



AFRL-RI-RS-TR-2024-070

**CROSS-LINGUAL QUESTION ANSWERING TO IDENTIFYING
INFORMATION DIFFERENCES BETWEEN ENGLISH AND
RUSSIAN WIKIPEDIA ARTICLES**

UNIVERSITY OF PENNSYLVANIA

JUNE 2024

FINAL TECHNICAL REPORT

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

STINFO COPY

**AIR FORCE RESEARCH LABORATORY
INFORMATION DIRECTORATE**

NOTICE AND SIGNATURE PAGE

Using Government drawings, specifications, or other data included in this document for any purpose other than Government procurement does not in any way obligate the U.S. Government. The fact that the Government formulated or supplied the drawings, specifications, or other data does not license the holder or any other person or corporation; or convey any rights or permission to manufacture, use, or sell any patented invention that may relate to them.

This report is the result of contracted fundamental research deemed exempt from public affairs security and policy review in accordance with SAF/AQR memorandum dated 10 Dec 08 and AFRL/CA policy clarification memorandum dated 16 Jan 09. This report is available to the general public, including foreign nations. Copies may be obtained from the Defense Technical Information Center (DTIC) (<http://www.dtic.mil>).

AFRL-RI-RS-TR-2024-070 HAS BEEN REVIEWED AND IS APPROVED FOR PUBLICATION IN ACCORDANCE WITH ASSIGNED DISTRIBUTION STATEMENT.

FOR THE CHIEF ENGINEER:

/ S /

ALEKSEY V. PANASYUK
Work Unit Manager

/ S /

MATTHEW J. KOCHAN
Technical Advisor
Intelligence Systems Division
Information Directorate

This report is published in the interest of scientific and technical information exchange, and its publication does not constitute the Government's approval or disapproval of its ideas or findings.

REPORT DOCUMENTATION PAGE

1. REPORT DATE		2. REPORT TYPE		3. DATES COVERED	
JUNE 2024		FINAL TECHNICAL REPORT		START DATE	END DATE
				MARCH 2023	MARCH 2024
4. TITLE AND SUBTITLE					
CROSS-LINGUAL QUESTION ANSWERING TO IDENTIFYING INFORMATION DIFFERENCES BETWEEN ENGLISH AND RUSSIAN WIKIPEDIA ARTICLES					
5a. CONTRACT NUMBER		5b. GRANT NUMBER		5c. PROGRAM ELEMENT NUMBER	
FA8750-23-C-0507		N/A		62788F	
5d. PROJECT NUMBER		5e. TASK NUMBER		5f. WORK UNIT NUMBER	
N/A		N/A		R3FE	
6. AUTHOR(S)					
Chris Callison-Burch, Ph.D; Marianna Apidianaki, Ph.D.					
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)				8. PERFORMING ORGANIZATION REPORT NUMBER	
University of Pennsylvania 3451 Walnut St STE 440A Philadelphia PA 19104					
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)			10. SPONSOR/MONITOR'S ACRONYM(S)	11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
Air Force Research Laboratory/RIEA 525 Brooks Road Rome NY 13441-4505			AFRL/RI	AFRL-RI-RS-TR-2024-070	
12. DISTRIBUTION/AVAILABILITY STATEMENT					
Approved for Public Release; Distribution Unlimited. This report is the result of contracted fundamental research deemed exempt from public affairs security and policy review in accordance with SAF/AQR memorandum dated 10 Dec 08 and AFRL/CA policy clarification memorandum dated 16 Jan 09.					
13. SUPPLEMENTARY NOTES					
14. ABSTRACT					
<p>In this project, we developed technologies for uncovering and the root cause of information discrepancies across cultures and languages. This followed the assumption that different historical accounts of major events and entities can lead to friction and misunderstanding between cultures. The main inquiry was into developing a system that reads Russian and English articles from Wikipedia on the same topic, and automatically identifies cultural differences, particularly persuasive language, between them. Importantly, we utilized the power of large language models (LLMs), such as GPT-4, combined with our decomposition of the larger problem of persuasion detection into subtasks, each of which uses a carefully designed prompt. We quantify the amount of persuasion in each article, which allows for various analyses and experiments. Notably, we generate two rankings, one per language, of articles by amount of persuasive content. These rankings match our intuitions on which subjects are meaningful to which cultures. In our development process, we identified limitations in prior datasets for multilingual persuasion detection. Using insights from our system development, we further released a large-scale, broadly scoped synthetic dataset.</p>					
15. SUBJECT TERMS					
Overlap summarization, alternate narrative analysis, semantic alignment structure, overlap metric, cross lingual question answering					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	
a. REPORT	b. ABSTRACT	c. THIS PAGE			
U	U	U	SAR	52	
19a. NAME OF RESPONSIBLE PERSON				19b. PHONE NUMBER (Include area code)	
ALEKSEY V. PANASYUK				N/A	

