Fine-Tuning BERT-Based Model for Detecting Social Media Manipulation in Low-Resource Settings

Vladyslav Hromenko[1,*,†], Viktor Shevchenko[1,†]

[1] *Institute of Software Systems National Academy of Sciences of Ukraine, 40 Academician Glushkov Avenue, building 5, Kyiv, 03187, Ukraine*[1]

Abstract

In this study, the multi-label classification of manipulative techniques in Ukrainian and Russian social media texts is investigated using transformer-based language models. A dataset comprising approximately 3,700 training posts, annotated with 10 distinct manipulation techniques per post, characterises a low-resource, class-imbalanced setting. A pre-trained Ukrainian RoBERTa model ("youscan/ukr-roberta-base") is fine-tuned, and a range of performance enhancement strategies were evaluated, including advanced tokenization, countermeasures for class imbalance: loss weighting, different loss functions, data augmentation via back-translation, layer-wise learning rate decay for stable fine-tuning, and post-training threshold optimization for prediction calibration. Experimental results indicate that the optimal model achieves a macro-averaged F1 score of approximately 0.40 on the validation set — a marked improvement over the presented by dataset publishers baseline of approximately 0.24. Detailed per-technique analysis reveals that while frequently occurring techniques (e.g., Loaded Language, F1 ≈ 0.70) are reliably detected, performance declines for rarer techniques (e.g., Bandwagon and Straw Man, F1 < 0.25). Although data augmentation and rebalancing strategies modestly enhance recall for under-represented techniques, they also contribute to an increase in false positives. Common error patterns, such as the confusion between related techniques, are discussed along with the limitations imposed by the small dataset. These findings offer valuable insights into effective practices for multi-label classification in low-resource settings and present the first results on the automated detection of manipulation techniques in Ukrainian texts, with significant implications for disinformation monitoring.

Keywords

Social Media Analysis, Natural Language Processing, Transformer Fine-Tuning, MultiLabel Classification, Ukrainian

1. Introduction

The proliferation of propaganda and manipulative content on social media has spurred research into automatic detection of specific manipulation techniques used to mislead or influence readers. Given a text (e.g. a Telegram post), the task is to identify which rhetorical or stylistic manipulation techniques (if any) are present – e.g. Loaded Language, Whataboutism, Straw Man, etc. This is inherently a multi-label classification problem: a single post may employ multiple such techniques. Prior work on propaganda detection has mostly focused on English news articles [1]. Notably, the SemEval-2020 Task 11 defined an inventory of propaganda techniques and provided an English dataset for identifying them. Top-performing systems there leveraged pre-trained Transformer language models and ensemble methods. However, for Ukrainian – which has become a hotspot for information warfare – there has been little to no existing data or models for this task. The UNLP 2025 Shared Task [2] addressed this gap by releasing a dataset of Ukrainian social media posts annotated with 10 manipulation techniques, such as Appeal to Fear, FUD (Fear, Uncertainty, Doubt), Glittering Generalities, etc. The language setting is particularly challenging: some posts are

Workshop "Intelligent information technologies" UkrProg-IIT`2025 co-located with 15th International Scientific and Practical Programming Conference UkrPROG'2025, May 13-14, 2025, Kyiv, Ukraine

† These authors contributed equally.

gromvlad12@gmail.com (V. Hromenko); gii2014@ukr.net (V. Shevchenko)

0009-0001-5285-9912 (V. Hromenko); 0000-0002-9457-7454 (V. Shevchenko)

in Ukrainian and others in Russian (both written in Cyrillic), requiring models to handle multilingual inputs.

In this paper, an approach to the multi-label classification of manipulation techniques in Ukrainian/Russian dataset is presented. This small dataset (only 3,800 training examples) poses significant challenges of low data regime and severe class imbalance – some techniques appear in hundreds of posts while others in only a few dozen. These issues can cause standard fine-tuning of large language models to overfit or to predict only the majority classes. Therefore several strategies were explored to address these challenges:

- Fine-tune a pre-trained Ukrainian RoBERTa transformer (125M parameters) as our base model, taking advantage of prior language knowledge. Also, there were experiments with a multilingual model to handle Russian text.
- Employ imbalance mitigation techniques, including algorithm-level methods like class-weighted loss and other types of loss functions that focus learning on rare positive labels by down-weighting easy negatives.
- Data augmentation was utilized to expand the effective training set. In particular, paraphrases of minority-class examples were generated via back-translation, in which posts were translated to another language and then back to Ukrainian/Russian [3].
- To fine-tune the transformer on limited data without overfitting, a layer-wise learning rate decay (LLRD) was applied [4], training higher layers more aggressively than lower ones (thereby preserving general linguistic features in early layers).
- Although experiments with per-label threshold optimization were conducted—where decision thresholds on the sigmoid outputs were tuned on a validation set in an attempt to maximize macro-F1 — with grid search for best thresholds no significant improvements were observed overusing a fixed threshold of 0.5; threshold optimization was therefore not included in the final system.

