Using LLMs for Pre-Annotation of Emotional Manipulation Techniques in a Low-Resource Language Corpus: Are We There Yet?

Rita Butkienė *, Algirdas Šukys, Edgaras Dambrauskas, Voldemaras Žitkus, Linas Ablonskis, Evaldas Vaičiukynas, Paulius Danėnas, Rimantas Butleris

Posted Date: 26 May 2026

doi: 10.20944/preprints202605.1695.v1

Keywords: prompt-engineering; manipulative technique detection; LLM-assisted annotation

 doi:10.20944/preprints202605.1695.v1

Article

Using LLMs for Pre-Annotation of Emotional Manipulation Techniques in a Low-Resource Language Corpus: Are We There Yet?

Rita Butkienė [1,*], Algirdas Šukys [1], Edgaras Dambrauskas [2], Voldemaras Žitkus [1], Linas Ablonskis [1], Evaldas Vaičiukynas [1], Paulius Danėnas [2] and Rimantas Butleris [1,2]

[1] Department of Information Systems, Faculty of Informatics, Kaunas University of Technology, 51368 Kaunas, Lithuania

[2] Centre of Information Systems Design Technologies, Faculty of Informatics, Kaunas University of Technology, 51368 Kaunas, Lithuania

* Correspondence: rita.butkiene@ktu.lt

Abstract

This paper examines whether incremental prompt engineering can enable reliable LLM based pre-annotation of corpus texts in a low resource language setting. Using Lithuanian as a case study, we systematically evaluate multiple LLM prompt designs and assess their suitability for generating emotional manipulation annotations for corpus development. We find that performance varies with task complexity, and systematic prompt refinement measurably reduces output instability. Cross-model evaluation of the best-performing prompting strategy shows consistent and similar trends over several modern LLMs. Our results demonstrate that while structured prompts substantially improve output consistency and LLM assisted annotation can roughly approximate human produced labels for well-defined categories, the quality of results produced by contemporary LLMs is unsatisfactory for automatic pre-annotation of emotional manipulation techniques in a low resource language.

Keywords: prompt-engineering; manipulative technique detection; LLM-assisted annotation

1. Introduction

Emotional manipulation can significantly influence human decisions and can be exploited by various malicious actors across public and social media. Propagandists have mastered a wide range of manipulative techniques [1,2]. Automating the detection of manipulative patterns in online texts would enable the monitoring of audience responses to public messages [3,4], the identification of emerging narrative trends, and the mitigation of harmful activities before they escalate [5]. Moreover, linguistic cues also contribute to the spread of misleading content: as ref. [6] show, emotionally charged and stylistically marked expressions can amplify the reach of deceptive messages on social media, reinforcing the need to capture such signals when modelling manipulative discourse systematically.

Large language models (LLMs) have rapidly expanded the possibilities for natural language processing in high-resource settings, yet their usefulness for low-resource languages remains underexplored. Researchers also started exploring the use of LLMs for detecting propaganda and persuasion techniques in text, both as standalone applications and as a tool to produce or augment existing corpora for related model training or finetuning tasks [7,8]. Reliable propaganda detection requires large and high-quality annotated datasets, which is costly to produce in terms of time and human effort. While existing datasets cover widely used languages such as English, datasets for smaller languages, including Lithuanian, are very sparse and lack scale and diversity. Consequently, the development of tools that automate the creation of annotated text corpora for low-resource

 doi:10.20944/preprints202605.1695.v1

languages is important. Researchers focusing on propaganda and persuasion corpora have noted the potential of LLMs to support annotation. Although current models struggle with the detection accuracy of both techniques and annotation spans, they can be easily exploited for propaganda generation [9], but the hope that LLMs may also serve as valuable assistants to human annotators [10] remains. This is also encouraged by recent research in related fields, such as event extraction with annotation guidelines [11], clinical named entity recognition [12], political leaning, emotional intensity and sarcasm [13] where LLMs were successfully applied for annotating specific segments. Unfortunately, most of such research is performed on high-resource languages like English and, as some authors point out, LLM-driven annotation may still fall-short for low-resource languages with high morphological richness [14].

Prior studies on LLM-based manipulation detection have experimented with a variety of prompting strategies. While recent work shows that carefully designed prompts can substantially improve LLM performance [15], we still lack systematic evidence on how far prompt engineering alone can go when linguistic resources, training data, and domain specific tools are scarce. A common finding across the existing studies is that robust performance ultimately depends on iterative prompt refinement [16]. The studies have identified multiple issues in prompt formulation, such as output instability and inconsistency, sensitivity to minor changes in task formulation, variations in phrasing, structure, or punctuation, which may lead to substantially inconsistent or even incorrect LLM responses [17,18]. The main reasons are attributed to both bias in training data distribution, as well as the probabilistic nature of LLMs. Therefore, while LLMs provide promising capability to handle natural semantic ambiguity and natural language, new prompt engineering approaches are necessary to provide a systematic path toward improving prompt quality.

This paper investigates prompt engineering as a practical strategy for enabling natural language processing (NLP) tasks in low resource contexts, using Lithuanian as a case study. We examine whether structured prompting can achieve reliable text classification and whether LLM generated outputs can serve as a foundation for future dataset development. In doing so, we also assess the feasibility of using LLMs to bootstrap annotation workflows which has potential to accelerate practical and precise dataset development - an important but secondary question for low resource research. We address the lack of systematic evidence on how incremental prompt-engineering strategies affect LLM stability and detection performance in a low-resource language setting and provide a set of evaluations and guidelines for future research.

Secondary, but equally important, we address the absence of a Lithuanian dataset annotated for emotional manipulation To provide a solution for this issue, we have developed a gold-standard corpus of 1000 Lithuanian news-portal comments with 5300 spans annotated by four experts, covering 14 manipulation techniques and have evaluated a set of progressively refined prompts using established LLMs with Lithuanian language support.

The objective of this study is to systematically develop and evaluate an iterative prompt-engineering strategies aimed at improving the reliability of LLMs in detecting emotionally manipulative text spans in a low-resource language. Specifically, the study seeks to answer the following research questions:

RQ1. How effectively can specific prompt engineering improve LLM performance on a low resource text tagging task for annotating emotional manipulation techniques? To what extent can it support the development of reliable datasets in low resource settings?

RQ2. Which prompting strategies, prompts, and model settings demonstrate the highest practical utility for assisting text corpus annotators?

RQ3. What degree of the output variance can be observed while executing the same prompt with different temperature settings or across different LLMs? Which prompting strategies or instruction designs minimize the output error?

Taken together, these questions guide a systematic examination of how prompt design influences model stability, accuracy, and applicability in the creation of high-quality annotated resources for low-resource language.

 doi:10.20944/preprints202605.1695.v1

2. Related Work

Research on the detection of manipulative and propaganda techniques has expanded in recent years. The following sections provide a structured overview of these developments, focusing on shared tasks, multilingual initiatives, annotated datasets, detection methods, and LLM applications.

2.1. Initiatives in Detecting Propaganda and Persuasion Techniques

Several shared tasks have shaped the development of computational methods for detecting propaganda and persuasion techniques. SemEval-2020 Task 11 has established a benchmark for span identification and technique classification in English news articles [19]. Subsequent tasks expanded the scope: SemEval-2021 incorporated multimodal detection from text and images [20], SemEval-2023 introduced a multi-lingual dataset with an extended taxonomy [21], and SemEval-2024 focused on meme-based propaganda [22].

Additional shared tasks further extended language coverage, including Arabic datasets from WANLP 2022 and ArabicNLP 2023 [23,24], a multi-lingual FIGNEWS task [25], the CLEF 2024 CheckThat! Lab [26], and the SlavicNLP 2025 shared task covering five Slavic languages [27]. The related UNLP 2025 task addressed social-media manipulation detection in Ukrainian [28].

Dataset development initiatives have further addressed resource gaps in specific languages and domains. These include a Mandarin corpus and a model for propaganda detection [29], a dataset for code-switched English–Roman Urdu posts [30], a large Czech language dataset with fine-grained annotations for pro-Russian propaganda [31], the Lithuanian HALT PROP corpus [32], and the MultiProSE dataset, which is jointly annotated with propaganda, sentiments, and emotions in Arabic [33]. Together, these initiatives have advanced methodological standards and stimulated the creation of several new datasets, which now constitute the basis for evaluating both supervised models and emerging LLM-based approaches.

2.2. Datasets for Detecting Manipulative Techniques

Existing datasets differ in structure, scope, and annotation strategy, reflecting both the complexity of manipulation as a phenomenon and the variability of human annotation practices.

A key distinction across datasets is the **annotation granularity**. Early datasets, such as Proppy [3], applied document-level labels, which tend to introduce noise during model training and provide no basis for explaining model outputs. The shift to span-level annotations, initiated in the PTC–SemEval2020 corpus [34] has improved both model accuracy and interpretability. Therefore, our analysis focuses exclusively on span-level datasets.

The main publicly available span-annotated datasets are summarized in Table 1. Detailed per-language distributions for multilingual datasets are reported in the original shared-task publications. All listed datasets are human-annotated and provide span-level labels for propaganda or persuasion techniques.

Table 1. Publicly available datasets annotated at the span-level.

Dataset	Language(s)	Source / Topic	#Techniques	#Texts	#Spans
SemEval-2023 [21]	EN, FR, DE, IT, PL, RU, GE, EL, ES	News articles; multi-topic	23	1835	~49,444
CLEF-2024 [26]	AR, BG, EN, SI, PT	News articles; multi-topic	23	1929	18,714
SlavicNLP-2025 [27]	BG, HR, PL, RU, SI	Parliament debates; social media	25	285	8857
PTC–SemEval-2020 [34]	EN	News articles; multi-topic	14	536	8981
WANLP 2022 Arabic [23]	AR	Tweets; multi-topic	20	930	1942

 doi:10.20944/preprints202605.1695.v1

ArAIEval [24]	AR	News article snippets, tweets, multi-topic	23	3189	2451
ArPro [35]	AR	News article snippets; multi-topic	23	8000	20,487
Propaganda-codeswitched-text [30]	EN-Urdu	Social media	20	1030	2577
ZenPropaganda [36]	RU	Social media; COVID-19	35	125	4684
Czech corpus [31]	CS	News articles; Russian propaganda	8	8646	~13,634
HALT-PROP [32]	LT	News articles; Russian propaganda	10	1000	13,591

Datasets also vary in their **annotation schemas**. While many rely on the fine-grained taxonomy introduced by ref. [34], several datasets include rare techniques and expand or adapt technique sets to specific media or domains. Examples include the reduction to 14 techniques in SemEval-2020, the expansion to 20 techniques in WANLP-2022, the adaptation of technique lists for meme-based propaganda in SemEval-2021, and the hierarchical 23 technique taxonomy introduced in SemEval-2023 and later adopted in ArAIEval, ArPro, and CLEF-2024. SlavicNLP-2025 expanded the taxonomy further by adding two more techniques. In contrast, the Czech corpus and HALT-PROP employ smaller sets of techniques, while ZenPropaganda includes the largest variety.

Text corpora used in datasets differ **in source type and thematic** focus. Many datasets comprise news articles or social media posts. In contrast, others focus on specific domains, such as parliamentary debates [27], state-backed information operations [29], and COVID-19-related content [36]. Multimodal initiatives such as SemEval2021 and SemEval2024 have extended these efforts to image-text combinations, capturing a broader spectrum of manipulative communication [20,22].

Across granularity, taxonomy, source type, language, and scale, current span-annotated datasets display substantial heterogeneity. This diversity underscores the conceptual and practical challenges of modelling manipulative content and highlights the need for more efficient corpus construction methodologies.

2.3. Methods for the Detection of Propaganda and Persuasion Techniques

Supervised machine-learning approaches continue to dominate propaganda-technique detection, with transformer-based models achieving state-of-the-art performance across multiple shared tasks (e.g., refs. [28,37,38]). However, the development of such models requires large, high-quality annotated corpora, which are both costly and labor-intensive to create. The limited availability of annotated datasets for low-resource languages—together with their narrow topical coverage and restricted source diversity—limits progress in propaganda-technique detection [21,22,28,39]. The emergence of LLMs shifted research focus toward methods that could reduce the dependence on extensive labelled data.