TABLE OF CONTENTS

List of Figures	ii
List of Tables	iii
1.0 SUMMARY.....	1
2.0 INTRODUCTION	2
3.0 METHODS, ASSUMPTIONS, AND PROCEDURES.....	4
4.0 RESULTS AND DISCUSSION	22
5.0 CONCLUSION.....	34
6.0 REFERENCES	36
APPENDIX A – Publications and Presentations	39
APPENDIX B – Prompts Used	40
APPENDIX C – Additional Figures and Tables.....	42
APPENDIX D – Using Open-Source LLMs for Identify-then-Extract.....	43
LIST OF SYMBOLS, ABBREVIATIONS, AND ACRONYMS	46

LIST OF FIGURES

Figure 1: Top: methodology to generate high-level questions; Bottom: querying with LLMs in Russian and English and comparing crosslingually.....	7
Figure 2: A comparison between two approaches to persuasion technique detection.....	8
Figure 3: The effectiveness of the classifiers after each feature reduction using ANOVA.....	12
Figure 4: Scatterplot for affirmative responses from identify and extract stages.....	13
Figure 5: High-level overview of the PAXQA approach to cross-lingual question answering generation.....	16
Figure 6: Comparison of cross-lingual QA between the English and Russian versions of Wikipedia on Georgy Zhukov.	17
Figure 7: DataDreamer workflow	19
Figure 8: Depiction of a territorial disputes task studied for BorderLines	21
Figure 9: Depiction of the method to compare persuasive language usage across languages.....	23
Figure 10: Rank-biased overlap (RBO) scores, calculated pairwise over rankings	25
Figure 11: A scatterplot with NPF values of Russian and English articles.	27
Figure 12: comparison of prompting behavior between Kani and other frameworks	28
Figure 13: Illustration of comparisons made for the CS metrics.....	30
Figure 14: Formulas for the CS metrics.....	31

LIST OF TABLES

Table 1: The raw counts for each persuasion technique for the SemEval task.....	9
Table 2: F1 of GPT-4 on binary persuasion detection at different confidence thresholds.	10
Table 3: Performance on binary classification task of SemEval	11
Table 4: Sample Model responses, on ‘Fire at the National Museum of Brazil’.....	14
Table 5: Top 20 articles with the most persuasion from Russian and English Wikipedia	24
Table 6: Top 5 and bottom 5 topics (Wikidata P31 instance of) by persuasive content.....	26
Table 7: Excerpted rows from the BorderLines table.....	30
Table 8: system prompts for BorderLines prompt modification experiment	32
Table 9: Results for BorderLines.....	32
Table 10: Responses from GPT-4 from querying on who controls Crimea, varying the strategy and the language	33
Table C-1: The 12 HLQs selected in English and Russian.....	42
Table D-1: GPT-4 and Llama response comparison to questions for the article “Fire at the National Museum of Brazil”.....	44

1.0 SUMMARY

In this project, we developed technologies for uncovering and the root cause of information discrepancies across cultures and languages. This followed the assumption that different historical accounts of major events and entities can lead to friction and misunderstanding between cultures. The main inquiry was into developing a system that reads Russian and English articles from Wikipedia on the same topic, and automatically identifies cultural differences, particularly persuasive language, between them. Importantly, we utilized the power of large language models (LLMs), such as GPT-4, combined with our decomposition of the larger problem of persuasion detection into subtasks, each of which uses a carefully designed prompt. We quantify the amount of persuasion in each article, which allows for various analyses and experiments. Notably, we generate two rankings, one per language, of articles by amount of persuasive content. These rankings match our intuitions on which subjects are meaningful to which cultures. In our development process, we identified limitations in prior datasets for multilingual persuasion detection. Using insights from our system development, we further released a large-scale, broadly scoped synthetic dataset. We have written 2 research papers for this project, which are in review at two conferences: SPIE (defense-oriented) and COLM (Natural Language Processing (NLP)-oriented).

Throughout the year-long duration of the UNCOVER grant, its open-ended nature encouraged our inquiries into related directions. This resulted in 3 publications at top-tier NLP venues. First, given the growing popularity with LLMs, their applications have become increasingly useful, and correspondingly complex. However, existing frameworks impose limitations on customizability and reproducibility. To solve this, we developed Kani: a lightweight, flexible, and model-agnostic open-source framework for building language model applications [1]. In the main inquiry, we utilized Kani throughout to accelerate development and iteration for both prompt engineering, and for different LLM experimentation. Second, we developed PAXQA, a methodology for synthetic cross-lingual question answering (QA) generation, which combines state-of-the-art English question generations and word alignment-informed translation to produce high-quality QA instances at scale [2]. Third, we used territorial disputes to investigate how multilingual LLMs respond inconsistently to the same underlying query, when posed in different languages [3]. We develop BorderLines, a multilingual dataset of 251 disputed territories, each of which has its own geopolitical complexities that are beyond the territory's stakeholders (e.g. Crimea or Taiwan). We also design an evaluation suite for BorderLines, and our benchmark of several LLMs show that all LLMs are susceptible to geopolitical bias and inconsistency across languages. For all three projects, we have released our datasets and open-sourced our code.