An extensive set of experiments was conducted to quantify the impact of these techniques. Results are reported in terms of macro-F1 (the official metric), as well as micro-F1, precision, and recall. The best configuration achieved a macro-F1 of approximately 0.40, which—although modest in absolute terms—is notable given the small size of the training data and the difficulty of the task (a baseline with no special handling scored below 0.30 macro-F1). An analysis of the techniques revealed which were most and least accurately detected and examples of typical errors were provided.

Finally, we discuss the limitations of our approach and promising directions for future improvements, such as leveraging unlabeled data or multi-task learning. In summary, our contributions are:

1. an effective fine-tuning approach for multi-label propaganda technique classification in a low-resource setting (Ukrainian/Russian).
2. an empirical study of imbalance mitigation, data augmentation, and thresholding techniques for the multi-label classification of manipulation techniques task.
3. the reported results on the UNLP 2025 Shared Task data, provide a benchmark macro-F1 ≈ 0.40.

We hope that our findings will inform future work in low-resource multi-label text classification and aid the development of tools to automatically monitor manipulation in information warfare contexts.

2. Related Work

2.1. Propaganda and Manipulation Detection.

Propaganda and manipulation detection in text have become increasingly important research areas within Natural Language Processing (NLP), particularly in the context of political discourse,

fake news, and social media. Early approaches in this field primarily relied on hand-crafted linguistic features and traditional machine learning classifiers such as Support Vector Machines (SVM), Naive Bayes, and Decision Trees. These systems often utilize syntactic and stylistic features like part-of-speech tags, sentiment polarity, and rhetorical structure to identify persuasive or manipulative cues [5].

However, with the rise of deep learning and transformer-based language models, the paradigm has shifted significantly. Pre-trained models such as BERT [19] and RoBERTa have demonstrated superior performance in domain-specific classification tasks, outperforming traditional methods [6]. For instance, in the SemEval 2023 Task 5 on clickbait spoiler classification, fine-tuned RoBERTa models significantly outperformed classifiers trained on hand-crafted features, demonstrating the capability of transformers in manipulation-related text classification [7].

Ensemble-based approaches have also been explored to further enhance performance by combining multiple transformer models. For example, combining RoBERTa, Transformer-XL, and XGBoost has been shown to outperform individual models in sentiment and manipulation-related tasks such as tweet classification [8]. Similarly, BoostingBERT integrates multi-class boosting techniques with BERT and RoBERTa to address difficult classification instances in NLP tasks, achieving state-of-the-art performance in multiple benchmarks [9].

Other studies have addressed the application of domain-specific transformer models, such as BERTweet and Bio_ClinicalBERT, in tasks involving informal or health-related language manipulation. These domain-tuned models have shown improved results over general-purpose models, highlighting the importance of language context and corpus similarity in manipulation detection [10].

Furthermore, text classification tasks targeting misinformation and deception—closely tied to propaganda—have benefited from fusion-based transformer architectures. For instance, fusion models combining BERT, RoBERTa, and XLNet outperformed both traditional and earlier deep learning models in detecting prescription medication abuse on Twitter, a context where language is often manipulated to conceal intent [11].

In sum, while early systems for propaganda and manipulation detection in NLP leveraged manual feature engineering, recent advancements have firmly established transformer-based and ensemble methods as the new standard, significantly improving performance in detecting subtle and complex forms of linguistic manipulation across diverse domains.

2.2. Transformer Fine‑Tuning in Low-resource languages.

Fine-tuning pre-trained transformer models for classification tasks in the Ukrainian language is a growing area within natural language processing (NLP). This trend is driven by the need to adapt advanced methods to low-resource languages. Recent studies have focused on customizing multilingual and general-purpose transformer models specifically for Ukrainian language tasks through supervised fine-tuning.

A significant contribution includes fine-tuning the Gemma and Mistral large language models (LLMs) using Ukrainian datasets. This approach has shown considerable improvements in various classification and instruction-following tasks. Additionally, the development of the Ukrainian Knowledge and Instruction Dataset (UKID) has significantly expanded available resources for training and evaluating models for the Ukrainian language [12].

Transformer-based methods have also been effective in grammatical error classification tasks. A two-stage fine-tuning approach, using synthetic data first and subsequently gold-standard data, was successfully applied to multilingual models such as mT5 and smaller seq2seq transformers, resulting in a strong performance on grammatical classification tasks [13].

For stance and argument classification, researchers have utilized adapter-based fine-tuning on multilingual transformers. Coupled with few-shot learning, this method effectively handled classification tasks related to political discussions about Ukraine, demonstrating its effectiveness even with limited labelled data [14].

Question answering (QA) has similarly been approached as a classification problem for Ukrainian. A multilingual BERT model was fine-tuned on a translated dataset similar to SQuAD, effectively demonstrating the potential of transfer learning by accurately identifying answers within Ukrainian Wikipedia articles, even in the absence of native QA datasets [15].

Further foundational research has improved technical text preprocessing methods for the Ukrainian language. Techniques like Cyrillic normalization, abbreviation handling, and compound word segmentation have been developed to significantly enhance the quality of input data for transformer models in domain-specific tasks [16].

Overall, these research efforts illustrate that fine-tuning transformer models for Ukrainian classification tasks is becoming increasingly practical. Strategies like synthetic data generation, adapter tuning, and targeted linguistic preprocessing continue to contribute to improved model performance.

