Hidetsune at SemEval-2026 Task 10: A Systematic Exploration of Training and Inference Strategies for Detecting Conspiracy Beliefs

Hidetsune Takahashi
Waseda University
takahashi78h@toki.waseda.jp

Abstract

This paper describes a system developed for SemEval-2026 Task 10 Subtask 2, which focuses on identifying conspiracy beliefs expressed in Reddit comments. The study begins with a comparative analysis of language models fine-tuned on the task data. In addition to fine-tuning, multiple auxiliary techniques were examined, including instruction-based prompting, data augmentation via back-translation, and a loss function designed to address label imbalance. In the final stage, the inference behavior was further examined by varying the decision threshold applied to the softmax output probabilities. The results highlight how choices made during model selection, training, and inference collectively affect performance, offering empirical insights into the challenges of conspiracy belief detection in social media contexts.

1 Introduction

This paper investigates SemEval-2026 Task 10 Subtask 2 (Samory et al., 2026), which addresses the binary classification of Reddit comments with respect to whether they express conspiracy beliefs. This task lies at the intersection of psychology and computational linguistics. With the continued expansion of social media platforms, conspiracy beliefs increasingly appear in online discussions, creating a growing need for natural language processing (NLP) approaches to analyze and detect such content (Enders et al., 2023).

Specifically, this work first examines several open-source language models fine-tuned on the provided dataset using undersampling. In addition, a range of experimental techniques including instruction tuning, back-translation, class-weighted cross-entropy loss, and focal loss are explored. Building upon these experiments, softmax-based decision threshold adjustment is further investigated to analyze how model behavior varies under different decision settings.

The results highlight performance differences across model choices and experimental configurations. While the examined experimental techniques did not consistently improve overall performance, a fine-tuned model combined with softmax-based decision threshold adjustment achieved a weighted F1 score of 81.8% during the development phase, whereas performance degradation was observed during the test phase. Through these findings, this work aims to clarify both the strengths and limitations of the adopted methodologies and to provide insights into their applicability for practical text classification tasks related to conspiracy beliefs.

2 Background

Conspiracy theories have been widely studied in the fields of social and political discourse, and both their causes and effects have received increasing attention in recent years (Douglas et al., 2019). Prior research suggests that the factors driving conspiracy beliefs are closely related to psychological processes, particularly epistemic, existential, and social motives (Douglas et al., 2017). Such beliefs manifest in various aspects of social life, including heated political contests, international conflicts, and the COVID-19 pandemic (Douglas, 2021).

Conspiracy beliefs are also prevalent on social media platforms (Cinelli et al., 2022), highlighting the growing importance of NLP techniques in this domain. For example, Moffitt et al. (2021) applied large language models (LLMs) to identify COVID-19–related conspiracy narratives. Similarly, Haupt et al. (2023) adopted a hybrid approach combining NLP and content coding to analyze conspiracy discourse surrounding 5G wireless technology, employing methods such as topic modeling and sentiment analysis. In addition, prior work has explored the detection of conspiracy propagators using NLP-based methods, including word embeddings coupled with convolutional neural network

Proceedings of the The 20th International Workshop on Semantic Evaluation (2026), pages 176–181

architectures (Giachanou et al., 2023).

Building upon these prior studies, SemEval-2026 Task 10 Subtask 2 addresses a binary classification problem that aims to determine whether a given Reddit comment expresses a conspiracy belief (Samory et al., 2026). The data provided for this task are derived from real Reddit discussions and are distributed across three splits: training, development, and test sets.

The datasets are released indirectly via hydration codes and provided in JSONL format, containing fields such as `text`, `id`, `subreddit`, `markers`, `annotator`, and `conspiracy`. The `text` field stores the Reddit comment text, while the `conspiracy` field contains the corresponding annotation label. Each instance is annotated with one of three labels: *Yes*, *No*, or *Can't tell*. In the training set, these labels consist of 1,541, 1,990, and 785 samples, respectively.

Participants are instructed to use the training set for system development and experimentation, the development set for submissions during the development phase, and the test set for final evaluation. System performance is evaluated using weighted F1 score, accuracy, and class-wise F1 scores for the *Yes* and *No* labels, with weighted F1 serving as the primary evaluation metric.