2.0 INTRODUCTION

Differences in perspectives are pronounced between cultures. The same events or entities will come to be viewed by different cultures with different viewpoints in terms of what is important or not, how this affects its own members, whether it was overall good or bad, and so on. This is an active process by the historians and politicians of a country, to use persuasive language to advance their narratives. Most straightforwardly, and often studied, is the prevalence of “fake news”, an umbrella term covering deliberate disinformation and misinformation. Countries such as Russia and China have allegedly invested billions into state-sponsored media campaigns into Western societies through platforms such as RT and CGTN respectively [5]. Fake news articles contain a plethora of factual information, that whether true or false, are conveyed through persuasive techniques such as loaded language, appeal to authority, or name calling.

However, it is important to realize that just as persuasion is used extensively in fake news and propaganda, so too is it used in more editorially neutral outlets, such as Wikipedia. More subtle use of persuasion can be just as effective. Furthermore, what is considered as a neutral perspective is subjective across cultures. We proceed with the insight that different cultures are characterized by their usage of different languages. Of particular interest to the broader intelligence community is Russia, whose state-run media aims to manipulate the global geopolitical landscape to its benefit.

In this work, we therefore develop systems for automatically identifying instances of persuasive techniques between English and Russian Wikipedia. We choose Wikipedia, a popular online encyclopedia, because it aims for an editorially neutral perspective; due to authors’ access to different primary sources in their respective languages, distinct perspectives will arise. Given the recent widespread adoption of large language models (LLMs), we show that LLMs can be used as a unified framework to handle techniques from translation, complex question answering and generation, and emotional analysis. We show that while zero-shot baselines with LLMs interpret propaganda differently than humans, we can decompose the persuasion detection into subtasks – each leveraging carefully designed prompts to LLMs, such as GPT-4 [6]. This builds up to our final system, which asks targeted high-level questions to extract emotional content from Russian and English article pairs. On a dataset of 22,046 paired articles, which were selected for the Russian article’s relevance, we use this system to rank articles based on emotional appeal and find that the top ranked articles that concern Russian (and vice versa English) readers concur with our judgment. For example, English Wikipedia highlights Gaza and China, while Russian focuses on Syria and Ukraine. We also verify that our high-level questions are effective, and elicit similar responses from LLMs, in both languages.

We furthermore are motivated by the heavy class imbalance of prior persuasion techniques, and release a synthetic dataset for propaganda detection, with balanced class distributions and broadly scoped topics.

As an open-ended research collaboration, in addition to the primary research thrust, the UNCOVER effort allowed us to explore several related directions, which has led to 3 research publications at top-tier NLP conferences and workshops, and several in-progress submissions. In this FTR, we incorporate text, figures, and results from these papers.

We organize the methodology and refer to this organization (e.g. I.A, and II), as follows:

- I. Uncovering Cultural Differences Across Russian and English Wikipedia (primary)
 - A. High-level QA
 - B. Factual Cross-lingual QA
- II. LLM Interface Development – Kani
 - A. Related Work – DataDreamer
- III. Evaluating Geopolitical Biases in Language Models

Discussed in greatest detail will be I., as this was the primary project. For that the contributions are as follows:

- We develop an LLM-powered system to identify instances of persuasive language in English and Russian texts, which automates insights at scale.
- We find that a baseline approach, which directly asks an LLM to identify persuasion used in a text, results in responses that are over-sensitive and over-confident.
- We propose a novel framework of high-level questioning, which reframes the persuasion detection task into a set of high-level questions (HLQs). A large number of HLQs are LLM-authored and are then filtered down to a small set best aligned to human labels of persuasion. On a binary persuasion detection task, HLQs achieve a 23.5% relative improvement in F1 ($0.751 > 0.608$).
- We study a dataset of 26K Wikipedia subjects, with paired articles in Russian and English. To handle the large-scale, we adopt a two-stage identify-then-extract prompting approach with HLQs, reducing inference costs by 85.2%.
- Using our approach and the results, we perform various quantified experiments into Wikipedia’s cross-cultural differences in perspective.
- We propose a bilingual synthetic propaganda dataset that prioritizes nation-state propaganda, incorporating examples across numerous politically relevant categories. This dataset addresses the imbalance present in existing resources, providing around 10K examples for each technique.

3.0 METHODS, ASSUMPTIONS, AND PROCEDURES

Preliminaries

We first establish some preliminaries. Large language models (LLMs) are language models that are large both in their parameter size and in the size of their training data. As of early 2024, a typical LLM has on the order of 10^9 parameters and has seen 10^{10} tokens of text.

Prompting is the paradigm of interacting with LLMs at inference-time by giving instructions, or a prompt, for a specific task [4]. From an LLM’s point of view, this is called *in-context learning*. Key to eliciting this ability are *few-shot* examples, where LLMs are given both the instructions and demonstrations of expected input and output.