Recent studies have examined the capacity of large language models to detect manipulative techniques, yet empirical results remain mixed. Experiments by ref. [40] demonstrated that GPT-4 can outperform baseline models using few-shot prompts, but follow-up work by [41] failed to replicate these findings and reported low stability and weak span level detection. Other studies likewise note high variability across prompting strategies, model versions, and linguistic settings, e.g., refs. [42,43].

Research on Arabic datasets shows that GPT-4 often underperforms finetuned transformers and needs prompts that contain extensive contextual guidance. Even then, the results require substantial postprocessing due to inaccurate span boundaries [35,44]. Nevertheless, several works indicate that LLMs can support corpus construction by assisting with span identification or pre-annotation. Ref. [45] demonstrated that GPT-3.5 can efficiently annotate rhetorical and linguistic features, and ref. [46]

 doi:10.20944/preprints202605.1695.v1

showed that prompt-based learning can reduce annotation costs while maintaining high label quality.

Authors of ref. [47] further expanded this line of inquiry by investigating whether LLMs can autonomously refine prompts for one-shot classification of 25 persuasion techniques in the SlavicNLP 2025 shared task. Using GPT-4.1-mini, they compared prompts incorporating human-authored definitions with those augmented by LLM-generated revised definitions. The refined definitions improved performance selectively — most notably for Slovene, Bulgarian, and Polish. The LLM-generated definitions were longer, more redundant in comparison to baselines, and their effectiveness varied substantially across techniques. This study reinforces the broader finding that prompting choices significantly affect model outputs and that improvements are often technique, language, and prompt dependent.

Across studies, LLMs show potential when supported by carefully engineered prompts and sufficient contextual scaffolding, yet their detection accuracy, consistency, and span-level precision remain limited. These limitations underscore the need for systematic investigation of prompt engineering strategies and highlight the value of LLM-assisted corpus construction. This is especially relevant for low-resource languages where high-quality annotated datasets remain scarce.

3. Research Methodology

To address the research questions, the study follows a structured experimental plan composed of eight stages (Figure 1): 1) construction of an annotated dataset of emotional manipulation techniques; 2) design of prompts following the selected prompting strategy; 3) execution of the designed prompts on GPT-4.1[1]; 4) post-processing of prompt outputs for analysis; 5) development of an analysis and evaluation framework; 6) analysis of prompt outputs; 7) evaluating prompting strategies; and 8) testing results with other LLMs.

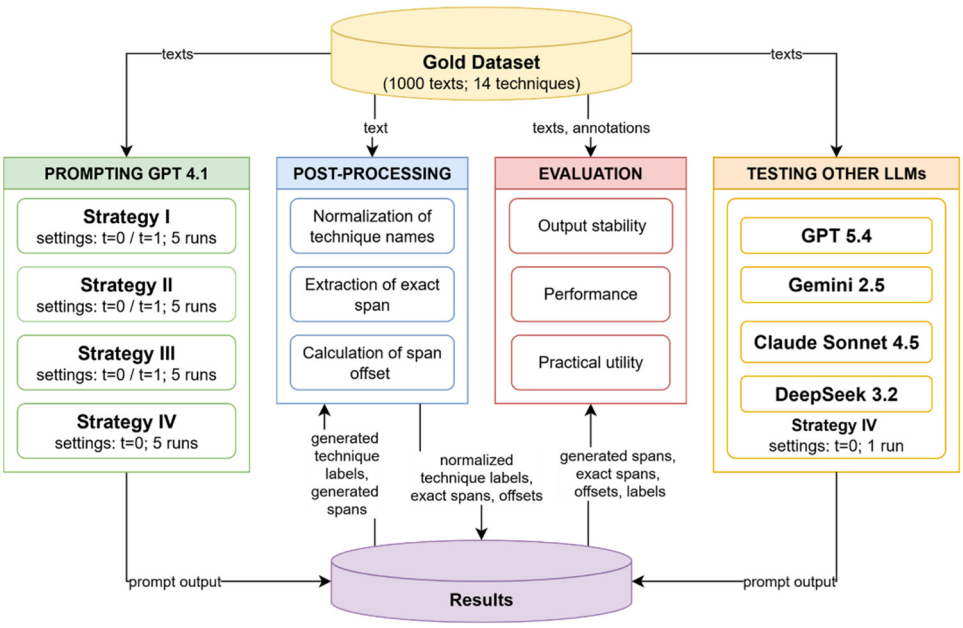


Figure 1. Research methodology.

3.1. Dataset Development

The gold dataset employed in this study was created by the authors, as no publicly available Lithuanian dataset annotated for emotional manipulation techniques existed prior to this work. The dataset, publicly available in ref. [48], consists of 1000 user-generated comments collected from major

[1] https://openai.com/index/openai-api/

 doi:10.20944/preprints202605.1695.v1

Lithuanian news portals and social media discussions. Most of the comments (895) originate from the LITIS corpus[2], which aggregates anonymous user comments posted under articles published between 2010 and 2014 on the Lithuanian news portals *Delfi.lt* and *Lrytas.lt*. These comments were initially retrieved using keyword-based queries targeting politically and socially sensitive topics, then manually filtered to retain instances that potentially contained manipulative rhetoric. To broaden topical coverage and ensure temporal relevance, an additional 105 highly engaging comments were collected from Facebook discussions (2021–2025) on the pages of major news outlets, focusing on articles that received at least 100 comments. Most comments are associated with political news, with smaller portions drawn from crime, health, and miscellaneous categories. The comments are unedited and reflect authentic online discourse, including non-standard language and spelling variation.

Each comment was annotated at the span level for 14 predefined emotional manipulation techniques. The category boundaries and annotation rules were described in guidelines, following established practices in prior studies [19,49]. Annotations could overlap and multiple labels could be applied to the same textual fragment, reflecting the layered nature of manipulative rhetoric. Spans were required to be self-contained and interpretable in isolation, ensuring semantic completeness when extracted from the broader comment context.

All comments were independently annotated by four annotators using the Label Studio platform. Annotators first labelled comments individually and could flag ambiguous cases for discussion. Inter-annotator agreement, measured using *Krippendorff's Alpha_U* [50], was moderate overall (0.59), with variation across techniques. Disagreements were systematically resolved through a structured, multistage consensus procedure involving pairwise discussions followed by full team review. This process ensured that all final annotations reflected shared expert agreement.

In total, the annotators identified 5300 manipulation spans across the corpus. The distribution of spans is highly imbalanced (Figure 2). On average, each comment contains 5.3 annotated spans, a direct consequence of the span-based, overlapping annotation strategy. Comment length exhibits substantial variability, ranging from 16 to 2000 characters; nevertheless, slightly more than half of the comments are relatively short, containing fewer than 200 characters. This variability further contributes to differences in the number and types of techniques identified across individual comments. This imbalance reflects both natural usage patterns in real-world discourse and the topical focus of the sampled material.

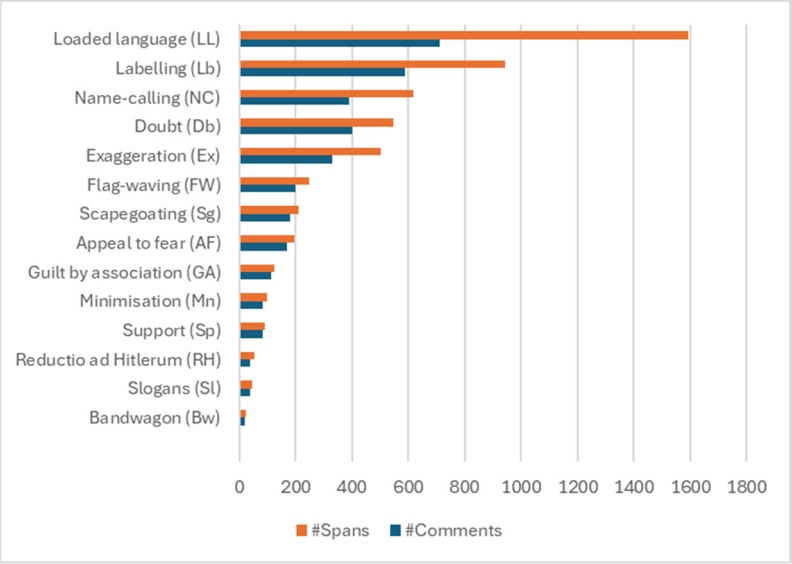


Figure 2. Distribution of annotated spans by technique.

[2] http://hdl.handle.net/20.500.11821/11

 doi:10.20944/preprints202605.1695.v1

3.2. Prompt Engineering Strategies

We have incrementally defined four prompting strategies. In accordance with the typology of prompting methods by [51], we have applied the prompt template and prompt answer engineering. The first prompt template was manually constructed and subsequently refined using systematic paraphrasing and reformulation techniques. Each subsequent prompting strategy builds upon incremental refinement of the preceding prompt template, introducing increased specificity, conceptual clarity, or functional constraints. A similar methodology can be found in refs. [35,42,44]. All prompts created in this study (original prompt texts and their English translations) are provided along with the dataset [48].

The idea behind **Strategy I** was to provide a short task description, specifying only what the LLM should do and how it should present the output. The model was not informed about which manipulation techniques to look for, nor which emotions the analyzed comment might aim to evoke. The prompt in English is provided along with the dataset. The purpose of this type of prompt was to test the model's ability to perform the task using only its existing knowledge. Similar prompts were explored in ref. [44]. Using Strategy I as a baseline, we investigated the impact of expanding context and modifying instructions on performance in further strategies.

The prompt of **Strategy II** provided the model with additional context. The prompt included a list of manipulation techniques and a list of emotions that the text may aim to elicit. Similar prompts had previously been tested for identifying technique occurrences in Arabic and several Slavic languages [35,41,44]. Apart from the inclusion of an emotion list, our prompt differed in its instructions regarding the output structure and format.

The prompts of **Strategy III** targeted only one manipulation technique at a time. A separate prompt was created for each technique, containing its definition and a list of possible induced emotions. Similar prompts were used for detecting techniques and/or fragment representations in English and several Slavic languages [5,43,47].

The output instructions in the prompts of Strategies I to III instructed the model to indicate whether at least one (or a specific) manipulation technique was found in the comment text. If a technique was detected, the model was to provide the technique name, the span representing the technique, and the emotion the text aimed to evoke.

Strategy IV was an attempt to compensate for the shortcomings of Strategy III. The prompts of Strategy IV were also designed to detect spans of only one manipulation technique at a time. The prompt formulation included a definition of the technique while omitting the list of emotions and an instruction to identify the intended emotional effect. The query was supplemented with explicit conditions that the span representing the technique must satisfy, as well as with modified output instructions. These modifications were aimed at refining the technique definition (the context) and encouraging the model to self-verify its reasoning.

All prompts were intended for span detection using a zero-shot strategy and were written in Lithuanian, except for the technique names and definitions, which were provided in English. The instructions were formulated in Lithuanian to provide a culturally and linguistically relevant context for the target text, while technique names and definitions were maintained in English to leverage the model's extensive pre-training on high-resource English datasets and established taxonomies, therefore mitigating potential semantic shift caused by translating technical terminology.

Strategies I to III were tested at temperatures of 0 and 1. Other studies usually executed prompts only at temperature 0, arguing that LLMs are by nature nondeterministic and do not guarantee reproducibility across runs of the same prompt [41], and that higher temperatures increase the likelihood of hallucinations. Even zero temperature in practice results in output variance, which some authors refer to as instability [52] We decided to test our prompts at temperature 1 as well, to observe the magnitude of this effect when analyzing Lithuanian texts. After trying the temperature of 1 with Strategies I to III and consistently finding considerable output instability compared to temperature 0, we use temperature 0 setting only for Strategy IV.

 doi:10.20944/preprints202605.1695.v1

Table 2 provides a structured comparison of the prompting strategies, summarizing the key dimensions along which they differ and overlap. Importantly, the differences between strategies are not isolated to a single experimental factor. The strategies vary concurrently in task formulation, contextual information, output structure, label space size, and the presence of explicit validation or self-restraint instructions. As a result, performance differences across strategies cannot be causally attributed to any single design decision. The comparison should therefore be interpreted as an exploratory analysis of prompt design variants rather than a controlled ablation study.