3. Data overview

The experiments were conducted on a dataset of Ukrainian text samples annotated for propaganda techniques, formulated as a multi-label classification task. Each text may employ one or more of 10 distinct propaganda techniques (e.g., loaded language, bandwagon, whataboutism, appeal to fear, straw man, etc.) [2], following a taxonomy similar to that used in propaganda detection tasks in news articles [1]. The dataset consists of 3,822 samples in total (after filtering), with an average of approximately 1.3 techniques labelled per sample. The class distribution is highly imbalanced, which represents a common challenge in multi-label datasets [17]. For instance, the most frequent label, loaded_language, appears in 1,973 samples (over 50% of all instances), whereas the rarest technique, straw_man, is found in only 138 samples (~3.6%). Such imbalance may bias models towards consistently predicting the majority class, resulting in neglect of minority classes. To mitigate this, stratified data splitting and customized loss functions are employed, as described below.

4. Model Development

4.1. Preparation

The dataset was split into training (80%) and validation (20%) subsets using a multilabel-stratified shuffle split, ensuring that each propaganda technique was represented proportionally across both subsets. This stratification prevented the exclusion of rare classes from the validation set. Minimal text preprocessing was performed, given that the pre-trained model is capable of handling raw text; only excessive whitespace and line breaks were removed. Emojis and other symbols present in the texts were intentionally retained, as these could convey sentiment or emphasis pertinent to specific propaganda techniques. Although the average text length was relatively short— typically 256 tokens — a maximum sequence length of 512 tokens was set to accommodate longer examples with maximum pre-trained model capacity. Labels for each sample were binarized into a 10-dimensional vector for model training.

4.2. Pre-trained language model

As base model pre-trained RoBERTa-base transformer model, specifically the publicly available youscan/ukr-roberta-base was chosen, originally trained on the Fill-Mask task, with an initial dataset consisting of 85 million lines of Ukrainian texts from Wikipedia, social networks and Ukrainian OSCAR deduplicated dataset [18]. The model architecture aligns with the RoBERTa base model [20], featuring 12 transformer layers, 768 hidden units per layer, 12 attention heads, and approximately 125 million parameters [18].

Initially, other pre-trained models were explored to compare performance, such as:

1. ukr-models/xlm-roberta-base-uk [21], a smaller version of XLM-RoBERTa [22], with only Ukrainian and some English embeddings left.
2. ukr-models/uk-summarizer [23], compressed mT5-base model [24], with Ukrainian and English left.
3. KoichiYasuoka/roberta-base-ukrainian [25], RoBERTa-based model, trained on UberText dataset [26].

However, it was observed that the multilingual model's tokenizer produced a significant number of <unk> (unknown) tokens for Ukrainian/Russian texts, typically due to emojis or informal vocabulary. In contrast, zero unknown tokens were generated by the monolingual Ukrainian RoBERTa model on the same dataset. Results are shown in Table 1

Table 1
Unknown token ratio

Model	Unknown token ratio
KoichiYasuoka/roberta-base-ukrainian	0.017
ukr-models/uk-summarizer	0.012
ukr-models/xlm-roberta-base-uk	0.013
youscan/ukr-roberta-base	0

The "unknown token ratio" was quantified for several models, resulting in values around 1–2% for models such as XLM-RoBERTa, whereas the selected monolingual model had a ratio of 0.0%. This result indicated vocabulary coverage by the youscan/ukr-roberta-base model, likely because its pre-training data closely matched the domain, including informal social media language. Consequently, and to maintain a relatively compact model size (125 million parameters), the monolingual RoBERTa-base model was chosen for all subsequent experiments.

Transfer learning was leveraged by adding a classification head to the pre-trained model, which consists of a feed-forward layer producing one logit per propaganda technique, with a sigmoid activation applied to each logit to obtain independent probabilities for each label. This configuration frames the task as 10 parallel binary classification problems, a standard approach in multi-label text classification.

4.3. Training details

The HuggingFace Trainer API was utilized to manage the training loops [27], with modifications introduced to incorporate various loss functions — namely, weighted Binary Cross-Entropy (BCE) Loss, Focal Loss [28], and unboundF1 Loss [29] — as detailed in the experiments section. A custom Trainer subclass was developed to implement weighted binary cross-entropy that redefines the loss computation by employing a Binary Cross-Entropy Loss with logits loss function, adjusted with a pre-computed weight tensor. Similarly, Focal Loss and unboundF1 Loss were implemented as separate modules and integrated into the training process [2]. During training, macro-F1 was adopted as the primary evaluation metric, reflecting the emphasis of the UNLP Shard Task competition on treating all techniques with equal importance, in contrast to micro-F1, which tends to be influenced by the majority class. Micro-F1 and per-class metrics were also recorded for further analytical purposes, although they were not used in the model selection process.

All layers of the RoBERTa model were fine-tuned on the training set. Training was conducted for up to 5 epochs, more epochs give worse results on validation performance. Evaluation of the

validation set was performed at the end of each epoch, and the best model—defined as the model with the highest validation macro-F1 — was saved. All experiments were carried out on a single NVIDIA 2060 RTX GPU using mixed precision (fp16) and train batch size from 8 to 12 to accelerate training and utilize all available memory (6 GB).