The persuasion detection task is to, given some input text, to identify the persuasion technique used. We adopted two task formulations from the SemEval series of workshops. The first and simpler task is *identification* -- predict whether a given context contains persuasion. The second is *span extraction* -- given a context, extract spans that utilize persuasion. For either task, *classification* can either be binary, or on a set of persuasive techniques. In this work, we adopt the 23 persuasive techniques as defined by the SemEval 2023 Task 3 dataset (to be referred to as SemEval) dataset [7]. SemEval has 11780 texts and covers 9 languages.

Dataset Creation

We collect a set of paired Wikipedia articles, between Russian (ru) and English (en). We download the full dumps of Wikipedia in both languages, then filter to the subjects where articles link to known Russian state-sponsored news websites. All files are downloaded on 2022-12-01. We use the `mwparsersfromhell` Python package to parse the XML format of Wikipedia articles. For each article, we only consider the article text, and as such remove tables, HTML, and excessive white space. A dump contains all articles from Wikipedia in a given language. The English dump is ~ 20 GB,¹ while the Russian is ~ 5 GB.

We further map Wikipedia article IDs, which are unique per language, to Wikidata item IDs, which are aligned across languages. For instance, for Belgium, the Wikidata ID is Q31, the English Wikipedia ID is 3343, and the Russian Wikipedia ID is 1130. This results in 1,304,255 paired articles.

We next want to filter to a smaller subset, focusing on subjects of interest to Russian authors and readers. Our primary filter considers the external links of articles. Namely, we consider only those subjects where either article links to known Russian state-sponsored media websites.²

We list the filters, and the number of entries remaining after each is applied, below:

- Filter 1 (1,008,298): remove entries whose Wikidata labels that start with ‘category:’ (258667 instances), ‘template:’ (34789), ‘wikipedia:’ (1493), ‘portal:’ (681), ‘module:’ (327).
- Filter 2 (947,834): remove entries whose Wikidata descriptions start with ‘wikimedia’. Example top entries and corresponding entry counts: wikimedia category: 256585,

¹ <https://dumps.wikimedia.org/enwiki/latest/>

² lenta.ru, kommersant.ru, ria.ru, tass.ru, rg.ru, gazeta.ru, rbc.ru, kremlin.ru, interfax.ru, demoscope.ru, vedomosti.ru, kp.ru, regnum.ru, vesti.ru, echo.msk.ru, novayagazeta.ru, ng.ru, rian.ru, publication.pravo.gov.ru, 1tv.ru, vz.ru, iz.ru, aif.ru, rosbalt.ru, izvestia.ru, intermedia.ru, top.rbc.ru, polit.ru, fontanka.ru, ntv.ru

wikimedia disambiguation page: 44562, wikimedia template: 33445, wikimedia list article: 14643, wikimedia set category: 1167, and so on.

- Filter 3 (53,158): consider only subjects linking to Russian media.
- Filter 4 (22,046): consider only “substantially narrative” articles. We define narrative as articles which have at least 2 narrative sections.³ Then, we consider only substantial articles with over 2000 characters.

The final dataset is quite large-scale, consisting of 22,046 paired articles. In addition to the ru and en settings, we consider 2 more: English translated to Russian (en2ru), and Russian translated to English (ru2en). Given the 4 settings, there are 88k total articles. At the paragraph-level, there are 245,778 ru entries (and ru2en), and 295,158 en entries (and en2ru), for >1m entries total.

Literature Review

Biases in Wikipedia. Because of community guidelines such as neutral point-of-view (NPOV), explicit biased statements in Wikipedia articles are removed by editors. Therefore, biases occur more subtly, through being systemic or implicit. Implicit bias occurs when articles selectively choose what details to emphasize or omit [10]. Identifying implicit bias in one article thus requires reference to another. Several authors use temporal edits of Wikipedia as references [11, 12]. They identify from the editing cycle which viewpoints are removed (biased against), and which are kept (biased towards). Our work takes a cross-cultural perspective, instead of a temporal one, in identifying biases; one language’s article use of persuasion is compared against another.

Other works have studied how Wikipedia can be biased across languages. Zhou et al. 2016 study how sentiment differs towards 200 entities in 5 languages [13]. Aleksandrova et al. 2019 develop a system to extract biased sentences in 3 languages [14]. There are several other relevant studies [15,16]. Our work is characterized by its much larger-scale (26K subjects), and its approach to extract potential bias at the span-level.

Multilingual biases of LLMs. While LLMs are able to understand and generate text in many languages, researchers have identified that LLM competency and responses differ cross-lingually. For cultural inquiries, LLMs favor Western values, even when interacting in languages where different cultural sensitivities are desired [17,18]. For factual inquiries, multilingual settings cause LLMs to answer inconsistently [3,19].

Russian vs Western perspectives. The Russian state has positioned itself in stark contrast to the West. As such, Russia has made concerted efforts to spread its narratives and alter public discourse in its favor. This ranges from foreign events to domestic issues: respectively, the 2016 US Presidential Election [20], and the 2022 Russian invasion of Ukraine [21].

Propaganda detection. The most effective forms of propaganda can manifest subtly and be difficult to detect; for example, by intertwining authentic elements with subtly manipulated components [24, 25, 26]. The methods of propaganda are diverse and multifaceted.