Table 2. Comparison of prompting strategies.

Strategy	Task Scope	Context Provided	Emotion Information	Output Instructions	Output Structure
I	Multiple techniques	No technique definitions or lists	None	output format, excerpt accuracy	manipulative or not, technique, span, emotion evoked
II	Multiple techniques	List of techniques	List of emotions	output format, excerpt accuracy	comment id, manipulative or not, technique, span, emotion evoked
III	Single technique per prompt	Technique definition	List of emotions	output format, excerpt accuracy	technique identified or not, span, emotion evoked
IV	Single technique per prompt	Technique definition	None	output format, excerpt accuracy, span validity conditions, self-verification	span or “not found”

3.3. Prompting

We have used GPT-4.1 for designing prompting strategies, because at the time it was the mainstream version reported to have good performance in related tasks [53,54]. Previous studies [40–42] have shown that GPT models generally exhibit superior performance compared with other LLMs, such as Gemma[3], Llama[4], or Mistral[5]. Moreover, not all LLMs demonstrate sufficient capability in processing and analyzing Lithuanian-language texts. For these reasons, we restricted the design phase of this study to OpenAI’s GPT model, specifically version 4.1.

In total, 30 prompts were created. To assess the stability of prompt performance, each prompt, except for those representing Strategy IV, was executed five times using a temperature of 0. A description of the prompting procedure, all prompts and their corresponding results, together with the gold-standard dataset, are available [48].

To examine the transferability of the results obtained with a GPT-4.1 model, the prompts of the final strategy (Strategy IV) were additionally evaluated by prompting other mainstream LLMs. The other mainstream LLMs tested are: GPT-5.4[6], Gemini 2.5[7], Claude Sonnet 4.5[8] and DeepSeek-V3.2[9]. Testing of these LLMs was performed at a temperature of 0, and each prompt was executed only once

[3] https://deepmind.google/models/gemma/

[4] https://www.llama.com/

[5] https://iamistral.com/

[6] https://openai.com/index/introducing-gpt-5-4/

[7] https://blog.google/innovation-and-ai/models-and-research/google-deepmind/gemini-model-thinking-updates-march-2025/

[8] https://www.anthropic.com/news/claude-sonnet-4-5

[9] https://arxiv.org/abs/2512.02556

 doi:10.20944/preprints202605.1695.v1

per model. The evaluation was focused solely on performance comparison and not on output variability or stability across runs.

3.4. Result Post-Processing

In prior studies [35,41,42,44], the prompts have instructed the model to return the offset positions of each identified text span. Empirical observations indicated that the model did not reliably produce accurate offsets. Consequently, we opted not to request positional information, as preliminary experiments demonstrated that the fragments extracted by the model did not consistently align with the exact spans in the source texts. Accordingly, all fragments generated by the model underwent a post-processing, whereby precise copies of the corresponding text spans were retrieved, and their start and end offsets were computed.

Furthermore, the output of the Strategy II prompts required additional post-processing, as the diversity of the generated labels for manipulation techniques greatly exceeded the predefined set of 14 categories. We have manually normalized the synonymous forms of assigned labels to enable comparative analysis of prompts.

4. Evaluation Framework

The analysis and evaluation of prompt outputs were conducted in accordance with three criteria—(1) the stability of prompt results, (2) the overall effectiveness of prompt performance, and (3) the suitability of the prompts for integration into the annotation workflow—each directly corresponding to the study's research questions on output variance (RQ3), extraction performance (RQ1), and practical utility for human annotators (RQ2).

4.1. Evaluation of the Prompt Results Stability

To systematically assess the stability of the model's performance under prompts constructed in accordance with each strategy, we have defined a set of evaluation metrics enabling a fine-grained assessment and identification of prompt behavior patterns. The metrics target four aspects of output stability:

1. *Comment level classification stability*. $RC_i = 1 - MC_i/MC$where $MC_i$is the number of comments flagged as manipulative in the run i, and MC is the total number of comments flagged as manipulative at least once across all runs. Comparing RC_iacross runs of the same prompt indicates the extent to which the model consistently classifies the same comments as manipulative during repeated execution of that prompt.

2. *Span level identification stability*. $RF_i = 1 - MF_i/MF$ where MF_i is the number of unique fragments identified in the run i, and MF is the number of unique fragments across all runs. Comparing RF_i across runs of the same prompt shows the stability of manipulative fragment identification by that prompt, considering both raw outputs and post-processed ones.

3. *Fragment reproduction stability*. $RIF_i = 1 - IF_i/IF$ where IF_i is the number of incorrect fragments per run i, and IF is the cumulative number across all runs. Comparing RIF_iacross runs of the same prompt shows to what extent the model generates incorrect text fragments—i.e., outputs that fail to exactly reproduce the target span— for that prompt.

4. *Stability of technique label generation*. $RL_i = L_i/L$ where $L_i$is the number of unique labels found in the run i, and L is the total number of unique labels across all runs. Comparing RL_iacross runs of prompting Strategies I and II show to what degree the model exhibits label stability.

The proposed metrics are explicitly intended to capture output consistency rather than annotation correctness. While consistency does not imply correctness—a prompt may consistently produce incorrect outputs—it represents an orthogonal and valid evaluation dimension in prompt-based annotation scenarios. Consistent model behavior across repeated executions is essential for reproducibility, comparability of prompting strategies, and reliable integration into downstream processing pipelines. By quantifying stability at multiple structural levels, the proposed metrics

 doi:10.20944/preprints202605.1695.v1

provide meaningful insight into prompt behavior patterns and support informed prompt design decisions, complementing—but not replacing—effectiveness-based evaluation.

4.2. Evaluation of the Prompts' Performance Effectiveness

Following established practices in the literature, we employ micro- and macro-level precision, recall, and *F1*-score to evaluate the prompts' effectiveness in identifying manipulative spans of text. To clarify, we have repeated the same prompts five times, and the calculation descriptions in this section apply separately to each run.

Let $C = \{c_1, c_2, \dots, c_N\}$, $N = 1000$ be a set of comments, comprising the corpus C. Each comment can be annotated with one of 14 emotional manipulation techniques, denoted as $T = \{t_1, t_2, \dots, t_M\}$, $M = 14$. For each comment $c \in C$ and for each technique $t \in T$, we define the set of gold spans $G_{c,t} = \{g_{c,t,1}, g_{c,t,2}, \dots, g_{c,t,|G_{c,t}|}\}$ and the set of predicted spans $P_{c,t} = \{p_{c,t,1}, p_{c,t,2}, \dots, p_{c,t,|P_{c,t}|}\}$ annotated with technique t. Each gold span g and predicted span p is a tuple, containing corresponding start and end symbol positions, i.e., $g = (g_s, g_e)$ and $p = (p_s, p_e)$.

Let us denote a chosen subset of techniques as S, $S \subseteq T$. We have conducted three distinct kinds of prompt result evaluation: a) per-technique evaluation, where $S = \{t\}$; b) micro-averaged overall evaluation where $S = T$; and c) macro-averaged overall evaluation where $S = T$.

For per-technique evaluation, we calculated the sums of true positives (*TP*), false positives (*FP*), and false negatives (*FN*) across all comments. For micro-averaged evaluation, we computed the sums of *TP*, *FP*, and *FN* across all comments and for all techniques using (1), (2), and (3).

$$TP(S) = \sum_{c \in C} \sum_{t \in S} TP_{c,t} \tag{1}$$

$$FP(S) = \sum_{c \in C} \sum_{t \in S} FP_{c,t} \tag{2}$$

$$FN(S) = \sum_{c \in C} \sum_{t \in S} FN_{c,t} \tag{3}$$

We calculate the *Precision* and *Recall* per-technique and micro-averaged metrics using (3) and (4), and macro-averaged metrics using (5) and (6), taking the calculated *Precision* and *Recall* of each technique accordingly.

$$Precision(S) = \frac{TP(S)}{TP(S) + FP(S)} \tag{3}$$

$$Recall(S) = \frac{TP(S)}{TP(S) + FN(S)} \tag{4}$$

$$Precision_{macro} = \frac{1}{T} \sum_{t \in T} Precision(t) \tag{5}$$

$$Recall_{macro} = \frac{1}{T} \sum_{t \in T} Recall(t) \tag{6}$$

For each technique, *micro-* and *macro-F1*-score are calculated with (7) by taking the corresponding *Precision* and *Recall* metrics.

$$F1 = 2 \cdot \frac{(Precision \cdot Recall)}{(Precision + Recall)} \tag{7}$$

We evaluated the results of prompts by using two levels of granularity: a) symbol-based matching and b) span-based thresholded matching.

The symbol-based matching scheme follows [55] and is widely used in SemEval tasks involving span identification. We denote the *true positives*, *false positives*, and *false negatives* calculated at the symbol-level as TP_{sm}, FP_{sm}, and FN_{sm} respectively. Symbols that occur in both predicted and gold spans are counted as TP_{sm}. Symbols that occur in predicted spans but are absent from gold spans are

 doi:10.20944/preprints202605.1695.v1

counted as FP_{sm}. Symbols present in gold spans but absent from the predicted spans are counted as FN_{sm}.

The span-based thresholded matching scheme quantifies the divergence between model predictions and human gold-standard annotations, providing a measure of alignment at the fragment level [26]. In span-based matching, we count true positives, false positives, and false negatives at the span level and denote them as TP_{sp}, FP_{sp}, and FN_{sp} respectively. First, the coverage length is calculated by (8). For verbosity, we denote $p_{c,t}$ as p and $g_{c,t}$ as g. Second, the covered portion of a gold span is calculated by using (9).

$$|p \cap g| = \max(0, \min(g_e, p_e) - \max(g_s, p_s)) \tag{8}$$

$$coverage(p, g) = |p \cap g|/|g| \tag{9}$$

Having the $coverage(p, g)$, we can calculate TP_{sp} and FP_{sp}. There are two cases in which a predicted span p is counted as *TP*. In the first case, p should be completely contained within a gold span g. In the second case, p may extend outside the gold span g, but the intersecting portion of p and g must exceed 50%. In both cases, p is considered useful, and thus *TP*, because it directs the annotator to the correct text position where annotation should be added.

When multiple predictions overlap the same gold span, we choose the one with the largest intersection with the gold span $g_i$to be considered as *TP*, the remaining overlapping predictions are counted as *FP*. We denote the *TP* span as $p_{max,i}$, as shown in (11). The $P'(g_i)$, shown in (10), represents a valid prediction list for a specific gold span.

$$P'(g_i) = \{p \in P \ \ if \ \ |p \cap g_i| = |p| \ \vee coverage(p, g_i) \geq 0.5\} \tag{10}$$

$$p_{max,i} = \max_{p \in P'} |p \cap g_i| \tag{11}$$

This approach prevents artificial inflation in precision measurement in cases where the model overly fragments the annotations, producing many shorter annotations instead of one longer one. In such instances, the human annotator would need to perform additional work to combine the small annotations into a single annotation. Thus, each gold span contributes at most one *TP*, and the total number of *TP*s can be determined by counting gold spans with at least one valid prediction, as shown in (12).

$$TP_{sp} = |\{g_i \in G \ if \ P'(g_i) \neq \emptyset\}| \tag{12}$$

For *FP*s, we count predictions that were never chosen as the best predictions using (8) by (13). For *FN*s, we count gold spans with no predictions using (14).

$$FP_{sp} = |P \setminus \{p_{max,i} \ if \ g_i \in G \ and \ P'(g_i) \neq \emptyset|$$ (13)

$$FN_{sp} = |\{g_i \in G \ if \ P'(g_i) = \emptyset\}| \tag{14}$$

These metrics are calculated separately for each comment and technique. To compute corpus-level and micro-averaged *TP*, *FP*, and *FN* metrics, we use (1), (2), and (3).

