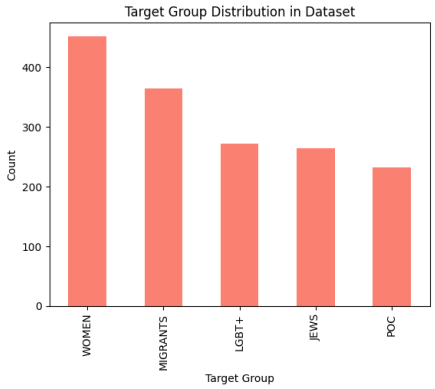

Figure 1: Target group distribution in the dataset.

3.2 Text Length Analysis

Table 1 summarizes the length statistics for HS and CN across languages. CNs are significantly longer, reflecting the complexity of respectful counter-narratives.

3.3 Heatmap of Target Groups across Languages

Figure 2 provides a heatmap of target group distribution across languages, highlighting uniform distribution across the dataset.

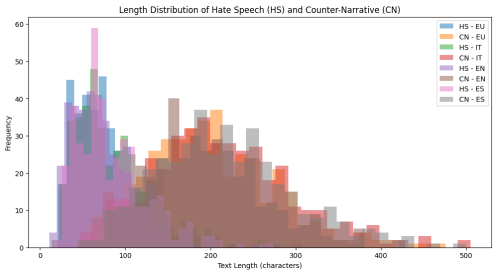

Figure 2: Distribution of target groups across languages.

4 System Architecture

4.1 Approach 1: T5-Small

The T5-Small model, a compact variant of the Text-to-Text Transfer Transformer (T5) architecture, was selected to balance computational efficiency and performance. The T5 framework reformulates all NLP tasks into a unified text-to-text format, making it an ideal choice for generating counter-speech by leveraging structured inputs and text-based reasoning.

4.1.1 Data Preprocessing

The preprocessing pipeline begins by filtering the dataset to include only English-language samples, ensuring consistency across the training, validation, and test splits. Each data point is reformatted into a structured prompt that integrates hate speech (HS), the corresponding target group (TARGET), and the available background knowledge (KN).

The input prompt is constructed as:

Prompt = "Generate respectful counterspeech. TARGET: TARGET HS: HS KN: KN"

This format encapsulates all contextual components to guide the model in producing nuanced counter-narratives (CNs). Empty strings were substituted for missing KN fields, and extraneous tokens like ‘<EOS>‘ were removed to ensure

Language	HS Mean	HS Std	CN Mean	CN Std	HS Max	CN Max
EN	74.5	40.1	191.4	69.5	275	432
ES	84.6	42.6	216.5	79.9	274	500
EU	81.2	41.7	200.2	74.1	307	476
IT	84.8	43.7	214.6	79.7	282	505

Table 1: Text Length Statistics for Hate Speech (HS) and Counter-Narratives (CN).

uniformity in inputs.

For example, an entry with **TARGET** as *Migrant*, **HS** as *"Go back to your country"*, and **KN** as *"Migrants contribute positively to the economy"* is formatted as:

" Generate respectful counterspeech.
TARGET: Migrant
HS: Go back to your country
KN: Migrants contribute positively to the economy "

This explicit representation ensures that the contextual and target information is preserved, which helps the model generate responses tailored to the input.

4.1.2 Tokenization and Training

The tokenization process was performed using the T5 tokenizer, which converts input prompts and target CNs into tokenized sequences. Specific details include:

- Inputs were tokenized to a maximum length of 128 tokens, balancing the inclusion of essential context (HS, TARGET, KN) with computational efficiency. This limit ensures most HS instances are fully captured while leaving space for additional contextual fields.
- Target CNs were tokenized to a maximum of 64 tokens to encourage concise, focused counter-narratives suitable for actionable responses.

During tokenization, padding and truncation were applied to maintain uniform input lengths for batching. The tokenized outputs included:

- **input_ids:** The tokenized representation of the input prompt.
- **attention_mask:** Masks differentiating actual tokens from padding tokens.
- **labels:** The tokenized representation of the target CN.

The training configuration involved the following hyperparameters:

- A learning rate of 3×10^{-5} was chosen to ensure stable gradient updates and gradual convergence.
- Batch sizes of 8 were used for both training and evaluation, balancing memory constraints and computational efficiency.
- Training was conducted for 5 epochs, with the AdamW optimizer employed for regularization.