³ For this, we consider a section non-narrative if its lowercased heading contains in English [*references, links, see, notes, further, sources, awards, club, results, honours, statistics, charts, works, cast, gallery, filmography, music, media, film, events, singles, discography, citations, league, team, games, video, record, albums, series, teams, nominations, literature, books, footnotes, certifications, honors, publications, records, appearances*] or in Russian [*примечания, ссылки, литература, также, см, награды, достижения, источники, результаты, ролях, фильмография, галерея, достопримечательности, демография, композиций, участники, дискография, чарты, творчество, работы, номинации, альбомы, публикации, синглы, сочинения, фотогалерея, труды, финала, книги, кино, матч, финалы, команды, музыка, выступления, медалисты, матчи, премии, фильмы*]

Early studies center on propaganda detection at the entire article-level. Rashkin et al. [27], ranked news articles based on four categories: propaganda, trusted, hoax, or satire. However, recent focus has shifted to more fine-grained analysis (the focus of our work), in both identifying specific text spans within a large document, and for each categorizing the type of propaganda.

The NLP4IF-2019 workshop released 2 shared tasks: multi-class fragment-level classification of propaganda techniques (FLC) and binary sentence-level classification (SLC) [28]. While several participants achieved high F1 scores for SLC (>60%), FLC proved to be difficult. The next iteration of the workshop, SemEval-2020 Task 11 [29], revised the FLC task by breaking it down into span identification and a revised 14-way propaganda technique classification. The top performing teams had an F1 of 51.55 for span identification and 62.07 for propaganda technique classification.

The latest iteration at SemEval-2023 Task 3 [7] introduced a multilingual component across eight languages, including Russian, and was conducted at paragraph level. The task provided a representative dataset of contemporary global issues, such as the COVID-19 pandemic, abortion-related legislation, and the Russo-Ukrainian war.

AI-assisted report generation. AI-assisted report generation uses AI tools (such as LLMs) to take in multiple documents and assemble a report which summarizes the key points for a specified audience. Barham et al. 2023 consider Wikipedias in 50 languages, to generate a large-scale dataset of 120m QA pairs, indexed to 71m reports [22]. For a given passage, its citations are used as reports, and an English question-answer pair is generated from it. Li and Callison-Burch propose a scalable approach to generate cross-lingual QA pairs on paired passages [1]. Reddy et al. 2023 develop an LLM-powered system to generate reports to assist decision-makers in high-stakes issues [23]. Our work takes inspiration from all of these, in studying Wikipedia, making cross-lingual comparisons, empowered by LLMs.

I. Uncovering Differences in Persuasive Language Across Russian and English Wikipedia

The following diagram gives an overview of the overall approach. Top of figure shows process where we have an LLM generate a large repository of high-level questions (HLQs), based on its own understanding of persuasion techniques. We then pose these HLQs to articles from a labeled persuasion dataset. We select the subset of questions which are most aligned to the human labels, using feature selection to arrive at 12 HLQs. At bottom of figure is process on a different dataset of interest, where we use HLQs to prompt an LLM to identify-then- extract persuasive spans. This is done over paired Wikipedia articles in Russian and English, facilitating cross-lingual comparison. For the LLM, we always use GPT-4, as we found it had the best performance. We also ran some experiments with Llama-2, an open-source model, and describe this in Appendix D.