4.3. Measures of the Suitability of the Prompts

We consider a prompt to be sufficiently robust for practical use if (a) it successfully retrieves at least half of the relevant text spans, and (b) if more than half of the text spans produced by the prompt do not require correction or rejection. To evaluate the practical utility of the prompts, we computed span-based *TP*, *FP*, and *FN* metrics, distinguishing exact, partial, spurious, and missed detections, using the symbol-based *F1-score* $F1_{sm}$, which we denote as *F1* for the sake of verbosity in this section. This approach allowed for estimating both the model's effective contribution and the additional human annotation workload required by the prompt. Specifically:

$TP_{sp,F1=1}$ – exact matches, *TP* fragments requiring no correction;

$TP_{sp,F1<1}$ – partial matches, *TP* fragments with ≥50% overlap requiring partial adjustment;

 doi:10.20944/preprints202605.1695.v1

$FP_{sp,F1=0}$ – incorrect fragments, *FP* fragments with no overlap requiring complete discarding;
$FP_{sp,F1>0}$ – likely incorrect fragments, *FP* fragments with <50% likely requiring removal;
FN_{sp} – *FN* fragments that the annotator must fully identify manually.

Let us denote the set of all (g,p) span pairs resulting in *TP*s as L. We express the distribution of *TP*s by symbol-based F1-score into categories (i.e., $TP_{sp,F1=1}$, and $TP_{sp,F1<1}$) as a ratio between the number of *TP*s in that category and the number of all *TP*s, as shown in (15).

$$TP_{sp,F1=1} = |\{(g,p) \in L \; if \; F1(g,p) = 1\}|/TP \tag{15}$$
$$TP_{sp,F1<1} = |\{(g,p) \in L \; if \; F1(g,p) < 1\}|/TP$$

In the case of *FP*s, we calculate the largest *F1*-score of the predicted span with any of the gold spans from the same comment using (16).

$$F1_{max,p} = \max_{g \in G_c} F1(g,p) \tag{16}$$

Having $F1_{max,p}$, we calculate the distribution of *FP*s into categories (i.e., $FP_{sp,F1=0}$ and $FP_{sp,F1>0}$) as a ratio between the number of *FP*s in that category and the number of all *FP*s using the following (17) and (18).

$$FP_{sp,F1=0} = \frac{|\{p \in FP \; if \; F1_{max,p} = 0\}|}{FP} \tag{17}$$

$$FP_{sp,F1>0} = \frac{|\{p \in FP \; if \; F1_{max,p} > 0\}|}{FP} \tag{18}$$

To assess the practical usefulness of the prompt, we define a custom *F1*-score, denoted as $F1'$. $F1'$ is calculated by using the average of each variable, taken across five runs, in (19).

$$F1' = \frac{2 * TP_{sp,F1=1}}{2 * TP_{sp,F1=1} + TP_{sp,F1<1} + FP_{sp,F1=0} + FP_{sp,F1>0} + FN_{sp}} \tag{19}$$

5. Results

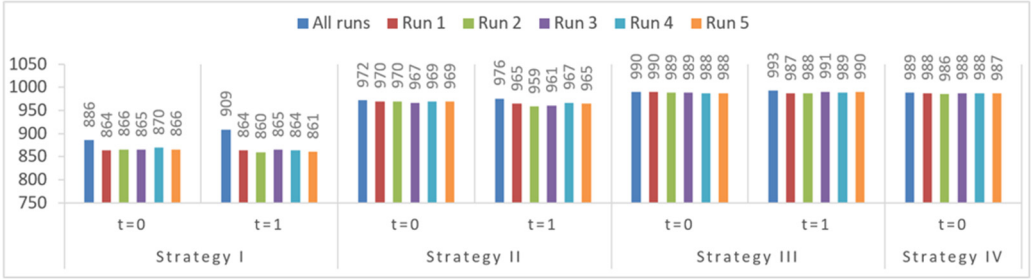
5.1. Measures of the Suitability of the Prompts

The analysis in this section uses the metrics described in Section 4.1.

5.1.1. Comment Level Classification Stability

This subsection addresses the research question RQ3 by examining the stability of the model outputs across repeated executions of identical prompts. We assessed the extent to which the model consistently classifies the same comments as manipulative across repeated executions of an identical prompt. Testing on 1000 comments shows that the results are not stable across runs (see Figure 3).

The greatest variation across runs occurred under Strategy I: with $t = 0$, detection differences RC_i were in range $[1.81\%; 2.48\%]$, and with $t = 1$, in range $[4.84\%; 5.39\%]$. Prompts of other strategies exhibited only minor variation across runs in the range $[0.1\%; 1.74\%]$, particularly when using $t = 1$. Furthermore, Strategy I prompts yielded the lowest detection rates, identifying manipulation in 88.6% $t = 0$ and 90.9% $t = 1$ of comments across all runs.



 doi:10.20944/preprints202605.1695.v1

Figure 3. Number of comments flagged as manipulative in the run i and at least once across all runs.

5.1.2. Span Level Identification Stability

Prompts of different strategies detected different numbers of manipulative spans. Even for the same strategy, different runs have resulted in somewhat different outputs. The diagrams in Figure 4 and Figure 5 present aggregated numbers of unique spans returned per strategy and per iteration. Columns marked "G" represent the spans as generated by the model. The model did not always follow the instructions to return exact copies of the original text: some returned spans were altered through corrected grammar, normalized spacing, modified punctuation, or other small edits. These modifications interfered with subsequent analyses and therefore required additional post-processing. To address this, all generated spans were verified and used to extract precise original spans from the texts, where possible. The "E" columns in the diagrams represent the number of unique spans remaining after the extraction and correction step.

An examination of both the aggregated and per-run results show that, for $t = 0$ and $t = 1$, all prompting strategies produced very similar numbers of unique spans within each run. However, when all runs are considered together, the total number of unique spans is higher than in any single run of the same strategy. This discrepancy persists even when using the exact copies of spans. The difference between per-run and cumulative unique span counts indicates variation of model results for the same task across executions. The greatest variation across runs occurred under all prompting strategies with $t = 1$—with detection differences RF_i was in range $[47.53\%; 63.24\%]$. Variation across runs with $t = 0$ was twice as low—RF_i was in range $[21.09\%; 37.79\%]$.

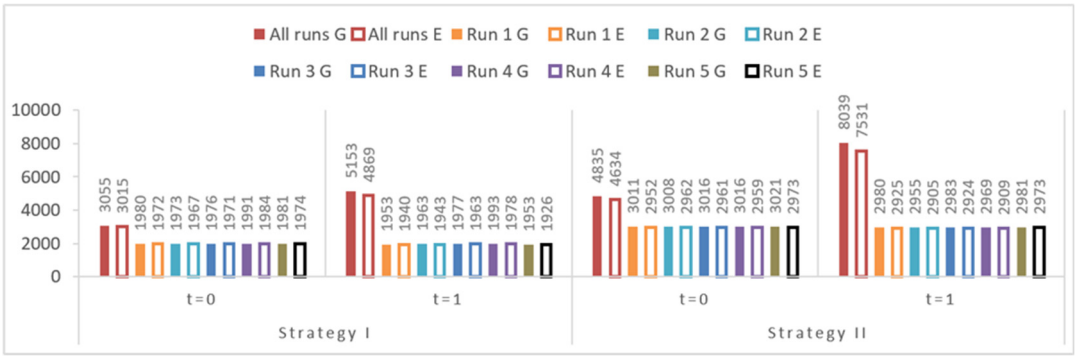


Figure 4. The number of unique fragments identified in the run i and across all runs by prompts of Strategy I and Strategy II. Note: model-generated spans ('G') underwent post-processing to extract exact textual matches ('E') from source comments, as models occasionally introduced modifications to grammar, spacing, or punctuation.

Figure 5. The number of unique span fragments identified in the run i and across all runs by prompts of Strategy III and Strategy IV. Note: model-generated spans ('G') underwent post-processing to extract exact textual matches ('E') from source comments, as models occasionally introduced modifications to grammar, spacing, or punctuation.

 doi:10.20944/preprints202605.1695.v1

5.1.3. Stability of Fragment Reproduction

We investigated the recurrent fragment reproduction stability issues observed in the outputs of prompts. The most common problems were: a) concatenated spans (e.g.,: *„putlerio"*, *„putleristai"*, *„kremliaus putleris"*, *„putleriui ir putleriukam"*); b) spans containing omissions marked with ellipses (e.g.,: *„naikinama rusų tauta... dideja mirtingumas"*); c) spans replaced or appended by explanations rather than actual excerpts (e.g.,: *„Siekiama sukelti nusivylimą, bejėgiškumą, nepasitikėjimą visomis pusėmis (NATO ir Rusija)."* in English *„The aim is to evoke disappointment, helplessness, and distrust toward all parties (NATO and Russia)"*)

The chart in Figure 6 shows the number of cases in which an exact original span could not be recovered. The variability of incorrect spans across runs is very high under prompts of all strategies executed with $t = 1 - RIF_i$ reaches the range of $[74.60\%; 85.71\%]$.

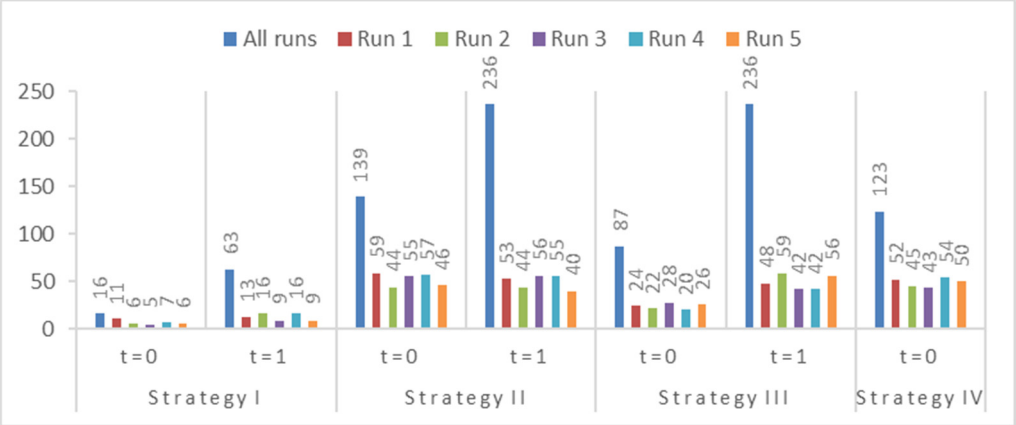


Figure 6. The number of incorrect span fragments per run i and across all runs.

Although the revised instructions introduced in Strategy IV reduced cross-run variability, they produced more incorrect spans than Strategy III. Most errors involved added explanations or summarized text. List-type output errors were eliminated, and only a few truncated span errors remained. Incorrect spans were unevenly distributed across technique-specific prompts. *Name-calling* and *Labeling* prompts generated the highest number of such cases (19 and 50), typically producing inferred descriptions rather than actual comment excerpts (e.g., "*those lazy journalists*"). For techniques such as *Bandwagon*, *Exaggeration*, *Doubt*, *Loaded Language*, and *Guilt by Association*, incorrect spans mostly consisted of explanations or condensed justifications. These differences reflect the technique-specific adaptations embedded in the prompt instructions. For example, *Name-calling* and *Labelling* prompts included additional constraints, not present in the *FW* prompt, and did not produce irreparable spans.

5.1.4. Stability of Technique Label Generation

Only Strategies I and II were analyzed with respect to label-generation behavior, as these were the only prompts instructing the model to output the name of the detected technique. The results (Figure 7) show that Strategy I—whose instructions did not specify technique names—produced an exceptionally large and highly variable set of labels.

 doi:10.20944/preprints202605.1695.v1

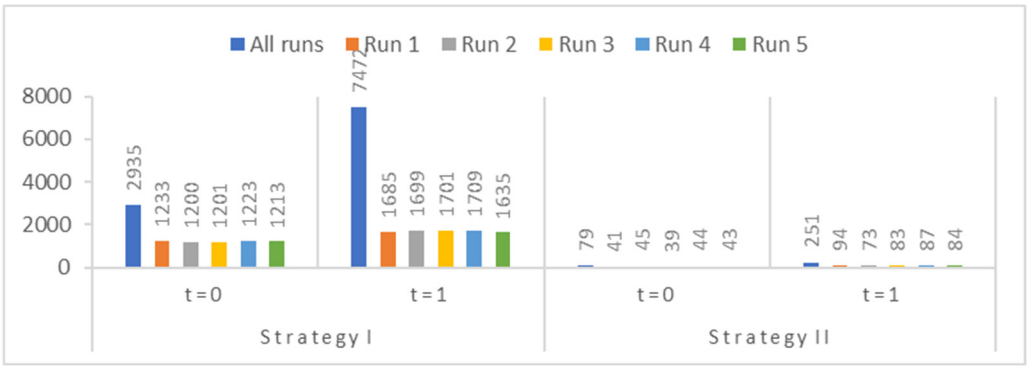


Figure 7. The number of unique labels found in the run i and across all runs.