4.4. Experiments

4.4.1. Addressing class imbalance

To mitigate strong class imbalance, cost-sensitive learning and alternative loss functions were experimented with. Initially, class-weighted Binary Cross-Entropy was implemented, in which each label's positive examples were assigned a weight inversely proportional to the label's frequency. Specifically, for each technique label a weight was computed as the ratio of the number of negative samples to the number of positive samples, as shown in Equation 1.

$$w_i = \frac{N_{\text{neg},i}}{N_{\text{pos},i}} = \frac{\text{total\_samples} - \text{count}_i}{\text{count}_i} \tag{1}$$

where:

- $N_{\text{pos},i} = \text{count}_i$ is the number of positive examples for the i-th label,
- $N_{\text{neg},i} = \text{total\_samples} - \text{count}_i$ is the number of negative examples for the i-th label.

Using these weights, the weighted Binary Cross-Entropy Loss for a given sample was defined as shown in Equation 2

$$l_{\text{WBCE}}(y, \hat{y}) = -w(y \log(\hat{y}) + (1 - y) \log(1 - \hat{y})) \tag{2}$$

where:

- $y \in \{0,1\}$ denotes the true label,
- $\hat{y} \in [0,1]$ is the predicted probability (typically obtained via a sigmoid function),
- w is the weight assigned to the class, calculated by the Equation 1.

This computation resulted in large weights for rare classes (for example, the rarest class, straw_man, received a weight of approximately 26.7) and a weight near 1.0 for the most common class, loaded_language. These weights were supplied to the Binary Cross-Entropy Loss function as the positive weight parameter. The rationale was that errors in minority classes should incur a higher penalty, compelling the model to pay more attention to these classes. Such re-weighting is recognized as an effective strategy for handling imbalanced data in deep learning [30].

Subsequently, Focal Loss was applied, a loss function designed to down-weight easy negatives and focus training on harder, misclassified examples, originally proposed in computer vision tasks [28]. The Equation 3 for Focal Loss is shown below.

$$\begin{cases} l_{\text{FL}}(y, \hat{y}) = -\alpha\,(1 - p_t)^{\gamma}\,\log(p_t) \\ p_t = y\,\hat{y} + (1 - y)\,(1 - \hat{y}) \end{cases} \tag{3}$$

where:

- $y \in \{0, 1\}$ is the true label,
- $\hat{y} \in [0,1]$ is the predicted probability,
- α is a balancing factor,
- $\gamma \geq 0$ is the focusing parameter that attenuates the contribution of well-classified, high-confidence examples to the overall loss.

Different values of tunable parameter, γ were tested during trials, with γ = 3.0 proved to be the best in conducted experiments. The objective was for Focal Loss to prevent the model from being overwhelmed by the abundant negative instances present for rare classes.

At last, the UnboundedF1 Loss was implemented, according to the Equation 4, as a custom loss function to directly optimize macro-F1 in a differentiable manner.

$$\begin{cases} TP = y \cdot \hat{y} \\ FP = (1 - y) \cdot \hat{y} \\ FN = y \cdot (1 - \hat{y}) \\ F1 = \dfrac{2 \cdot \mathrm{TP}}{2 \cdot \mathrm{TP} + FP + FN + \epsilon} \end{cases} \tag{4}$$

$$L_{\mathrm{UnboundedF1}} = 1 - F1$$

where:

- $y \in \{0, 1\}$ is the true label,
- $\hat{y} \in [0,1]$ is the predicted probability,

While the true F1 score is non-differentiable due to its reliance on discrete predictions, this approach uses a smooth approximation by treating the sigmoid-activated outputs as continuous probabilities. For each class, it computes a rough surrogate of true positives, false positives, and false negatives, enabling the calculation of a differentiable proxy for the F1 score as the harmonic mean. This method is particularly useful in multi-label classification, where aligning the training objective more closely with the evaluation metric can improve performance on underrepresented classes. In prior literature, such approaches are often referred to as F1 surrogate losses or soft/unbounded F1 objectives. [29].

Finally, Table 2 summarizes the loss function comparison results. All runs in this comparison used the same base model (youscan/ukr-roberta-base) and hyperparameters (5s epochs, learning rate equals 2e-5, max token length 256) varying only the loss function.

Table 2
Different loss function comparison

Loss Function	Macro F1 on validation
BCE (no class weights)	0.25
Weighted BCE	0.38
Focal Loss (γ=3.0)	0.27
UnboundedF1 Loss	0.34