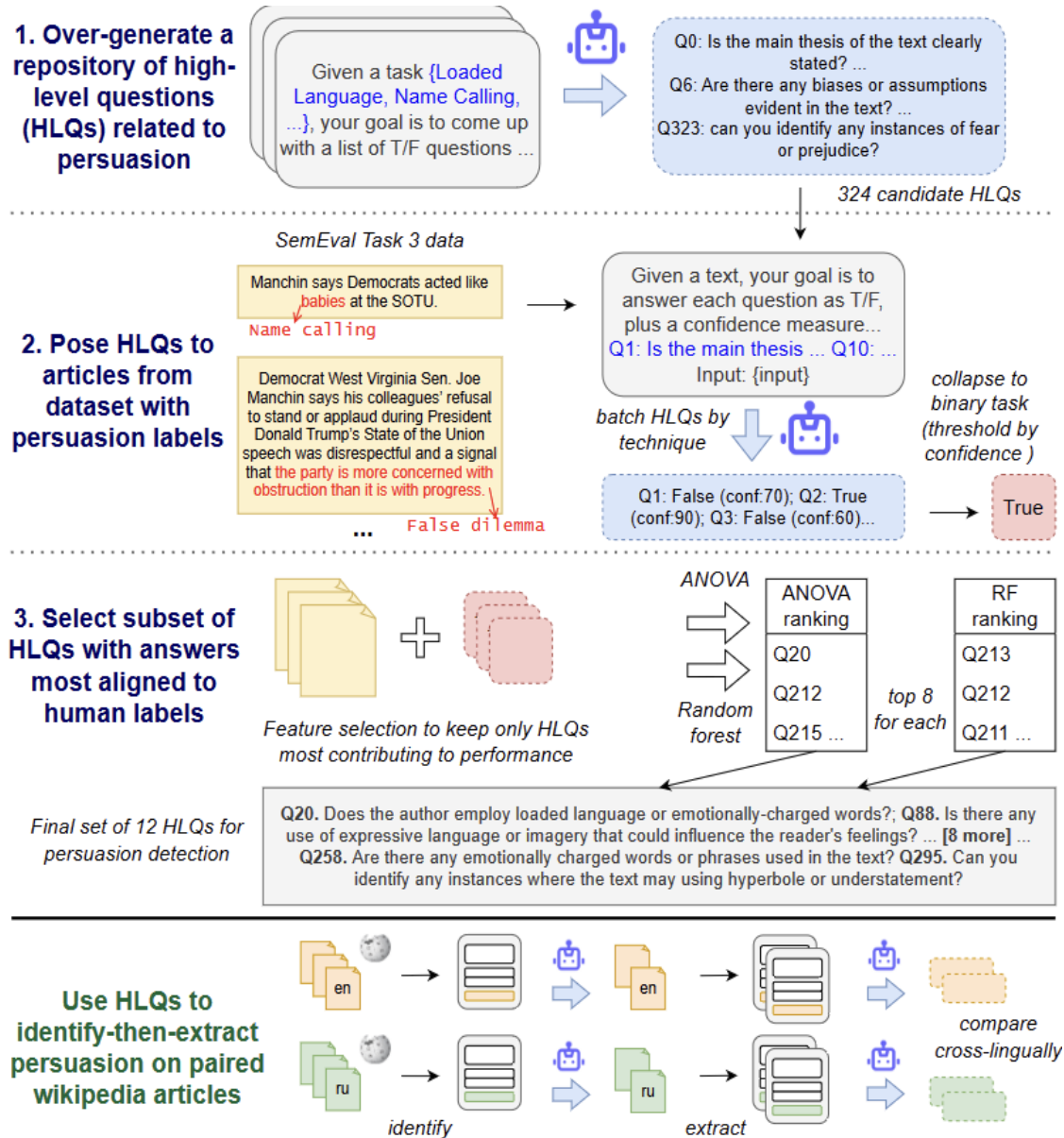


Figure 1: Top: methodology to generate high-level questions; Bottom: querying with LLMs in Russian and English and comparing crosslingually

I.A. High-level QA

We experimented with two approaches towards identifying whether a context contains persuasion, a vanilla prompting baseline, and the proposed high-level questioning (HLQ) approach. We describe the methods in detail ahead, and depict them in the diagram below. Figure 2 shows a comparison between two approaches to persuasion technique detection. The baseline directly uses the human-authored definitions in the prompt. However, as these definitions were written for trained human annotators, the LLM misunderstands them and is over-sensitive and over-confident. This is addressed in the proposed approach by leveraging the LLM’s own intuition to generate high-level questions (HLQs).