Examination of the outputs revealed that the model frequently combined technique names, added descriptive modifiers, or generated synonymous formulations, resulting in a degree of variability that precluded reliable accuracy assessment or systematic analysis ($RL_i \in [58\%; 78\%]$). For instance, for the *Doubt* technique the following labels were generated (the authors' translation is provided): *„Sowing Doubt / Stirring Suspicion"*, *„Doubt about the Fairness of the System / Stirring Suspicion"*, *„Sowing Doubt and Distrust"*, *„Sowing Doubt / Encouraging Distrust in Institutions"*, *„Doubt / Shaming Strategy"*, *„Encouraging Doubt and Distrust"*, *„Introducing Doubts about the Actors" Morality"*, and *„Raising Doubts about Law-Enforcement Decisions"*. Normalizing such labels, whether manually or automatically, necessitates significant effort and risks introducing potential bias into comparative analysis.

The prompt of Strategy II, having lists of 14 techniques and 21 emotions, decreased the number of unique technique labels by 97% compared with Strategy I, though it remained considerably high. For analytical purposes, the generated labels were grouped into several categories: 1) technique names from the provided list; 2) emotion names from the prompt; 3) *„Appeal to + emotion/feeling"* constructions; 4) other composite or modified technique names (e.g., *„Comparison (Bandwagon / Support for US actions)"*); and 5) labels containing appended numerical identifiers. A distribution of label names is presented in Figure 8.

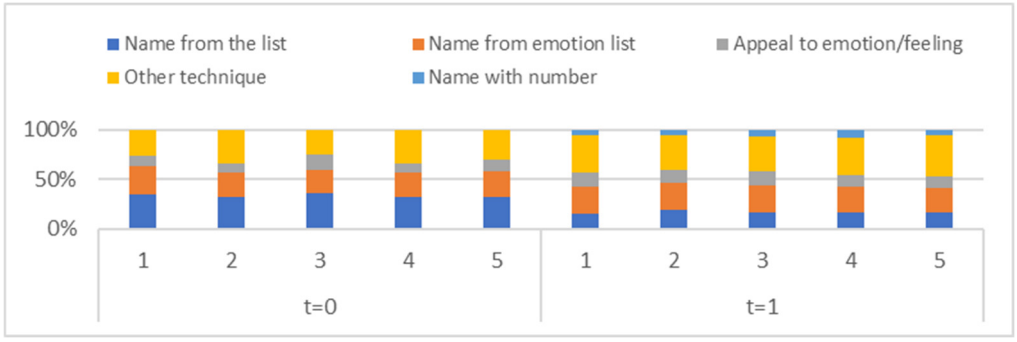


Figure 8. A distribution of label names generated by Strategy II prompt across runs.

Although most of the newly generated labels appeared only once, the model predominantly relied on the technique names explicitly listed in the prompt, with emotion names forming the second most frequent group (Figure 9). Alternative naming patterns were rare at $t = 0$, but became more prevalent at $t = 1$, indicating greater variability at higher temperature settings.

 doi:10.20944/preprints202605.1695.v1

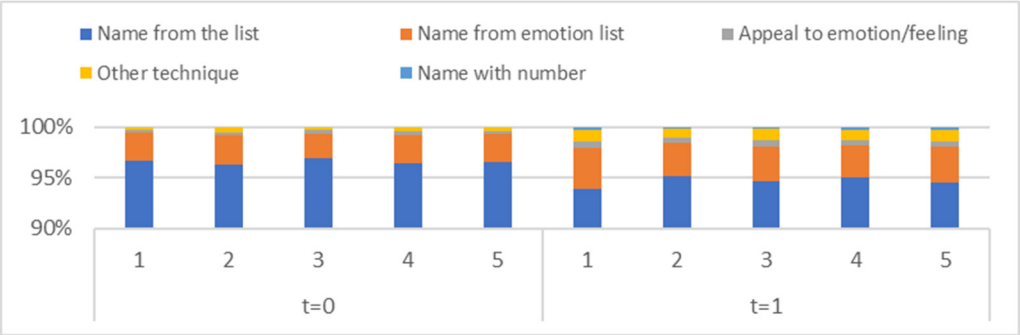


Figure 9. A distribution of the assigned labels generated by Strategy II prompt across runs.

For use in annotation or downstream detection tasks, the instability of the generated labels necessitates a clear strategy for managing variability. In this study, most labels of Strategy II prompts were manually normalized: categories such as *Other techniques* and *Names with number* were eliminated, and the number of fragments assigned to label groups derived from the emotion list or *Appeal to* construction was reduced. Only spans for which a standardized technique label could be assigned were included in the accuracy evaluation.

5.2. Performance of Prompts

This subsection provides the empirical answer to the research question RQ1 by evaluating the extraction performance of each prompting strategy using the metrics described in Section 4.2.

5.2.1. Performance in Accordance with the Symbol-Based Match Method

The averaged evaluation scores for Strategies II–IV across all techniques and five runs are shown in Figure 10. Strategy I is not included, as technique-specific metrics could not be computed due to the lack of normalized labels.

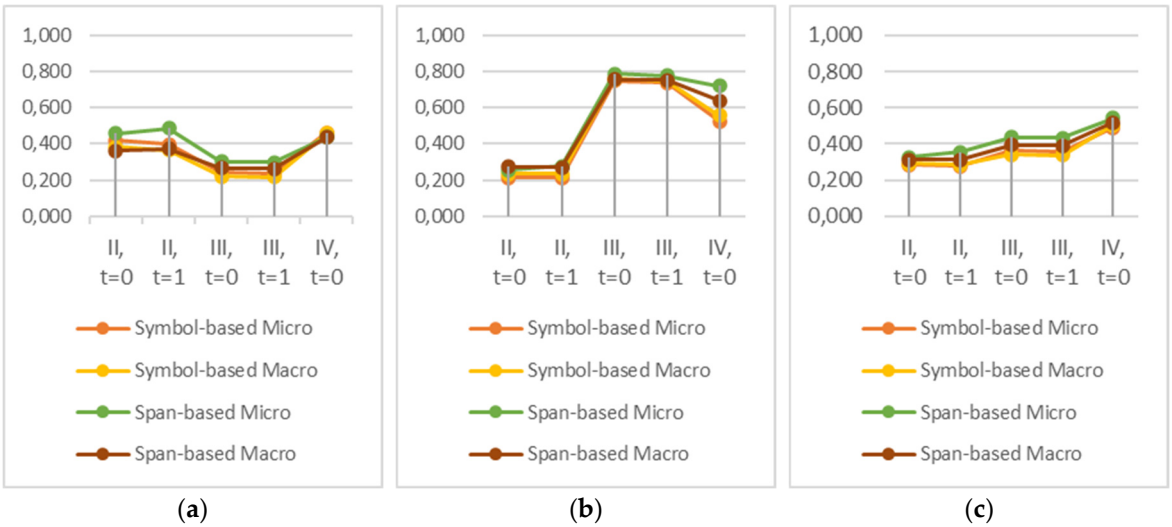


Figure 10. Averaged the prompts' performance evaluation: (**a**) Precision; (**b**) Recall; (**c**) *F1*-score.

Overall, when evaluated using the symbol-based method, Strategy IV achieved the highest precision. Under span-based matching, Strategy II demonstrated the best precision at the micro-level, whereas Strategy III consistently produced the lowest precision across both evaluation methods. Nonetheless, Strategy III obtained the highest recall, while Strategy II yielded the lowest. Although Strategy IV's recall was slightly weaker than that of Strategy III, it remained strong enough for Strategy IV to achieve competitive *F1*-scores. Temperature had only a minor influence on performance, with slightly reduced scores at $t = 1$.

 doi:10.20944/preprints202605.1695.v1

At the level of individual techniques, the impact of prompting strategies varied. Charts (a) of Figures 11–13 represent the symbol-based results for Strategies II–IV at $t = 0$. Results at $t = 1$ are omitted due to space constraints, because differences are minimal. Strategy IV outperformed Strategy III in precision for all techniques and outperformed Strategy II for nine techniques. Strategy II performed best for *Doubt* (*Db*) and *Slogans* (*Sl*).

In terms of recall, Strategy III was the strongest except for the *Support* technique. Strategy IV performed almost identically to Strategy III for *Name-calling* (*NC*), *Labelling* (*Lb*), *Reductio ad Hitlerum* (*RH*), and *Sl*, and surpassed Strategy II for all techniques except *Bandwagon* (*Bw*).

Based on *F1*-scores, Strategy IV improved detection performance for 10 out of 14 techniques. Slightly lower scores were observed for *NC*, *Appeal to Fear* (*AF*), and *Loaded language* (*LL*) compared with one of the other strategies.

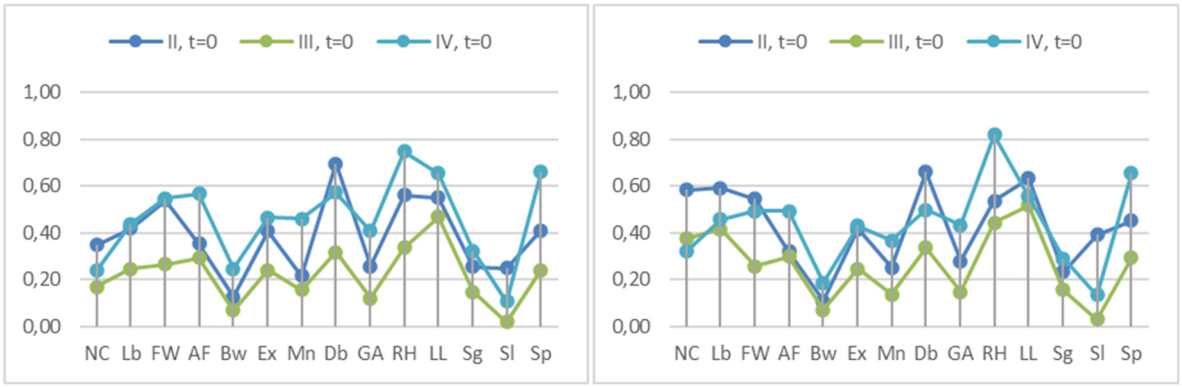


Figure 11. Precision of prompts for each technique, calculated using (**a**) symbol-based matching and (**b**) span-based matching.