As shown in Table 2, A macro-F1 score of only 0.25 was achieved by the baseline model trained with standard binary cross-entropy, reflecting very poor performance on under-represented techniques. In this run, the majority class was nearly always predicted, and an F1 score close to 0 was obtained on the rare classes. When class weights were incorporated, Macro F1 was dramatically improved to 0.38, representing a substantial absolute gain of over 0.13. The weighted loss enabled the recovery of some of the minority classes, as predictions for these classes were increased, thereby improving recall. In contrast, the focal loss approach underperformed, given a macro-F1 of approximately 0.27, only slightly above the baseline. Additional experiments were conducted with γ set to 0.5 and 1.5, and no improvements were observed. It is suspected that further tuning of the focal loss hyperparameter, γ, might be necessary to achieve better results, given the sensitivity of γ [28]. The UnboundedF1 loss achieved a macro-F1 of 0.34, outperforming the plain baseline but not reaching the effectiveness of the simpler weighted binary cross-entropy. Although the custom loss explicitly attempted to optimize macro-F1, it may have been more

difficult to optimize or required more epochs. Studies on differentiable F1 losses have reported mixed results, sometimes necessitating careful calibration to outperform binary cross-entropy [29]. In this case, weighted binary cross-entropy proved to be the most reliable and effective approach for addressing class imbalance, following common practice for imbalanced data. Based on these results, weighted binary cross-entropy was adopted as the primary loss function for the remaining experiments.

4.4.2. Hyperparameters tuning

Firstly, The effect of sequence length was examined. The RoBERTa tokenizer is capable of truncating inputs that exceed a set maximum length, potentially resulting in the loss of important information. In initial experiments, a maximum length of 256 tokens was employed, which covered most of the data. It was found that approximately 16.46% of the samples exceeded 256 tokens, with the longest cases reaching around 1468 tokens. Consequently, the maximum sequence length was increased to 512 tokens (the maximum available length for the chosen model) and the model was re-trained with the weighted loss. A slight improvement in validation performance was observed, with macro-F1 increasing from approximately 0.38 to 0.396 and micro-F1 rising from roughly 0.47 to 0.474. These modest gains suggest that, for a few samples, retaining the full text rather than a truncated 256-token segment aided in the correct identification of additional manipulation techniques. A maximum length of 512 tokens was therefore adopted for subsequent experiments, as the increase in computational cost was manageable.

After the best base loss function was identified, training hyperparameters were further tuned to boost performance using detailed hyperparameters and architectural adjustments. Discriminative fine-tuning was applied by assigning distinct learning rates to different layers of the model; a lower learning rate of 1×10^{-5} was allocated to the pre-trained Transformer layers, while a higher learning rate of 1×10^{-3} was assigned to the newly added classifier layer. This strategy, inspired by fine-tuning practices such as those employed in ULMFiT and BERT adaptations [31], enabled the classifier to rapidly adapt to class-specific nuances while preventing excessive updates to the sensitive lower layers. In parallel, the weight decay was increased from 0.01 to 0.1 to impose stronger regularization on the model's weights, thereby aiming to reduce overfitting. With the weighted BCE and a maximum sequence length of 512 tokens in place, training was extended to 8 epochs, during which validation loss and macro-F1 scores were continuously monitored and early stopping was implemented if performance deteriorated. Under these settings, training gives the best validation macro-F1 of approximately 0.41 after 6 epochs, after which performance plateaued. For comparison, training the same model with a uniform learning rate of 2×10^{-5} for all layers over 7 epochs resulted in a macro-F1 of around 0.40, indicating that the layer-wise learning rate provided a measurable improvement.

4.4.3. Final Model

A final model was configured with weighted BCE, a maximum sequence length of 512 tokens, discriminative learning rates, and a weight decay of 0.1, and a macro-F1 of approximately 0.41 was achieved on the validation set, with the corresponding micro-F1 reaching around 0.48. The gap between micro and macro F1 is attributed to class imbalance, as micro-F1 is influenced more by the numerous easy negatives and the single frequent class, whereas macro-F1 is reduced by the poor performance on rarer classes [32]. This final model was also evaluated on the held-out test set, which exhibited a similar class distribution, and a macro-F1 of approximately 0.40 was obtained on the test data. Precision-recall breakouts and further analysis are provided in the next section. Overall, it is demonstrated by these results that, with appropriate handling of imbalance and careful tuning of hyperparameters, a relatively compact monolingual transformer is capable of achieving around 0.4 macro-F1 on this challenging multi-label task. For context, it should be noted that this performance is competitive with systems from related manipulation detection competitions; for instance, in a recent English-language propaganda detection task, macro-F1

scores in the range of 0.5–0.6 were achieved by the best systems through the use of significantly more training data [1].

4.4.4. Additional experiments

Back-translation augmentation was applied by translating training examples from rare classes into English language using Helsinki-NLP Opus MT translation models [33] and then back to Ukrainian, thereby creating paraphrased variants of the original text, which potentially may lead to better model results [34]. The five most under-represented classes (appeal_to_fear, bandwagon, straw_man, whataboutism, and fud) were augmented by roughly doubling their positive samples through back-translation, resulting in an augmented training set of slightly larger size. However, when the model was fine-tuned on this augmented data using the same weighted loss setup, no improvement was observed in the validation macro-F1; in one trial, a slight decrease to approximately 0.36 was noted. It is believed that the additional synthetic examples did not enhance the model's ability to discriminate those techniques. A possible explanation is that while the back-translated texts exhibited a range of vocabulary, they did not introduce new propaganda signals and may have increased language variation among the under-represented classes, thereby making it harder for the model to learn clear patterns. Additionally, inaccuracies in back-translation may have led to a loss of meaning or the omission of important content, which can be critical for detecting manipulation techniques. Examples of such mistranslations are provided in Table 3.