3 System Description

This study first examines several open-source models through fine-tuning on the provided training dataset. In addition, experimental techniques such as back-translation and focal loss are evaluated. Finally, a softmax-based decision threshold adjustment is applied using the best-performing configuration obtained from the preceding experiments. The code is available on GitHub[1].

4 Experimental Setup

4.1 Comparison of Models

In the primary stage of the experiments, multiple models were compared to identify which could effectively contribute to the performance of the task. Specifically, three encoder-based Transformer models including BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019), and TwHIN-BERT (Zhang et al., 2022) were investigated, each with two variants. In addition, two variants of GPT-2 (Radford et al., 2019) were evaluated to examine the differences between decoder-based and encoder-based Transformer architectures. This selection of baseline models is consistent with previous work, in which fine-tuned Transformer models achieved reliable results across multiple languages even with relatively straightforward data processing (Takahashi et al., 2024).

For each model and its variants, appropriate hyperparameters such as training batch size and the number of warm-up steps were carefully selected. Early stopping was also applied to reduce the risk of overfitting. To further address data-related issues, undersampling was employed to mitigate the class imbalance between the *Yes* and *No* labels. In addition, 20% of the training data was held out as a validation set during the fine-tuning process. Table 1 summarizes model performance and the corresponding hyperparameters.

As shown in Table 1, BERT-large-uncased and RoBERTa-large achieved relatively strong performance, whereas the other variants of these models, as well as TwHIN-BERT, yielded lower scores. Although the lack of explicit imbalance-handling at this stage may have influenced the results, these findings suggest that models with a larger number of tunable parameters might be more suitable for this task.

Furthermore, both variants of GPT-2 underperformed compared to the encoder-based models. This outcome is expected, as encoder architectures are designed to capture and understand input representations, whereas decoder architectures are primarily optimized for text generation.

Based on these results, RoBERTa-large was selected as the base model for fine-tuning and subsequent experiments, owing to its relatively superior performance and previously reported improvements over BERT-base (Liu et al., 2019), despite sharing a similar underlying architecture.

4.2 Experimental Techniques

Several experimental techniques were explored to improve the classification performance of the chosen model. They include data augmentation, prompt-based architectural modifications, and a loss function designed to handle class imbalance.

4.2.1 Instruction Tuning

Instruction tuning was explored to examine whether the model could benefit from explicit textual instructions. Specifically, the following prompt was appended to the input during both training and

[1] https://github.com/Hidetsune/SemEval2026_Task10.git

Model	Learning rate	Epoch	Batch size	Weight decay	Warmup steps	Weighted F1	Accuracy
BERT-base	2e−5	2	16	0.01	56	0.707	0.701
BERT-large	1e−5	5	8	0.01	113	0.792	0.792
RoBERTa-base	2e−5	2	16	0.01	56	0.707	0.701
RoBERTa-large	1e−5	4	8	0.01	113	0.790	0.792
TwHIN-BERT-base	2e−5	4	16	0.01	56	0.720	0.714
TwHIN-BERT-large	1e−5	1	8	0.01	113	0.732	0.727
GPT-2	5e−6	4	8	0.01	113	0.615	0.662
GPT-2-medium	3e−6	1	4	0.01	225	0.607	0.597

Table 1: Comparison of model performance

inference:

```
Question: Does the following text
express conspiracy belief?
Answer with yes (1) or no (0).

Text: {text}
```

where `{text}` denotes the original input text.

4.2.2 Back-Translation

Back-translation was employed to augment the data for underrepresented labels. This technique could be effective because it generates additional sentence patterns without changing the original logical meaning, which gives the model more varied expressions to learn from (Edunov et al., 2018). Such variation may help the model become less dependent on a limited set of surface forms, particularly when the available training data are sparse or unevenly distributed. This expectation is also partly supported by previous work, where data augmentation and class balancing showed some potential in multilingual classification settings (Takahashi et al., 2025).

Specifically, instances with the Yes label in the conspiracy category were randomly sampled such that the combined number of original and augmented instances matched that of the No label. The sampled texts were first translated into Spanish using a neural machine translation model (Tiedemann and Thottingal, 2020) and then back-translated into English using a corresponding model (Tiedemann and Thottingal, 2020). This process increases the number of samples for minority labels while preserving the original semantic content, although the surface forms of the sentences may differ.