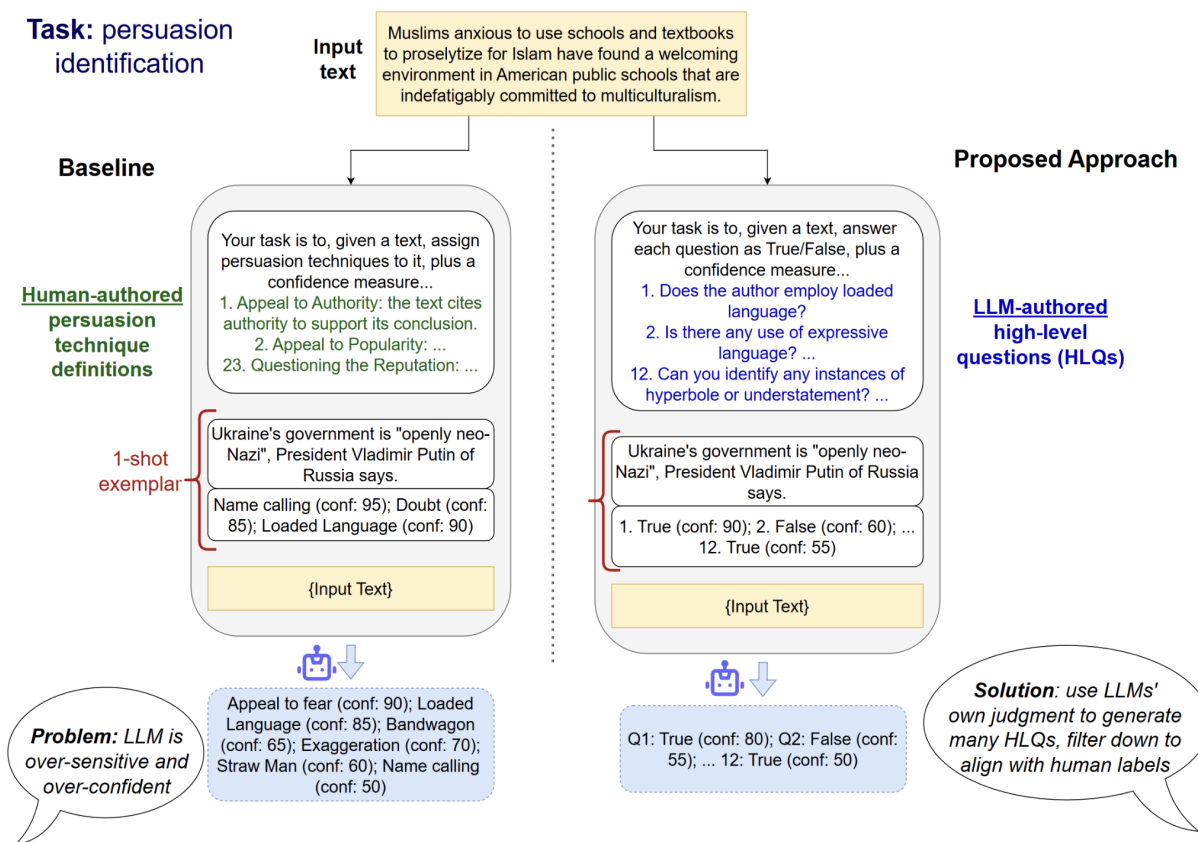


Figure 2: A comparison between two approaches to persuasion technique detection.

Baseline for Persuasive Language Detection

The baseline prompt⁴, as shown on the left of the diagram above, includes each persuasive technique, as well as a human-authored definition from SemEval. We further ask the LLM to generate a confidence score for each predicted technique, which we use to threshold predictions (described shortly ahead).

⁴ The full text used for this baseline prompt, as well as all other prompts, can be found in Appendix B.

The main issue with this direct approach is that understanding of persuasion is extremely subjective. In collecting the gold labels for SemEval, Piskorski et al 2023 invested significant efforts into training 35+ human annotators (multilingually) and revising instructions throughout. So, by directly giving the LLM the final definitions, we cannot expect it to be aligned with the judgements specific to this annotation task.

Baseline makes LLMs over-sensitive. To demonstrate the divergence in LLM understanding of persuasion vs. humans, the following Table 1 compares the raw counts for each persuasion technique over SemEval (11780 total contexts).

Table 1: The raw counts for each persuasion technique for the SemEval task [7], English split only. GPT-4 predictions are thresholded to confidence ≥ 50 .

Persuasion Technique	Gold Count	Vanilla Count	Persuasion Technique	Gold Count	Vanilla Count
<i>None</i>	6945	1450	Conversation Killer	115	120
Loaded Language	2277	2484	Red Herring	63	101
Name Calling-Labeling	1226	1871	Guilt by Association	63	339
Doubt	703	2824	Appeal to Popularity	48	478
Repetition	684	407	Appeal to Hypocrisy	45	104
Exaggeration-Minimisation	576	1571	Obfuscation-Vagueness-Confusion	30	482
Appeal to Fear-Prejudice	442	2260	Straw Man	24	19
Flag Waving	376	46	Whataboutism	18	179
Causal Oversimplification	236	848	Appeal to Values	0	1938
False Dilemma	180	307	Consequential Oversimplification	0	361
Slogans	180	124	Appeal to Time	0	577
Appeal to Authority	179	6286	Questioning the Reputation	0	483
Total (all 23 excluding <i>None</i>)	7465	24209			

We observe that the gold labels are highly imbalanced. Notably, 59% of texts contain no persuasion (6945); however, GPT-4 predicts only 12.3% (1450). For the gold labels, 47% of non-None labels are from *Loaded Language*. 11 classes appear less than 1% of the time. Furthermore, as this is a multi-class labeling task, those <1% labels often appear with *Loaded Language*. GPT-4 does not have a sense as to the class priors and over-predicts the prevalence of all 23 techniques -- 24,209 vs. 7,465.

For example, consider *Appeal to Authority* (6286 vs. 179). The vanilla prompt incorrectly assigns this to most mentions of people’s titles (e.g. “President Vladimir Putin of Russia”) or news sources (e.g. “New York Times”). Also, for *Obfuscation-Vagueness-Confusion* (482 vs. 30), the LLM assigns this to any short or unclear context (e.g. “Email *”), while the technique has a more specific definition. For *Flag Waving* (46 vs 376), the LLM rarely assigns it. We observe it uses other techniques as catchalls -- again, the subjectivity of the task allows for overlap between techniques, yet the annotators still followed specified guidelines. In sum, we can say that LLMs are over-sensitive to persuasion.

Baseline makes LLMs over-confident. We measure confidence by the model output’s confidence scores for each predicted persuasive technique. As there are 23 techniques, for analysis purposes we make a task simplification to binary classification. A context is ‘True’ if any predicted technique has confidence $\geq x$, else ‘False’. We then use F1 to choose the optimal threshold.

Table 2: F1 score of GPT-4 on the binary persuasion task, at different confidence thresholds.