Table 3
Inaccurate back-translation examples

Original part	Translated part	Back-translated part
Жорсткий снарядний голод.	A severe shell famine is required.	Обов'язковим є сильний голод.
Мда, сподіваюсь його знайдуть, а то щось довго чекали))	Mda, I hope to find him, or have waited a long time for him)	Mda, я сподіваюся знайти його, або чекати на нього довгий час).
Якщо речі прямо називати своїми іменами.	If you're going to call it your name	Якщо ви збираєтеся назвати це вашим ім'ям

Finally, although previous research found back-translation to be useful to be useful in certain text classification scenarios [34], the experiments conducted with social media news in Ukrainian and Russian using the Helsinki-NLP Opus MT translation models were not as effective.

Threshold tuning was also experimented with, since the default decision threshold of 0.5 on the sigmoid outputs may not be optimal for each class. A grid search was performed to determine per-label thresholds that maximised macro-F1 on the validation set. Although this process showed a slight increase in validation F1, the application of these optimized thresholds to the test set resulted in a decrease in macro-F1 by approximately 0.04. In other words, the threshold tuning with the grid search approach was found to have overfit the validation idiosyncrasies. Consequently, a uniform threshold of 0.5 was reinstated for final evaluation, as it proved to be more robust.

5. Results evaluation

To better understand the model's behaviour, its performance on each propaganda technique label was examined. In Table 4, the per-technique precision, recall, and F1 scores on the validation set for the best model are provided.

A clear correlation between a label's frequency and the model's F1 score is observed from these results. Best performance is recorded in frequent classes such as Loaded Language, which receives the highest support. For loaded_language, an F1 of approximately 0.72 is achieved, with a precision of about 0.79 and a recall of approximately 0.66. This indicates that both high sensitivity and specificity in detecting loaded language are attained, likely due to the abundance of examples from which the associated patterns—such as strong emotional or exaggerated wording—could be learned. A decent F1, approximately 0.61, is also observed for another relatively frequent class, glittering_generalities. In contrast, much lower F1 scores, roughly 0.20–0.35, are obtained for rare techniques such as bandwagon, whataboutism, and straw_man. For instance, the bandwagon, which accounts for only about 3–4% of samples, is recorded with an F1 of about 0.2, a precision of 0.14, and a recall of 0.35. Some bandwagon instances are identified, as evidenced by a recall of nearly 35%, but this is achieved at the cost of a large number of false positives, resulting in a precision of only 14%. A similar trend is observed for whataboutism (F1 ~0.24) and appeal_to_fear (F1 ~0.29), where a moderate recall (in the range of 0.4–0.5) is accompanied by very low precision, indicating that many segments are flagged as these techniques, albeit often incorrectly. This behaviour is attributed directly to the Weighted BCE Loss strategy, in which high weight is assigned to minority classes during training. Consequently, a recall-oriented approach is adopted for these classes, with a preference for predicting a rare technique—even at the cost of triggering some false alarms—in order to avoid missing true instances. From a macro-F1 perspective, this trade-off is acceptable because a slight increase in recall can boost the F1 score as long as precision is not drastically reduced; however, it does imply that further post-processing or human vetting would be required in practical applications where false positives are of concern.

Table 4
Model evaluation results per-technique

Technique	Precision	Recall	F1-score	Number of instances
appeal_to_fear	0.2131	0.4333	0.2857	300
bandwagon	0.1447	0.3548	0.2056	157
cherry_picking	0.3801	0.6373	0.4762	512
cliche	0.2292	0.6237	0.3353	463
euphoria	0.3129	0.5543	0.4000	462
fud	0.3548	0.7143	0.4741	385
glittering_generalities	0.5180	0.7423	0.6102	483
loaded_language	0.7827	0.6658	0.7196	1973
straw_man	0.2459	0.5357	0.3371	138
whataboutism	0.1685	0.4688	0.2479	158

Relatively strong performance on a few medium-frequency classes was achieved by the model. For example, F1 scores of 0.47 were recorded for cherry_picking and fud (fear, uncertainty, doubt),

which, while not high, were superior to those of the rare classes. A few hundred training instances were provided for these techniques, which appeared to be sufficient for the model to capture distinguishing features In the case of cherry_picking (selective truth), which is often characterized by numerical or factual claims, these patterns were captured, as demonstrated by a precision of 0.38 and a recall of 0.63. This high recall suggests that sensitivity to any pattern resembling a factual claim or data point has been developed, although some non-cherry-picking content was also mislabeled. Conversely, slightly lower F1 scores of approximately 0.33 for cliché and 0.4 for euphoria were observed, despite similar representation, possibly because these classes are more abstract or subjective, making consistent learning more difficult.

Overall, a macro-averaged precision of approximately 0.33 was obtained across classes, while a macro-averaged recall of about 0.57 (resulting in a macro-F1 of roughly 0.40) was recorded. This imbalance between precision and recall confirms that, under the influence of a weighted BCE loss, an over-prediction of minority labels is favored to capture as many true instances as possible. In contrast, different results are shown by the micro-averaged scores: a micro-precision of about 0.4 and a micro-recall of approximately 0.62 were observed, reflecting performance across all label decisions collectively. The higher micro-precision compared to the precision of many individual rare classes indicates that, when all negative examples are considered, the model is correct in most cases by not predicting a rare label where it is not present. A micro-recall of 0.62 was recorded, reflecting the overall proportion of true labels that were correctly predicted. This value was largely influenced by the very high recall achieved on the fud, glittering_generalities and loaded_language classes, which contributed a significant number of true labels.