4.2.3 Class-weighted Cross-entropy Loss

To address class imbalance at the level of the loss function, a class-weighted cross-entropy loss was employed. Generally, the cross-entropy loss $L_{\mathrm{CE}}(\theta)$ is defined as

$$L_{\mathrm{CE}}(\theta) = -\sum_{k} t_k \log y_\theta(k \mid x) \tag{1}$$

where θ denotes the model weight vector, x the input, y_θ the probability distribution output by the model parameterized by θ, and t the ground-truth probability distribution.

Here, let N_k denote the number of training samples belonging to class k, and let

$$w_k = \frac{N}{N_k} \tag{2}$$

where $N = \sum_k N_k$ is the total number of samples.

Using these class weights, the cross-entropy loss in Eq. (1) is modified as

$$L_{\mathrm{WCE}}(\theta) = -\sum_{k} w_k \, t_k \log y_\theta(k \mid x) \tag{3}$$

where $L_{\mathrm{WCE}}(\theta)$ denotes the weighted cross-entropy loss.

According to Eq. (2), labels with limited training instances are assigned higher class weights w_k, strengthening their impact on weighted cross-entropy loss L_{WCE}. This reweighting approach reduces the dominance of majority classes and facilitates a more even distribution of learning signals under imbalanced data conditions.

4.2.4 Focal Loss

Using the cross-entropy loss defined in Eq. (1), the predicted probability assigned to the correct class can be expressed as

$$p_y = \exp(-L_{\mathrm{CE}}(\theta)) \tag{4}$$

since the cross-entropy loss for a single instance reduces to $L_{\mathrm{CE}}(\theta) = -\log p_y$.

Based on p_y, a focal weighting term is applied to modulate the loss:

$$L_{\mathrm{FL}}(\theta) = (1 - p_y)^{\gamma} \; L_{\mathrm{CE}}(\theta) \tag{5}$$

Method	Weighted F1	Accuracy
Instruction tuning	0.757	0.753
Back-translation	0.771	0.766
Weighted CE	0.754	0.753
Focal loss	0.714	0.714

Table 2: Performance comparison across methods

Threshold	Weighted F1	Accuracy
0.30	0.794	0.792
0.35	0.806	0.805
0.40	0.818	0.818
0.45	0.804	0.805
0.50	0.790	0.792
0.60	0.776	0.779

Table 3: Performance comparison by thresholds (dev)

where $\gamma \geq 0$ is a focusing parameter that controls the degree to which well-classified samples are down-weighted and harder examples are emphasized during training. In this experiment, the focusing parameter was set to $\gamma = 2.0$.

The results of the four experimental techniques are summarized and compared in Table 2. Although all methods achieved weighted F1 scores above 0.70, each of them underperformed the simple fine-tuning of RoBERTa-large with undersampled data reported in Table 1, with performance drops ranging from approximately 2% to 8%, depending on the technique.

Regarding instruction tuning, the resulting performance was 0.757 in weighted F1 and 0.753 in accuracy, whereas the baseline fine-tuning of RoBERTa-large achieved a weighted F1 score of 0.792. This degradation is likely attributable to the structure of the input text. In particular, the self-attention mechanism may have allocated attention to the appended instructional tokens instead of focusing exclusively on the original text, leading to an undesirable dispersion of attention. Furthermore, as RoBERTa-large is an encoder-based model optimized primarily for representation learning rather than instruction following, the additional prompt may have functioned as noise rather than providing effective guidance.

Back-translation resulted in a decrease of approximately 2% in weighted F1. This may be due to subtle distortions in textual nuance introduced during the translation process, or because the back-translated sentences remained highly similar to the original texts and therefore provided limited diversity for fine-tuning. In addition, the original undersampling strategy did not discard a substantial amount of data, as the class imbalance was relatively moderate, with 1,541 instances labeled as *Yes* and 1,990 as *No*.

Similar considerations may explain the reduced effectiveness of class-weighted cross-entropy loss and focal loss. These loss-based techniques are generally more advantageous under conditions of severe class imbalance; however, the results suggest that, given the data distribution and task setting, the simpler undersampling approach was more suitable for this scenario.