4.1.3 Generation and Inference

During inference, test samples were preprocessed and tokenized in the same manner as the training data. The model generated counter-narratives using beam search decoding, which explores multiple paths to identify optimal outputs. Key parameters used during decoding include:

- **Beam Size:** Set to 5, enabling exploration of diverse generation paths.
- **Maximum Length:** The output sequences were constrained to 50 tokens to ensure concise yet contextually rich responses.
- **Sampling Strategies:** Techniques like top-k sampling (with $k = 50$) and nucleus sampling ($p = 0.95$) were employed to introduce variability while maintaining relevance.

To ensure human-readable outputs, all special tokens were removed during post-processing.

4.1.4 Error Handling and Validation

The following strategies were implemented to handle errors and validate outputs:

- The dataset was analyzed to verify the completeness and informativeness of the HS and KN fields. No entries were found to have empty or insufficiently informative KN fields.
- Failed or incomplete outputs were identified and reprocessed to maintain response quality.
- The final submission file was validated to ensure compliance with the shared task's formatting guidelines.

4.1.5 Submission Preparation

The generated predictions were compiled into a CSV file adhering to the shared task guidelines. The fields included:

- **ID:** A unique identifier for each test example.
- **KN:** Left empty as per submission requirements.
- **KN_CN:** The generated counter-narrative corresponding to the input HS.

4.2 Approach 2: DistilBART

DistilBART, a distilled version of BART (Lewis, 2019), is designed to retain the powerful sequence-to-sequence generation capabilities of BART while significantly reducing the computational requirements. This approach leverages the compact architecture of DistilBART to efficiently generate counter-narratives (CNs) without compromising quality.

4.2.1 Dataset Preparation

The dataset preparation process began with filtering the test split of the LanD-FBK/ML_MTCONAN_KN dataset to include only English-language samples. This decision was driven by several considerations. First, focusing on English allowed us to leverage pre-trained transformer models like T5-Small and DistilBART, which are optimized for English text and offer robust performance for text-to-text generation tasks. Second, processing a single language reduced computational overhead, enabling more efficient training and evaluation. While multilingual approaches are viable, they require additional preprocessing and fine-tuning steps to handle linguistic diversity.

Challenges Addressed:

- **Consistency in IDs:** The concatenation of PAIR_ID and LANG ensured that each entry had a unique identifier across the dataset.
- **Filtering Non-English Entries:** Explicit filtering of non-English entries streamlined the focus on relevant data and eliminated potential noise in the test data.

4.2.2 Prediction Integration and Postprocessing

The predictions generated by DistilBART were formatted into a CSV file containing only the counter-narratives. To ensure consistency with the T5-Small model, we used the same training and evaluation parameters for DistilBART. Postprocessing was required to prepare the final submission file, ensuring it adhered to the shared task's format. This process involved:

1. Verifying that the number of rows in the filtered test split matched the rows in the predictions file to ensure consistency.
2. Adding the ID field, derived from the test split, to the predictions.
3. Introducing a blank KN column, as required by the submission guidelines.
4. Reordering columns to the specified format: ID, KN, and KN_CN.

Implementation Highlights:

- The verification step served as a critical checkpoint to detect mismatches between the dataset and predictions, reducing the risk of submission errors.
- The systematic addition of required columns (ID, KN) ensured compliance with submission rules.

4.2.3 Challenges and Resolutions

Data Mismatch: A potential mismatch between the test split and the predictions was identified as a critical risk. This was mitigated through an explicit validation step that compared the row counts of the dataset and predictions.

Submission Compliance: Ensuring compliance with the submission format required a systematic approach to reordering columns and validating the output. This structured process minimized the likelihood of formatting errors.

4.2.4 Conclusion on DistilBART

DistilBART demonstrated the feasibility of using compact, distilled architectures for generating high-quality counter-narratives in a computationally efficient manner.

5 Evaluation and Results

5.1 Evaluation Metrics

The generated counter-narratives (CNs) were evaluated using a combination of ranking-based and reference-based metrics:

- **JudgeLM (Zubiaga et al., 2024)**: A large language model-based ranking system, evaluating CNs for quality and alignment with human judgment.
- **BLEU**: Assesses lexical overlap by comparing n-grams between predictions and references.
- **ROUGE-L**: Measures structural similarity through the longest common subsequence between predictions and references.
- **BERTScore (Zhang et al., 2019)**: Computes semantic similarity using contextual embeddings.
- **Novelty**: Captures creativity by identifying unique n-grams in the generated text not present in training data.
- **Gen_len**: Reports the average length of generated CNs to assess verbosity.

This evaluation framework enabled a balanced assessment of fluency, creativity, and contextual relevance in counter-speech generation.

5.2 Final Rankings and Performance Metrics

Our submissions, **RSSN Run 1** and **RSSN Run 2**, were evaluated using JudgeLM alongside reference-based metrics, including ROUGE-L, BLEU, and BERTScore. The final results and rankings are presented in Table 2.