In summary, these metrics indicate that high effectiveness is achieved for the majority class and reasonable performance is obtained for several mid-frequency classes, while the rarest techniques continue to pose challenges, with some instances being detected (non-zero recall) but many false positives being produced, thereby resulting in low precision and F1.

6. Conclusion

In conclusion, it was demonstrated that fine-tuning a pre-trained transformer model, when combined with strategies for addressing class imbalance and careful hyperparameter tuning, can provide a strong baseline for multi-label propaganda technique classification in Ukrainian/Russian languages. A macro-F1 score of approximately 0.41 was attained on the evaluation set and 0.4 on test set, representing a significant improvement over baseline methods. It was shown that transfer learning from a large unlabeled corpus can provide a solid foundation even in a low-resource setting, while the implementation of a weighted BCE loss function was found to be essential for mitigating the effects of extreme class imbalance.

Discriminative fine-tuning, in which different learning rates were applied to distinct layers, was also found to help preserve the pre-trained language representations while allowing rapid adaptation of the classifier. Conversely, data augmentation via back-translation did not produce the expected gains, suggesting that further research is needed to generate synthetic training examples that capture the nuanced patterns of propaganda language. Future work is recommended to explore ensemble methods, cross-lingual transfer, and techniques aimed at improving precision for minority classes without sacrificing recall. It is hoped that these findings will be used to refine existing methods and inspire the development of new approaches for detecting propaganda and misinformation, particularly in low-resource environments.

Declaration on Generative AI

During the preparation of this work, the authors used ChatGPT, Grammarly in order to: Grammar and spelling check, Paraphrase and reword. After using this tool/service, the authors reviewed and edited the content as needed and takes full responsibility for the publication's content.

References

[1] A. Barrón-Cedeño, G. Da San Martino, H. Wachsmuth, R. Petrov, and P. Nakov. SemEval-2020 Task 11: Detection of Propaganda Techniques in News Articles. In: Proceedings of the SemEval-2020 Workshop, 2020.

[2] UNLP. UNLP Shared Task. Available at: https://unlp.org.ua/shared-task/ (accessed 2025-05-03). Content licensed under CC BY-NC-SA 4.0.

[3] Y. Chai, H. Xie, and J. S. Qin. Text Data Augmentation for Large Language Models: A Comprehensive Survey of Methods, Challenges, and Opportunities. ArXiv, abs/2501.18845, 2025.

[4] J. Howard and S. Ruder. Universal Language Model Fine-tuning for Text Classification. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 328–339, Melbourne, Australia, 2018. Association for Computational Linguistics.

[5] B. Sabiri, A. Khtira, B. Asri, and M. Rhanoui. Analyzing BERT's Performance Compared to Traditional Text Classification Models. 2023, pp. 572–582. doi:10.5220/0011983100003467.

[6] A. Imran, H. Hodnefjeld, Z. Kastrati, N. Fatima, S. Daudpota, and M. Wani. Classifying European Court of Human Rights Cases Using Transformer-Based Techniques. IEEE Access 11 (2023) 55664–55676. doi:10.1109/ACCESS.2023.3279034.

[7] J. Keller, N. Rehbach, and I. Zafar. nancy-hicks-gribble at SemEval-2023 Task 5: Classifying and generating clickbait spoilers with RoBERTa. In: Proceedings of the SemEval-2023 Workshop, pp. 1712–1717, 2023. doi:10.18653/v1/2023.semeval-1.238.

[8] P. Tumuluru, S. Hussain, L. Kankanala, S. Shoaib, D. Madhu, and C. Kumar. Advancing Twitter Sentiment Analysis: An Ensemble Approach with Transformer-XL, RoBERTa, and XGBoost. In: 2023 International Conference on Self Sustainable Artificial Intelligence Systems (ICSSAS), pp. 944–950, 2023. doi:10.1109/ICSSAS57918.2023.10331828.

[9] T. Huang, Q. She, and J. Zhang. BoostingBERT: Integrating Multi-Class Boosting into BERT for NLP Tasks. ArXiv, abs/2009.05959, 2020.

[10] Y. Ren, D. Wu, A. Khurana, G. Mastorakos, S. Fu, N. Zong, J. Fan, H. Liu, and M. Huang. Classification of Patient Portal Messages with BERT-based Language Models. In: 2023 IEEE 11th International Conference on Healthcare Informatics (ICHI), pp. 176–182, 2023. doi:10.1109/ICHI57859.2023.00033.

[11] M. Al-Garadi, Y. Yang, H. Cai, Y. Ruan, K. O'Connor, G. Graciela, J. Perrone, and A. Sarker. Text classification models for the automatic detection of nonmedical prescription medication use from social media. BMC Medical Informatics and Decision Making 21 (2021). doi:10.1186/s12911-021-01394-0.