4.3 Thresholding

In the final step of the experiments, probability thresholding was applied to the prediction outputs. Let K denote the number of classes, and let

$$\mathbf{z} = (z_1, z_2, \ldots, z_K) \in \mathbb{R}^K \tag{6}$$

be the output logits produced by the model. The softmax function is defined as

$$\text{softmax}(z_k) = \frac{\exp(z_k)}{\sum_{j=1}^{K} \exp(z_j)} \tag{7}$$

where $k \in \{1, \ldots, K\}$ is the index of the class. This function maps the real-valued logit vector $\mathbf{z}$ to a probability distribution over the K classes, where each probability lies in the range $[0, 1]$. By obtaining probability distributions on predicted labels, the RoBERTa-large model fine-tuned on the undersampled data was evaluated under several thresholds on the prediction side during the development phase.

As shown in Table 3, the model achieves its highest performance at a decision threshold of 0.4, with performance decreasing as the threshold is either increased or decreased from this value. This trend indicates a mild but consistent dependence on the prediction threshold, suggesting that 0.4 is a relatively suitable choice for this model under the development dataset. This configuration ranked 4th out of 45 teams in the development phase.

5 Results and Discussion

During the test phase, the system was evaluated on the test dataset provided by the organizers. For the final submission, RoBERTa-large fine-tuned on undersampled data was used, and inference was conducted under several decision-threshold settings.

Threshold	W-F1	Acc.	$F1_{Yes}$	$F1_{No}$
0.30	0.721	0.721	0.710	0.731
0.40	0.710	0.709	0.689	0.727
0.50	0.722	0.722	0.692	0.747
0.60	0.734	0.735	0.697	0.765
0.70	0.728	0.731	0.685	0.766

Table 4: Performance comparison by thresholds (test)

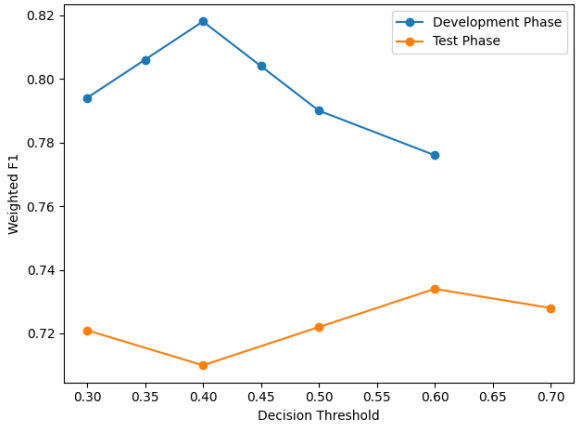

Figure 1: Scores against different decision thresholds

The results are shown in Table 4. The configuration with a decision threshold of 0.6 achieved a comparatively higher weighted F1 score and was therefore selected as the final submission. It ranked 31st out of 51 teams in the test phase, whereas the same method placed 4th out of 45 teams in the development phase.

Comparing the results across the two phases, it is probable that the characteristics of the datasets differ to some extent. Figure 1 illustrates the relationship between the decision threshold and the weighted F1 score for both the development and test datasets. While the development results exhibit a clear increase followed by a decrease around a peak value, the test results show a relatively weaker dependency on the threshold. In addition, the overall performance differs between the two phases, with the weighted F1 score in the test phase being approximately 0.1 lower than that in the development phase. These observations collectively suggest inherent differences between the two datasets, resulting in different threshold sensitivities and overall performance levels.

6 Limitations

One limitation of this study is the relatively limited analysis of the provided data. Although several experimental techniques, such as instruction tuning and class-weighted cross-entropy loss, were explored, limited emphasis was placed on fundamental data preprocessing steps. Consequently, basic yet potentially effective procedures, including data cleaning and lemmatization, as well as the use of supplementary data sources, might have contributed to further performance improvements, particularly when combined with loss-function-based approaches.

Another limitation is that this study focuses exclusively on Subtask 2 and treats it independently of Subtask 1, which involves identifying textual spans associated with specific types of conspiracy markers. Jointly addressing both subtasks and incorporating the extracted spans into model training could have enabled more effective utilization of our approach, for instance by assigning higher importance to span-level information during fine-tuning.

7 Conclusion

Throughout this study, language models were fine-tuned and compared to examine their fundamental capabilities. In addition, a range of experimental techniques, including instruction tuning and back-translation, were explored, followed by approaches related to the loss function such as class-weighted cross-entropy loss and focal loss.

Although the investigated experimental techniques did not consistently lead to improvements in overall performance, a fine-tuned model combined with a softmax-based decision threshold adjustment achieved a weighted F1 score exceeding 81.8% on the development dataset. In contrast, the performance on the test dataset was suboptimal and showed a reduced dependence on the decision thresholds, suggesting potential differences in data characteristics between the two phases. These observations raise the possibility that some of the examined techniques may become more effective in scenarios with a greater degree of class imbalance or different data distributions.