5.3 Analysis of Results

RSSN Run 1: Achieved a higher overall rank due to its superior performance across most metrics, particularly in *ROUGE-L* (46.3%), *BLEU* (35.7%), and *BERTScore* (78.8%). This indicates strong contextual relevance and fluency, making it well-suited for long-form counter-narrative generation.

RSSN Run 2: While ranked lower overall, this run demonstrated higher novelty (*80.8%*), highlighting its capability to generate diverse and creative outputs. However, its lower performance in *BLEU* (13.2%) and *BERTScore* (69.2%) suggests areas for improvement in maintaining contextual coherence and factual accuracy.

5.4 Insights from Rankings

The comparative evaluation highlights the complementary strengths of the two runs:

- **Run 1:** Optimized for contextually relevant, fluent, and coherent counter-narratives.
- **Run 2:** Emphasized novelty and diversity, making it suitable for applications requiring unique responses.

These findings reinforce the potential for hybrid approaches that combine the contextual strength of Run 1 with the creative diversity of Run 2, enabling a balanced solution for counter-speech generation tasks.

6 Error Analysis and Discussion

6.1 Error Patterns

Analysis of generated CNs revealed common issues:

- **Lack of Specificity:** CNs often lacked context-specific details, making them appear generic.
- **Repetition:** Certain CNs included repeated phrases, reducing their coherence and impact.
- **Context Misinterpretation:** Models occasionally failed to integrate the background knowledge (KN) effectively.

6.2 Comparative Analysis of Predictions

The comparison between Run 1 and Run 2 provides valuable insights into their respective strengths and weaknesses:

- **Textual Fluency:** Run 1 excels in generating fluent and cohesive narratives, making it suitable for detailed and engaging counter-narratives. Run 2, however, occasionally suffers from abrupt transitions or incomplete ideas.

Submission	JudgeLM Score	ROUGE-L (%)	BLEU (%)	BERTScore (%)	Gen_Len	Novelty (%)
RSSN Run 1	681.5	46.3	35.7	78.8	40.8	78.4
RSSN Run 2	59.0	24.5	13.2	69.2	31.0	80.8

Table 2: Final rankings and evaluation metrics for RSSN submissions.

- **Contextual Integration:** Run 1 naturally integrates contextual knowledge (KN) into responses, while Run 2 explicitly incorporates KN, often enhancing factual accuracy but at the expense of fluency.
- **Handling of Hate Speech:** Run 1 adopts a constructive and empathetic tone, whereas Run 2 uses a direct rebuttal style, offering clarity but lacking nuanced approaches in some cases.
- **Content Length:** Run 1 produces longer, more detailed responses suitable for in-depth discussions, whereas Run 2 provides concise outputs ideal for short-form applications.
- **Application Suitability:** Run 1 is well-suited for long-form content like essays or blogs, while Run 2 is more effective for short-form platforms such as social media.

These findings suggest the potential for a hybrid approach, combining the fluency and contextual integration of Run 1 with the factual accuracy and conciseness of Run 2, to optimize counter-speech generation across diverse applications.

7 Conclusion and Future Work

This paper demonstrates the effectiveness of lightweight transformer models, such as T5-Small and DistilBART, for counter-speech generation. While our approach achieved competitive performance across multiple metrics, limitations in contextual specificity and diversity were identified.

Future work will focus on:

- Enhancing preprocessing techniques to better handle complex input contexts.
- Incorporating external knowledge sources to enrich the CN generation process.
- Exploring ensemble approaches and advanced decoding strategies to improve robustness.
- Evaluating the system on unseen datasets to assess scalability and generalization.

Our findings emphasize the importance of balancing computational efficiency with output quality, paving the way for further advancements in counter-speech generation systems.

References

Yi-Ling Chung, Serra Sinem Tekiroglu, and Marco Guerini. 2021. Towards knowledge-grounded counter narrative generation for hate speech. *arXiv preprint arXiv:2106.11783*.

Jaylen Jones, Lingbo Mo, Eric Fosler-Lussier, and Huan Sun. 2024. A multi-aspect framework for counter narrative evaluation using large language models. *arXiv preprint arXiv:2402.11676*.

M Lewis. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Serra Sinem Tekiroglu, Helena Bonaldi, Margherita Fanton, and Marco Guerini. 2022. Using pre-trained language models for producing counter narratives against hate speech: a comparative study. *arXiv preprint arXiv:2204.01440*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Irune Zubiaga, Aitor Soroa, and Rodrigo Agerri. 2024. A llm-based ranking method for the evaluation of automatic counter-narrative generation. *arXiv preprint arXiv:2406.15227*.