[12] A. Kiulian, A. Polishko, M. Khandoga, O. Chubych, J. Connor, R. Ravishankar, and A. Shirawalmath. From Bytes to Borsch: Fine-Tuning Gemma and Mistral for the Ukrainian Language Representation. ArXiv, abs/2404.09138, 2024. doi:10.48550/arXiv.2404.09138.

[13] F. Gomez, A. Rozovskaya, and D. Roth. A Low-Resource Approach to the Grammatical Error Correction of Ukrainian. In: Proceedings of the Second Ukrainian Natural Language Processing Workshop (UNLP), 2023. doi:10.18653/v1/2023.unlp-1.14.

[14] J. Rieger, K. Yanchenko, M. Ruckdeschel, G. Von Nordheim, K. Von Königslöw, and G. Wiedemann. Few-shot learning for automated content analysis: Efficient coding of arguments and claims in the debate on arms deliveries to Ukraine. ArXiv, abs/2312.16975, 2023. doi:10.48550/arXiv.2312.16975.

[15] S. Tiutiunnyk and V. Dyomkin. Context-Based Question-Answering System for the Ukrainian Language, 2020.

[16] M. Sergii and O. Oleksandr. Data preprocessing and tokenization techniques for technical Ukrainian texts. Applied Aspects of Information Technology, 2023. doi:10.15276/aait.06.2023.22.

[17] A. Tarekegn, M. Giacobini, and K. Michalak. A review of methods for imbalanced multi-label classification. Pattern Recognit. 118 (2021) 107965. doi:10.1016/j.patcog.2021.107965.
[18] Hugging Face. youscan/ukr-roberta-base. Available at: https://huggingface.co/youscan/ukr-roberta-base (accessed 2025-04-03).
[19] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. 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), pp. 4171–4186, Minneapolis, Minnesota, 2019. Association for Computational Linguistics.
[20] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, … V. Stoyanov. RoBERTa: A robustly optimized BERT pretraining approach. ArXiv preprint arXiv:1907.11692, 2019.
[21] Hugging Face. ukr-models/xlm-roberta-base-uk. Available at: https://huggingface.co/ukr-models/xlm-roberta-base-uk (accessed 2025-04-03).
[22] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, … V. Stoyanov. Unsupervised cross-lingual representation learning at scale. ArXiv preprint arXiv:1911.02116, 2019.
[23] Hugging Face. ukr-models/uk-summarizer. Available at: https://huggingface.co/ukr-models/uk-summarizer (accessed 2025-04-03).
[24] L. Xue, N. Constant, A. Roberts, M. Kale, R. Al-Rfou, A. Siddhant, … C. Raffel. mT5: A massively multilingual pre-trained text-to-text transformer. ArXiv preprint arXiv:2010.11934, 2020.
[25] Hugging Face. KoichiYasuoka/roberta-base-ukrainian. Available at: https://huggingface.co/KoichiYasuoka/roberta-base-ukrainian (accessed 2025-04-03).
[26] UberText dataset. Available at: https://lang.org.ua/uk/corpora/ (accessed 2025-04-03).
[27] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. Von Platen, C., Y. Jernite, J. Plu, C. Xu, T. Scao, S. Gugger, M. Drame, Q. Lhoest, and A. Rush. HuggingFace's Transformers: State-of-the-art Natural Language Processing. ArXiv, abs/1910.03771, 2019.
[28] T. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár. Focal Loss for Dense Object Detection. IEEE Transactions on Pattern Analysis and Machine Intelligence 42 (2020) 318–327. doi:10.1109/TPAMI.2018.2858826.
[29] G. B'en'edict, V. Koops, D. Odijk, and M. De Rijke. sigmoidF1: A Smooth F1 Score Surrogate Loss for Multilabel Classification. ArXiv, abs/2108.10566, 2021.
[30] M. Rezaei-Dastjerdehei, A. Mijani, and E. Fatemizadeh. Addressing Imbalance in Multi-Label Classification Using Weighted Cross Entropy Loss Function. In: 2020 27th National and 5th International Iranian Conference on Biomedical Engineering (ICBME), pp. 333–338, 2020. doi:10.1109/ICBME51989.2020.9319440.
[31] J. Howard and S. Ruder. Universal Language Model Fine-tuning for Text Classification. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 328–339, 2018. Association for Computational Linguistics. doi:10.18653/v1/P18-1031.
[32] M. C. Hinojosa Lee, J. Braet, and J. Springael. Performance Metrics for Multilabel Emotion Classification: Comparing Micro, Macro, and Weighted F1-Scores. Applied Sciences 14(21) (2024) 9863. doi:10.3390/app14219863.
[33] J. Tiedemann, M. Aulamo, D. Bakshandaeva et al. Democratizing neural machine translation with OPUS-MT. Lang Resources & Evaluation 58 (2024) 713–755. doi:10.1007/s10579-023-09704-w.
[34] T. Bourgeade, S. Casola, A. M. Wizan, and C. Bosco. Data Augmentation through Back-Translation for Stereotypes and Irony Detection. In: Proceedings of the 10th Italian Conference on Computational Linguistics (CLiC-it 2024), pp. 90–97, December 2024.