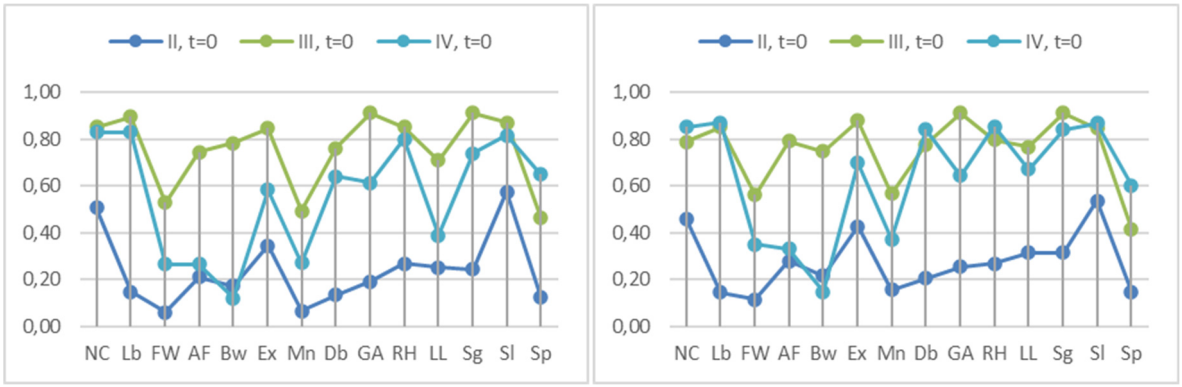
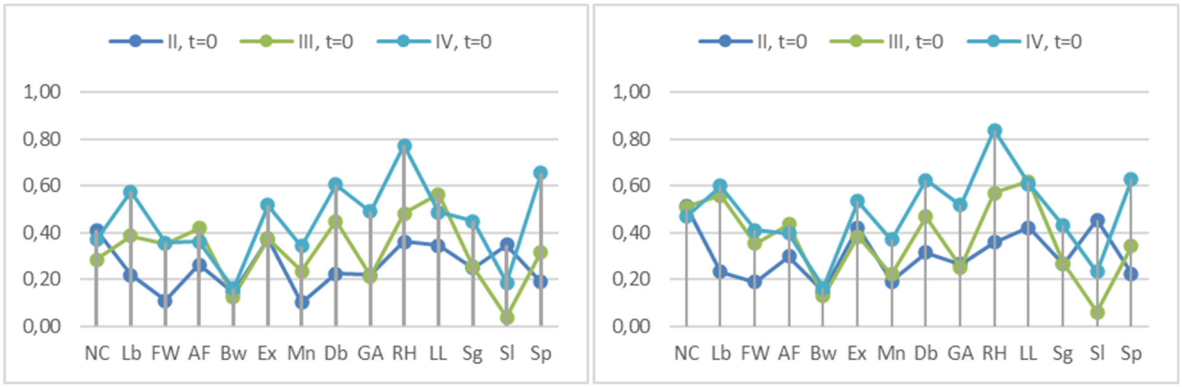


Figure 12. Recall of prompts for each technique, calculated using (**a**) symbol-based matching and (**b**) span-based matching.



 doi:10.20944/preprints202605.1695.v1

(**a**) (**b**)

Figure 13. F-1 score of prompts for each technique, calculated using (**a**) symbol-based matching and (**b**) span-based matching.

5.2.2. Performance in Accordance with the Span-Based Match Method

When calculating the prompt performance metrics using the span-based match method, the ranking of the prompting strategies shifted (see charts (b) of Figures 11-13). This evaluation method generally yielded higher precision values, though the extent of improvement differed by technique. The precision of the *NC* prompt under Strategy IV increased slightly when calculated using this method; however, this strategy ranked lowest, as the precision gains of Strategies II and III were larger. Under span-based matching, Strategy II achieved the highest precision for *Lb*, *Flag waving* (*FW*), and *LL*. Strategy IV's precision decreased for *Scapegoating* (*Sg*), and Strategy III remained the weakest performer overall. The recall values under binary matching showed little change; however, Strategy III was no longer the leading approach. Strategy IV achieved higher recall for *NC*, *Lb*, *Db*, *RH*, and *Sl*. Judging by the span-based *F1*-score, Strategy IV improved the detection performance of most techniques.

5.3. Results with Other LLMs

With strategy IV being most evolved, we have tested its prompts with several other mainstream LLMs: GPT-5.4, Gemini 2.5, Claude Sonnet 4.5 and DeepSeek-V3.2. The prompts were run at a temperature of 0. The results are illustrated by the plot of F1-scores across prompts of different techniques and LLMs, shown in Figure 14. The thick dashed line represents baseline results produced by GPT-4.1. Other mainstream LLMs, including a later version of GPT, mostly do not fare better than the baseline model and, in many cases, fare quite significantly worse.

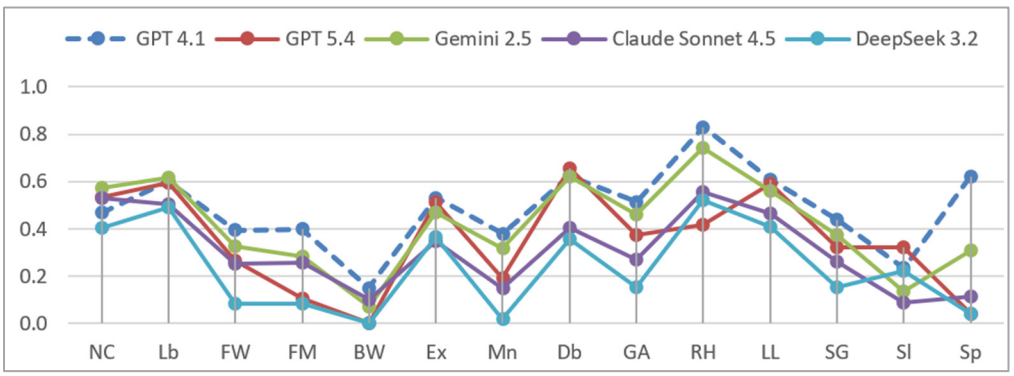


Figure 14. F1-scores for Strategy IV across different techniques and LLMs.

5.4. Evaluation of Practical Utility

This subsection addresses the research question RQ2 by assessing the practical suitability of each prompting strategy. The performance metrics indicate that Strategy IV, being the most evolved, improves the detection of many technique representations. However, for some prompts, the improvement is still insufficient to render them suitable for practical use. While Strategy IV substantially increased precision — sometimes by several multiples (e.g., *Bw*, *Sl*)—some techniques still display low absolute values. This indicates that further refinement of the prompt is required before it can be reliably used for large-scale annotation.

Following the methodology described in Section 4.3, we have calculated the practical-utility metrics for the Strategy II–IV prompts at $t = 0$ and assessed their overall suitability for annotation. The results are presented in Table 3, which compares Strategies II-IV, showing the extent to which each strategy improved or worsened performance metrics relative to the other strategy. The relative

 doi:10.20944/preprints202605.1695.v1

change is calculated by applying $vs(a,b) = \{a/b, if\ a \geq b;\ -b/a, otherwise\}$. Positive values show relative improvement; negative values show relative degradation.

Table 3. Comparison of performance metrics for strategy II-IV prompts.

Technique	*Precision*				*Recall*				*F1*-score			
	Relative change			IV	Relative change			IV	Relative change			IV
	III vs II	IV vs II	IV vs III		III vs II	IV vs II	IV vs III		III vs II	IV vs II	IV vs III	
NC	-2,03	-1,46	1,39	0,24	1,69	1,64	-1,03	0,83	-1,45	-1,11	1,30	0,37
Lb	-1,71	1,05	1,79	0,44	6,01	5,56	-1,08	0,83	1,75	2,61	1,49	0,57
FW	-2,04	1,02	2,07	0,55	8,65	4,33	-2,00	0,26	3,21	3,25	1,01	0,36
AF	-1,21	1,60	1,94	0,57	3,54	1,26	-2,81	0,26	1,59	1,37	-1,16	0,36
Bw	-1,84	1,94	3,57	0,25	4,58	-1,44	-6,58	0,12	-1,15	1,10	1,27	0,16
Ex	-1,69	1,14	1,93	0,47	2,46	1,69	-1,45	0,58	1,01	1,38	1,38	0,52
Mn	-1,40	2,12	2,97	0,46	7,46	4,12	-1,81	0,27	2,33	3,37	1,45	0,34
Db	-2,19	-1,22	1,80	0,57	5,66	4,75	-1,19	0,64	1,99	2,68	1,35	0,60
GA	-2,11	1,59	3,35	0,41	4,78	3,22	-1,48	0,61	-1,02	2,24	2,28	0,49
RH	-1,67	1,33	2,23	0,75	3,19	2,99	-1,07	0,80	1,33	2,13	1,60	0,77
LL	-1,18	1,19	1,40	0,65	2,82	1,55	-1,82	0,39	1,63	1,41	-1,15	0,49
Sg	-1,72	1,27	2,19	0,32	3,72	3,02	-1,23	0,74	1,02	1,80	1,77	0,45
Sl	-12,98	-2,37	5,48	0,11	1,51	1,42	-1,06	0,82	-9,24	-1,86	4,96	0,19
Sp	-1,71	1,60	2,74	0,66	3,80	5,32	1,40	0,65	1,68	3,48	2,06	0,65

As can be seen in Table 4, across strategies, TP_{sp} values show improvement, but for most techniques, they remain below 25%. Thus, even with Strategy IV, GPT-4.1 produces only a limited proportion of fully correct spans, and most outputs still require correction or removal. Compared to Strategy III, FP_{sp} values likewise are better, except for *Name-calling*. The number of completely incorrect spans $F1_{sm} = 0$ decreased substantially, indicating a reduced burden of eliminating irrelevant fragments.

Although Strategy IV increased FN_{sp} for several techniques relative to Strategy III, it yielded markedly fewer undetected spans than Strategy II. The most notable gains in Strategy IV occurred for *RH*, where the $F1$'-score exceeded 0.5, indicating that the prompt is genuinely viable for annotation. Substantial improvements were observed for *Lb* and *Sp*. However, the *Lb* prompt generates considerable "noise", while the *Sp* prompt fails to detect one-third of the relevant spans, and one-third needs to be corrected.

Table 4. Results of technique-specific prompt evaluations (best results in bold).

Technique	Strategy	Percentage of Spans					*F1′*
		TP_{sp}		FP_{sp}		FN_{sp}	
		$F1_{sm}$=1	$F1_{sm}$<1	$F1_{sm}$>0	$F1_{sm}$=0		
Name-calling (NC)	II	13.8%	20.8%	**0.2%**	**24.6%**	40.6%	0.242
	III	11.0%	**23.1%**	0.3%	56.6%	9.1%	0.198
	IV	**18.5%**	11.9%	1.3%	62.9%	**5.3%**	**0.313**
Labelling (Lb)	II	5.6%	7.6%	**0.0%**	**9.1%**	77.7%	0.106
	III	14.5%	**24.2%**	0.4%	54.1%	6.8%	0.253
	IV	**30.1%**	12.6%	1.6%	49.4%	**6.4%**	**0.463**
Flag-waving (FW)	II	0.4%	10.2%	**0.7%**	**8.0%**	80.7%	0,007
	III	4.9%	**16.5%**	2.7%	59.1%	**16.8%**	0,093
	IV	**14.1%**	11.5%	2.4%	24.1%	47.9%	**0,247**
Appeal to fear (AF)	II	3.5%	14.1%	**1.3%**	36.0%	45.0%	0.068
	III	5.4%	**23.3%**	3.2%	61.7%	**7.4%**	0.102
	IV	**10.7%**	14.2%	3.4%	**21.8%**	49.8%	**0.194**
Bandwagon (Bw)	II	1.6%	**6.3%**	**0.0%**	63.5%	28.6%	0.031

 doi:10.20944/preprints202605.1695.v1

	III	1.6%	5.2%	**0.0%**	90.8%	**2.4%**	0.031
	IV	**2.6%**	5.3%	**0.0%**	**39.5%**	52.6%	**0.051**
Exaggeration (Ex)	II	7.0%	**19.7%**	**1.1%**	**36.3%**	35.8%	0.131
	III	5.7%	18.1%	2.5%	70.4%	3.3%	0.107
	IV	**22.4%**	13.9%	3.5%	44.5%	15.7%	**0.366**
Minimization (Mn)	II	0.4%	10.3%	**0.0%**	**31.5%**	57.5%	0.014
	III	5.3%	7.3%	2.9%	75.1%	**9.5%**	0.101
	IV	**10.3%**	**12.1%**	6.1%	33.3%	38.2%	**0.187**
Doubt (Db)	II	3.5%	16.0%	**0.3%**	**8.3%**	71.8%	0.068
	III	7.1%	23.8%	3.5%	57.0%	**8.5%**	0.133
	IV	**17.3%**	**28.1%**	9.4%	36.4%	8.7%	**0.295**
Guilt by association (GA)	II	4.3%	11.1%	1.0%	**38.6%**	44.9%	0.083
	III	3.7%	10.8%	0.6%	83.5%	**1.4%**	0.071
	IV	**22.1%**	**12.6%**	**0.4%**	45.5%	19.5%	**0.362**
Reductio ad Hitlerum (RH)	II	1.5%	20.6%	4.4%	14.7%	58.8%	0.029
	III	5.5%	**34.5%**	1.8%	48.2%	**10.0%**	0.103
	IV	**40.0%**	32.3%	**1.5%**	**13.8%**	12.3%	**0.571**
Loaded language (LL)	II	7.2%	19.5%	**0.6%**	**14.9%**	57.8%	0.134
	III	12.3%	**32.3%**	5.2%	36.6%	**13.6%**	0.220
	IV	**16.4%**	27.2%	8.1%	26.8%	21.5%	**0.282**
Scapegoating (Sg)	II	4.0%	11.4%	1.2%	**49.9%**	33.6%	0.076
	III	3.3%	**12.1%**	**1.1%**	82.0%	**1.5%**	0.064
	IV	**15.6%**	12.0%	2.5%	64.9%	5.1%	**0.269**
Slogans (Sl)	II	**10.8%**	**18.1%**	**0.0%**	**45.8%**	25.3%	**0.196**
	III	1.4%	1.6%	**0.0%**	96.4%	**0.6%**	0.029
	IV	7.8%	5.4%	0.3%	84.4%	2.0%	0.145
Support (Sp)	II	3.6%	9.1%	**0.9%**	**14.5%**	71.8%	0.070
	III	5.4%	15.6%	2.2%	47.8%	29.0%	0.102
	IV	**12.2%**	**33.3%**	7.3%	17.1%	**30.1%**	**0.217**

For *Ex*, *Db*, *GA*, and *Sg*, precision improved, and the proportion of FP spans decreased, though residual noise levels remained high. By contrast, the *Bw* prompt remains ineffective: despite high gains in symbol-based precision over Strategies II and III, absolute precision remains too low for practical use, and recall did not improve at all. Similarly, the *Sl* prompt remains unsuitable: while recall is high, the model systematically over-detects spans. The *NC* prompt shows moderate *F*1' performance yet still requires refinement to reduce irrelevant outputs.