Future work may involve integrating more fundamental approaches, such as data preprocessing techniques, with the experimental settings explored in this study. In addition, leveraging textual span identification from related subtasks could further enhance the model's ability to capture conspiracy-related cues, thereby broadening the applicability of this approach to conspiracy belief detection and related tasks in psychological text analysis.

References

Matteo Cinelli, Gabriele Etta, Michele Avalle, Alessandro Quattrociocchi, Niccolò Di Marco, Carlo Valensise, Alessandro Galeazzi, and Walter Quattrociocchi. 2022. Conspiracy theories and social media platforms. *Current Opinion in Psychology*, 47:101407.

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.

Karen M Douglas. 2021. Are conspiracy theories harmless? *The Spanish journal of psychology*, 24:e13.

Karen M Douglas, Robbie M Sutton, and Aleksandra Cichocka. 2017. The psychology of conspiracy theories. *Current directions in psychological science*, 26(6):538–542.

Karen M Douglas, Joseph E Uscinski, Robbie M Sutton, Aleksandra Cichocka, Turkay Nefes, Chee Siang Ang, and Farzin Deravi. 2019. Understanding conspiracy theories. *Political psychology*, 40:3–35.

Sergey Edunov, Myle Ott, Michael Auli, and David Grangier. 2018. Understanding back-translation at scale. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 489–500, Brussels, Belgium. Association for Computational Linguistics.

Adam M Enders, Joseph E Uscinski, Michelle I Seelig, Casey A Klofstad, Stefan Wuchty, John R Funchion, Manohar N Murthi, Kamal Premaratne, and Justin Stoler. 2023. The relationship between social media use and beliefs in conspiracy theories and misinformation. *Political behavior*, 45(2):781–804.

Anastasia Giachanou, Bilal Ghanem, and Paolo Rosso. 2023. Detection of conspiracy propagators using psycho-linguistic characteristics. *Journal of Information Science*, 49(1):3–17.

Michael Robert Haupt, Michelle Chiu, Joseline Chang, Zoe Li, Raphael Cuomo, and Tim K Mackey. 2023. Detecting nuance in conspiracy discourse: Advancing methods in infodemiology and communication science with machine learning and qualitative content coding. *Plos one*, 18(12):e0295414.

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. *CoRR*, abs/1907.11692.

JD Moffitt, Catherine King, and Kathleen M Carley. 2021. Hunting conspiracy theories during the covid-19 pandemic. *Social Media+ Society*, 7(3):20563051211043212.

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

Mattia Samory, Felix Soldner, and Veronika Batzdorfer. 2026. SemEval-2026 task 10: PsyCoMark – psycholinguistic conspiracy marker extraction and detection. In *Proceedings of the 20th International Workshop on Semantic Evaluation (SemEval-2026)*.

Hidetsune Takahashi, Xingru Lu, Sean Ishijima, Deokgyu Seo, Yongju Kim, Sehoon Park, Min Song, Kathylene Marante, Keitaro-luke Iso, Hirotaka Tokura, and Emily Ohman. 2024. OZemi at SemEval-2024 task 1: A simplistic approach to textual relatedness evaluation using transformers and machine translation. In *Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024)*, pages 7–12, Mexico City, Mexico. Association for Computational Linguistics.

Hidetsune Takahashi, Sumiko Teng, Jina Lee, Wenxiao Hu, Rio Obe, Chuen Shin Yong, and Emily Ohman. 2025. OZemi at SemEval-2025 task 11: Multilingual emotion detection and intensity. In *Proceedings of the 19th International Workshop on Semantic Evaluation (SemEval-2025)*, pages 109–115, Vienna, Austria. Association for Computational Linguistics.

Jörg Tiedemann and Santhosh Thottingal. 2020. OPUS-MT – building open translation services for the world. In *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, pages 479–480, Lisboa, Portugal. European Association for Machine Translation.

Xinyang Zhang, Yury Malkov, Omar Florez, Serim Park, Brian McWilliams, Jiawei Han, and Ahmed El-Kishky. 2022. Twhin-bert: A socially-enriched pre-trained language model for multilingual tweet representations. *arXiv preprint arXiv:2209.07562*.