For *FW* and *AF*, Strategy IV doubled precision relative to Strategy III, but recall remained low. Although Strategy IV improved precision for *Minimization* (*Mn)* by several multiples, recall remains insufficient, many relevant spans are missed, and numerous FP are generated. Detection performance for *LL* showed little improvement. The prompt identifies many fragments that need to be corrected and fails to detect ~20% of the actual representations.

6. Conclusions and Discussion

6.1. Summary

We presented a systematic investigation of an incremental LLM prompt engineering approach for the purposes of detecting occurrences of emotional manipulation in Lithuanian language texts. The main part of work was done on the basis of GPT-4.1. We have also tested the final prompt engineering results with other contemporary mainstream LLMs: GPT-5.4, Gemini 2.5, Claude Sonnet 4.5 and DeepSeek-V3.2.

The GPT-4.1 based results demonstrate that prompt formulation shapes output stability and model performance. The Strategy I prompt exhibits substantial instability which makes it unsuitable

 doi:10.20944/preprints202605.1695.v1

for automating text annotation. Strategy II prompt, incorporating explicit lists of techniques and emotions, exhibits reduction of label variability by 97% relative to Strategy I. However, it maintains high result instability, suggesting that multi-label tasks with limited contextual guidance are insufficient for reliable detection. Restricting the per-prompt task to a single technique and providing richer contextual guidance significantly improves consistency. Results of Strategies III and IV do not exhibit the multi-label variability found in results of Strategies I and II. Yet the results of Strategy III lacked stability and precision, necessitating the development of Strategy IV featuring technique-specific definitions and explicit extraction rules. Strategy IV further enhanced result stability and improved precision across most techniques. Nonetheless, performance gains across techniques were not uniform, suggesting that further prompt refinement should be explicitly tailored to each separate technique.

Prompt temperature was found to strongly influence stability of LLM result. Prompts executed at temperature 1 produced markedly greater output variability, more deviations from task instructions, and a higher proportion of non-reproducible spans. This suggests that for purposes of emotional manipulation detection, it is best to run prompts at temperature of 0.

All the mainstream LLMs tested with Strategy IV prompts displayed results that were in a small number of techniques slightly better than the baseline produced by GPT-4.1 but for most techniques results were worse. The trends between techniques are closely following the trends displayed by the baseline LLM. This indicates that the results of our research are transferable between different LLMs.

6.2. Theoretical Implications

Our findings reinforce the prevailing conclusion that LLM performance in fine-grained manipulation span detection is highly sensitive to prompt structure and contextual scaffolding. Consistent with earlier work [41,54], the study shows that LLMs do not function as stable classifiers. The outputs vary across repeated executions of identical prompts and suggest interactions among instruction framing, contextual cues, and model internal priors. This instability reinforces the need for theoretical models that explicitly account for stochasticity and instruction sensitivity in LLM reasoning [17,56].

The results expose limitations of multi-label prompting, which often leads to semantic drift in label generation [35,42]. Single-technique prompts with explicit constraints appear to be more robust in eliciting consistent span-level behavior. At the same time, the issues with manipulation techniques having diffuse or overlapping semantic boundaries suggest that some technique definitions and their operational annotation instructions require refinement. Better conceptual distinctions and more precise annotation criteria could enhance both human and LLM-based labelling, particularly in tasks where rhetorical categories are ambiguous.

6.3. Practical Implications

The evaluation framework used in our work combines stability metrics, symbol- and span-based matching, and a custom measure of practical utility for assessing both the amount of useful work the model performs and the manual annotation burden it adds. This offers a method for assessing whether prompts would meaningfully reduce annotator workload, addressing challenges noted in propaganda span detection [35,40]. Applying our framework in comparative analyses of prompts targeting the same technique allows informed discrimination of the effect of specific prompt modifications and guides further refinement.

Currently, mainstream LLMs were found to be generally unsuitable for automatic annotation of emotional manipulation techniques in a low-resource language (like Lithuanian) corpus. The results for most techniques are of low quality and therefore cannot be trusted. *Reductio ad Hitlerum* technique, having the best *F1*-score at 0.82, would be a lone exception for a pre-annotation task if used in a workflow like one described in ref. [10]. This is consistent with findings from other multilingual evaluations [26,57]. These insights are directly applicable to the development of misinformation detection corpora, computational social science datasets, and automated content moderation tools.

 doi:10.20944/preprints202605.1695.v1

6.4. Limitations and Future Work

Our dataset consists solely of Lithuanian news portal comments, restricting the applicability of the results to other domains or languages. The multi-stage, multi-run methodology suggested in this study increases API operational costs and computational resource requirements, which may pose a barrier to large-scale applications.

Our decision to use temperature 0 setting only for Strategy IV constitutes a practical limitation of this study. As a result, potential instability effects for Strategy IV were not empirically verified within the current experimental setup.

Future work should explore cross-lingual transferability. Incorporating self-verification or multi-pass prompting techniques may further stabilize outputs. Attention should be given to techniques that consistently exhibit weak performance, such as *Bandwagon* and *Slogans*, to better understand the linguistic or conceptual factors underlying their low detectability in the Lithuanian context.

Author Contributions: Conceptualization, R.B. (Rita Butkienė) and L.A.; methodology, R.B. (Rimantas Butleris); software, A.Š.; validation, E.V. and P.D.; formal analysis, L.A.; investigation, R.B. (Rita Butkienė); resources, A.Š. and E.D.; data curation, R.B. (Rita Butkienė), A.Š., E.D., V.Ž; writing—original draft preparation, R.B. (Rita Butkienė); writing—review and editing, L.A., E.D., E.V., and P.D.; visualization, A.Š.; supervision, R.B. (Rimantas Butleris); project administration, R.B. (Rimantas Butleris); funding acquisition, R.B. (Rimantas Butleris). All authors have read and agreed to the published version of the manuscript.

Funding: This work was conducted as part of the execution of the project "Mission-driven Implementation of Science and Innovation Programs" (No. 02-002-P-0001), funded by the Economic Revitalization and Resilience Enhancement Plan "New Generation Lithuania".

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data and code used in the study are openly available in the *Zenodo* repository at DOI: 10.5281/zenodo.20053557.

Conflicts of Interest: The authors declare no conflicts of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript:

GPT	Generative Pre-trained Transformer
FN	False negative
FP	False positive
LLM	Large language model
NLP	Natural language processing
TP	True positive

References

1. Pratkanis, A.R.; Aronson, E. Age of Propaganda: The Everyday use and Abuse of Persuasion; Macmillan, 2001.
2. Shabo, M. Techniques of Propaganda and Persuasion; Prestwick House Inc, 2008.
3. Barrón-Cedeño, A.; Jaradat, I.; Da San Martino, G.; Nakov, P. Proppy: Organizing the news based on their propagandistic content. *Information Processing & Management* **2019**, *56*, 1849–1864, DOI:10.1016/j.ipm.2019.03.005.

 doi:10.20944/preprints202605.1695.v1

4. Da San Martino, G.; Shaar, S.; Zhang, Y.; Yu, S.; Barrón-Cedeno, A.; Nakov, P. Prta: A system to support the analysis of propaganda techniques in the news. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations* **2020**, 287–293, DOI:10.18653/v1/2020.acl-demos.32.
5. Zavolokina, L.; Sprenkamp, K.; Katashinskaya, Z.; Jones, D.G.; Schwabe, G. Think Fast, Think Slow, Think Critical: Designing an Automated Propaganda Detection Tool. *Proceedings of the CHI Conference on Human Factors in Computing Systems* **2024**, 1–24, DOI:10.1145/3613904.3642805.
6. Zhou, C.; Li, K.; Lu, Y. Linguistic characteristics and the dissemination of misinformation in social media: The moderating effect of information richness. *Information Processing & Management* **2021**, *58*, 102679, DOI:10.1016/j.ipm.2021.102679.
7. Liu, J.; Ai, L.; Liu, Z.; Karisani, P.; Hui, Z.; Fung, Y.; Nakov, P.; Hirschberg, J.; Ji, H. Propainsight: Toward deeper understanding of propaganda in terms of techniques, appeals, and intent **2025**, 5607–5628.
8. Gao, Y.; Gao, P.; Bao, H.; Li, B.; Luo, J.; Wang, Z.; Chen, W. Boosting Large Language Models for Mental Manipulation Detection via Data Augmentation and Distillation **2026**, 9033–9043.
9. Barfar, A.; Sommerfeldt, L. Propaganda by prompt: Tracing hidden linguistic strategies in large language models. *Information Processing & Management* **2026**, *63*, 104403, DOI:10.1016/j.ipm.2025.104403.
10. Sahitaj, A.; Sahitaj, P.; Solopova, V.; Li, J.; Möller, S.; Schmitt, V. Hybrid annotation for propaganda detection: integrating LLM pre-annotations with human intelligence. *Proceedings of the Fourth Workshop on NLP for Positive Impact (NLP4PI)* **2025**, 215–228, DOI:10.18653/v1/2025.nlp4pi-1.18.
11. Srivastava, S.; Pati, S.; Yao, Z. Instruction-tuning LLMs for event extraction with annotation guidelines **2025**, 13055–13071.
12. Jang, E.H.; Aguirre, J.; Lee, S.; Moon, H.; Cha, W.C. Span-based annotation framework for LLM-based clinical named entity recognition: development and validation using Korean emergency department notes. *JAMIA open* **2025**, *8*, ooaf157.
13. Bojić, L.; Zagovora, O.; Zelenkauskaite, A.; Vuković, V.; Čabarkapa, M.; Veseljević Jerković, S.; Jovančević, A. Comparing large Language models and human annotators in latent content analysis of sentiment, political leaning, emotional intensity and sarcasm. *Scientific reports* **2025**, *15*, 11477.
14. Jadhav, S.; Shanbhag, A.; Thakurdesai, A.; Sinare, R.; Joshi, R. On limitations of LLM as annotator for low resource languages **2025**, 277–282.
15. Constantin Hellwig, N.; Fehle, J.; Kruschwitz, U.; Wolff, C. Do we still need Human Annotators? Prompting Large Language Models for Aspect Sentiment Quad Prediction. *arXiv e-prints* **2025**, arXiv: 2502.13044.
16. Huang, K.; Chan, H.P.; McKeown, K.; Ji, H. ManiTweet: A new benchmark for identifying manipulation of news on social media. *Proceedings of the 31st International Conference on Computational Linguistics* **2025**, 11161–11180.
17. Wang, B.; Liu, Z.; Huang, X.; Jiao, F.; Ding, Y.; Aw, A.; Chen, N. Seaeval for multilingual foundation models: From cross-lingual alignment to cultural reasoning. *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)* **2024**, 370–390, DOI:10.18653/v1/2024.naacl-long.22.
18. Razavi, A.; Soltangheis, M.; Arabzadeh, N.; Salamat, S.; Zihayat, M.; Bagheri, E. Benchmarking Prompt Sensitivity in Large Language Models. *: Hauff, C.*, et al. *Advances in Information Retrieval. ECIR 2025. Lecture Notes in Computer Science* **2025**, *15574*, 303–313, DOI:10.1007/978-3-031-88714-7_29.
19. Da San Martino, G.; Barrón-Cede Ño, A.; Wachsmuth, H.; Petrov, R.; Nakov, P. SemEval-2020 Task 11: Detection of Propaganda Techniques in News Articles. *Proceedings of the Fourteenth Workshop on Semantic Evaluation* **2020**, 1377–1414, DOI:10.18653/v1/2020.semeval-1.186.
20. Dimitrov, D.; Bin Ali, B.; Shaar, S.; Alam, F.; Silvestri, F.; Firooz, H.; Nakov, P.; Da San Martino, G. SemEval-2021 Task 6: Detection of Persuasion Techniques in Texts and Images. *Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021)* **2021**, 70–98, DOI:10.18653/v1/2021.semeval-1.7.
21. Piskorski, J.; Stefanovitch, N.; Da San Martino, G.; Nakov, P. Semeval-2023 task 3: Detecting the category, the framing, and the persuasion techniques in online news in a multi-lingual setup. *Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023)* **2023**, 2343–2361, DOI:10.18653/v1/2023.semeval-1.317.

 doi:10.20944/preprints202605.1695.v1

22. Dimitrov, D.; Alam, F.; Hasanain, M.; Hasnat, A.; Silvestri, F.; Nakov, P.; Da San Martino, G. Semeval-2024 task 4: Multilingual detection of persuasion techniques in memes. *Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024)* **2024**, 2009–2026, DOI:10.18653/v1/2024.semeval-1.275.
23. Alam, F.; Mubarak, H.; Zaghouani, W.; Da San Martino, G.; Nakov, P. Overview of the WANLP 2022 shared task on propaganda detection in Arabic. *Proceedings of the Seventh Arabic Natural Language Processing Workshop (WANLP)* **2022**, 108–118, DOI:10.18653/v1/2022.wanlp-1.11.
24. Hasanain, M.; Alam, F.; Mubarak, H.; Abdaljalil, S.; Zaghouani, W.; Nakov, P.; Da San Martino, G.; Freihat, A. ArAIEval shared task: Persuasion techniques and disinformation detection in Arabic text. *Proceedings of ArabicNLP 2023* **2023**, 483–493, DOI:10.18653/v1/2023.arabicnlp-1.44.
25. Zaghouani, W.; Jarrar, M.; Habash, N.; Bouamor, H.; Zitouni, I.; Diab, M.; El-Beltagy, S.R.; AbuOdeh, M. The FIGNEWS shared task on news media narratives. *Proceedings of the Second Arabic Natural Language Processing Conference* **2024**, 530–547, DOI:10.18653/v1/2024.arabicnlp-1.56.
26. Piskorski, J.; Jorge, A.; Silvano, M.d.P.; Guimarães, N.; Pacheco, A.F.; Yu, N. Overview of the CLEF-2024 checkthat! lab task 3 on persuasion techniques. *CEUR Workshop Proceedings* **2024**, 299–310.
27. Piskorski, J.; Dimitrov, D.; Dobranić, F.; Ernst, M.; Haneczok, J.; Koychev, I.; Ljubešić, N.; Marcińczuk, M.; Modzelewski, A.; Moravski, I. SlavicNLP 2025 Shared Task: Detection and Classification of Persuasion Techniques in Parliamentary Debates and Social Media. *Proceedings of the 10th Workshop on Slavic Natural Language Processing (Slavic NLP 2025)* **2025**, 254–275, DOI:10.18653/v1/2025.bsnlp-1.27.
28. Kyslyi, R.; Romanyshyn, N.; Sydorskyi, V. The UNLP 2025 shared task on detecting social media manipulation. *Proceedings of the Fourth Ukrainian Natural Language Processing Workshop (UNLP 2025)* **2025**, 105–111, DOI:10.18653/v1/2025.unlp-1.12.
29. Chang, R.; Lai, C.; Chang, K.; Lin, C. Dataset of propaganda techniques of the state-sponsored information operation of the People's Republic of China **2021**, DOI:10.48550/arXiv.2106.07544.
30. Salman, M.; Hanif, A.; Shehata, S.; Nakov, P. Detecting propaganda techniques in code-switched social media text. *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing* **2023**, 16794–16812, DOI:10.18653/v1/2023.emnlp-main.1044.
31. Horák, A.; Sabol, R.; Herman, O.; Baisa, V. Recognition of propaganda techniques in newspaper texts: Fusion of content and style analysis. *Expert Systems with Applications* **2024**, *251*, 124085, DOI:10.1016/j.eswa.2024.124085.
32. Rizgelienė, I.; Zubaitienė, V.; Maliukevičius, N.; Marcinkevičius, V. HALT-PROP: Human-Annotated Lithuanian Textual Corpus for Propaganda Narratives and Techniques. *Sci Data* **2025**, *13*, 47, DOI:10.1038/s41597-025-06367-w.
33. Al-Henaki, L.; Al-Khalifa, H.; Al-Salman, A.; Alqubayshi, H.; Al-Twailay, H.; Alghamdi, G.; Aljasim, H. Multiprose: A multi-label Arabic dataset for propaganda, sentiment, and emotion detection. *In: Ichise, R. (eds) Natural Language Processing and Information Systems. NLDB 2025. Lecture Notes in Computer Science* **2026**, *15836*, 156–172, DOI:10.1007/978-3-031-97141-9_11.
34. Da San Martino, G.; Yu, S.; Barrón-Cedeno, A.; Petrov, R.; Nakov, P. Fine-grained analysis of propaganda in news articles. Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP) **2019**, 5636–5646, DOI:10.18653/v1/D19-1565.
35. Hasanain, M.; Ahmed, F.; Alam, F. Can GPT-4 Identify Propaganda? Annotation and Detection of Propaganda Spans in News Articles. Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024) **2024**, 2724–2744.
36. Chernyavskiy, A.; Shomova, S.; Dushakova, I.; Kiriya, I.; Ilvovsky, D. ZenPropaganda: A comprehensive study on identifying propaganda techniques in Russian coronavirus-related media. *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)* **2024**, 17795–17807.
37. Ahmad, P.N.; Shah, A.M.; Guo, J.; Liu, Y. ProST: spotting propaganda span and technique classification in news articles. *Aslib Journal of Information Management* **2025**, DOI:10.1108/AJIM-08-2024-0660.

 doi:10.20944/preprints202605.1695.v1

38. Chernyavskiy, A.; Ilvovsky, D.; Nakov, P. Unleashing the power of discourse-enhanced transformers for propaganda detection. Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers) **2024**, 1452–1462, DOI:10.18653/v1/2024.eacl-long.87.
39. Haider, S.; Luceri, L.; Deb, A.; Badawy, A.; Peng, N.; Ferrara, E. Detecting social media manipulation in low-resource languages. *Companion Proceedings of the ACM Web Conference 2023* **2023**, 1358–1364, DOI:10.1145/3543873.3587615.
40. Sprenkamp, K.; Jones, D.G.; Zavolokina, L. Large Language Models for Propaganda Detection. *arXiv preprint* **2023**, DOI:10.48550/arXiv.2310.06422.
41. Szwoch, J.; Staszkow, M.; Rzepka, R.; Araki, K. Limitations of Large Language Models in Propaganda Detection Task. *Applied Sciences* **2024**, *14*, 4330, DOI:10.3390/app14104330.
42. Gaeta, A.; Loia, V.; Lorusso, A.; Orciuoli, F.; Pascuzzo, A. Towards a LLM-based intelligent system for detecting propaganda within textual content. *Computers and Electrical Engineering* **2025**, *128, Part B*, 110765, DOI:10.1016/j.compeleceng.2025.110765.
43. Jose, J.; Greenstadt, R. Are Large Language Models Good at Detecting Propaganda? **2025**, DOI:10.48550/arXiv.2505.13706.
44. Hasanain, M.; Ahmad, F.; Alam, F. Large Language Models for Propaganda Span Annotation. *Findings of the Association for Computational Linguistics: EMNLP 2024* **2024**, 14522–14532, DOI:10.18653/v1/2024.findings-emnlp.850.
45. Hamilton, K.; Longo, L.; Bozic, B. GPT Assisted Annotation of Rhetorical and Linguistic Features for Interpretable Propaganda Technique Detection in News Text. *Companion Proceedings of the ACM Web Conference 2024* **2024**, 1431–1440, DOI:10.1145/3589335.3651909.
46. Maarouf, A.; Bär, D.; Geissler, D.; Feuerriegel, S. HQP: a human-annotated dataset for detecting online propaganda. *Findings of the Association for Computational Linguistics: ACL 2024* **2024**, 6064–6089, DOI:10.18653/v1/2024.findings-acl.363.
47. Sawi, M.; Węcel, K.; Księżniak, E. Multilabel Classification of Persuasion Techniques with self-improving LLM agent: SlavicNLP 2025 Shared Task. *Proceedings of the 10th Workshop on Slavic Natural Language Processing (Slavic NLP 2025)* **2025**, 231–253, DOI:10.18653/v1/2025.bsnlp-1.26.
48. Butkienė, R.; Šukys, A.; Dambrauskas, E.; Žitkus, V.; Ablonskis, L.; Vaičiukynas, E.; Danėnas, P.; Butleris, R. Lithuanian Emotional Manipulation Dataset [Data set] **2026**, DOI:10.5281/zenodo.20053557.
49. Piskorski, J.; Stefanovitch, N.; Bausier, V.; Faggiani, N.; Linge, J.; Kharazi, S.; Nikolaidis, N.; Teodori, G.; De Longueville, B.; Doherty, B.; Gonin, J.; Ignat, C.; Kotseva, B.; Mantica, E.; Marcaletti, L.; Rossi, E.; Spadaro, A.; Verile, M.; Da San Martino, G.; Alam, F.; Nakov, P. News categorization, framing and persuasion techniques: Annotation guidelines. Technical Report JRC-132862. *European Commission Joint Research Centre* **2023**.
50. Krippendorff, K. Content Analysis: An Introduction to its Methodology; Sage publications, 2018.
51. Liu, P.; Yuan, W.; Fu, J.; Jiang, Z.; Hayashi, H.; Neubig, G. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM computing surveys* **2023**, *55*, 1–35, DOI:10.1145/3560815.
52. Atıl, B.; Aykent, S.; Chittams, A.; Fu, L.; Passonneau, R.J.; Radcliffe, E.; Rajagopal, G.R.; Sloan, A.; Tudrej, T.; Türe, F. Non-Determinism of "Deterministic" LLM System Settings in Hosted Environments **2025**, 135–148.
53. Fachada, N.; Fernandes, D.; Fernandes, C.M.; Ferreira-Saraiva, B.D.; Matos-Carvalho, J.P. GPT-4.1 Sets the Standard in Automated Experiment Design Using Novel Python Libraries. *Future Internet* **2025**, *17*, 412, DOI:10.3390/fi17090412.
54. Dong, J.; Zhang, Y.; Liu, Y.; Zhong, Z.; Wei, T.; Zhang, C.; Qiu, H. Revisiting the Reliability of Language Models in Instruction-Following. *arXiv preprint* **2025**, DOI:10.48550/arXiv.2512.14754.
55. Potthast, M.; Barrón-Cedeño, A.; Eiselt, A.; Stein, B.; Rosso, P. Overview of the 2nd international competition on plagiarism detection. *CEUR Workshop Proceedings* **2010**, *1176*, 1–14.

 doi:10.20944/preprints202605.1695.v1

56. Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Ichter, B.; Xia, F.; Zhou, D. Chain of thought prompting elicits reasoning in large language models. *Proceedings of the 36th International Conference on Neural Information Processing Systems (NIPS '22)* **2022**, *2201*, 24824–24837, DOI:10.5555/3600270.3602070.
57. Jose, J.; Greenstadt, R. LLMs for detection and classification of persuasion techniques in Slavic parliamentary debates and social media texts. *Proceedings of the 10th Workshop on Slavic Natural Language Processing (Slavic NLP 2025)* **2025**, 202–216, DOI:10.18653/v1/2025.bsnlp-1.23.