Confidence Threshold ($\geq x$)	F1	‘True’ LLM output	Confidence Threshold ($\geq x$)	F1	‘True’ LLM output
$x = 20$	0.469	10079	$x = 70$	0.509	9401
$x = 30$	0.459	10165	$x = 80$	0.582	7233
$x = 40$	0.454	10223	$x = 85$	0.608	4264
$x = 50$	0.447	10293	$x = 90$	0.573	2833
$x = 60$	0.45	10266	$x = 95$	0.448	99

Table 2 shows the F1 score at different confidence thresholds. The best F1 is at $x=85$. ‘False’ counts can be obtained by $11,780 - \text{‘True’ count}$ -- for example, for $x = 20$, ‘False’ count == 1701. Compare these predictions to the gold label count of ‘True’ = 6945. Until $x=60$, the model assigns $>87\%$ (10266) of texts as containing persuasion. F1 is maximized at $x=85$, at 0.608, compared to .469 without considering confidence. We therefore have shown both that the model is over-confident, and that our method to prompt and threshold for confidence scores substantially improves performance.

High-level questioning

The high-level questioning approach to persuasion detection is depicted at the top of Figure 1. At a high-level, the idea behind HLQs is to leverage LLMs’ own (many different) intuitions on a task, then filter down to those that best align with gold labels on a reference dataset. Considering Figure 2, we see that from the LLM’s perspective, both baseline and HLQ prompts are constructed similarly. However, the LLM-authored reframing improves performance quite significantly.

Motivating High-level questioning. Let us consider the two typical prompt engineering approaches used when an LLM fails to adequately perform on a prompt. First, more detailed instructions can be written. In our case, this would need to be done for all 23 techniques. This would result in one huge prompt, or even 23 separate prompts. While this would likely work after some trial and error, it would have hugely increased inference costs.

Second, we can include more few-shot exemplars. In the baseline, we used a single, static exemplar. Suppose one wanted to use multiple, dynamic exemplars. Typical prompting techniques would, for each inference entry, randomly draw exemplars from a train split (which is provided for SemEval). We piloted this, but it did improve over the baseline due to the challenges of having 23 distinct techniques, and the uneven class priors. Therefore, we have motivated using high-level questioning for persuasive language detection.

Step 1: Over-generating a repository of questions. In the first step, we write a simple zero-shot prompt which tasks an LLM to generate a list of True/False questions for a specified persuasion

technique. For this step, the key is getting LLM’s zero-shot understanding from various angles through its own generations.

It is expected that many questions overlap in coverage; while to a human, two questions may appear to ask similar things, they may effect large differences in LLMs. We do, however, manually filter out questions which are nearly identical. Our manual analysis also finds that most questions are reasonable and reduce subjectivity vs. persuasion techniques. This results in a repository of 324 questions.

Step 2: Posing HLQs on a labeled dataset. Given the repository of HLQs, can we find which ones are most effective at detecting persuasive language? We do so by leveraging existing annotations from SemEval for the ground truth (step 2 of Figure 1).

Given the repository of HLQs, we batch them into sets containing all generated HLQs for a technique. Then, in a single prompt, an LLM is asked to answer True/False for the batched HLQs over the entire SemEval dataset (11,780 entries).

To compare to the gold annotations, we follow what was done for the baseline, to simplify and collapse SemEval to a binary classification task. We train a logistic regression model over the set of HLQs, using five-fold cross-validation, and recording the mean F1 across the folds. As shown in Table 3 below, prompting with all 324 HLQs improved F1 by 23.5% relative over the baseline: F1 of 0.751 > 0.608.

Table 3: Performance on binary classification task of SemEval (confidence threshold $\alpha=0.85$)

Method	P	R	F1
24 definitions	0.607	0.613	0.608
12 HLQs	0.757	0.748	0.751
324 HLQs	0.746	0.733	0.737

Selecting a subset of most-aligned HLQs. The approach so far works well but uses many prompts -- one prompt (with multiple questions) for each of the 23 techniques. In our next step, we select a subset of top-ranked HLQs, which maintains performance, while fitting into 1 prompt (step 3 of Figure 1).

In our feature selection methodology, we embarked on a systematic elimination of features, ranking them using either ANOVA or Random Forest with Gini impurity and removing the least important feature one at a time. First, the SemEval data was employed in a 5-fold cross-validation across a suite of classifiers, which included Logistic Regression, Random Forest, and Support Vector Classifier (SVC). We controlled the randomness inherent in the machine learning algorithms by fixing the random seed parameter across all classifiers. The mean F1 across the 5 folds was recorded. Following the reduction of each feature set, we applied 5-fold cross-validation using the suite of classifiers again. For each iteration, we computed the F1-score to quantitatively assess the classifiers’ performance on the reduced feature set.

Figure 3 illustrates the impact of feature reduction on the classifiers’ effectiveness by plotting the F1-score against feature set size using ANOVA. It showcases the relationship between the F1-

score and the progressively diminished feature sets. This displays the performance of the classifiers at the reduced feature set range from 30 to 1. It reveals a stable performance across classifiers until 10 features remain and highlights a decline in performance after the last 8 features remain. Elimination of features for both ANOVA and Random Forest reveals a stable performance across classifiers until around 8 features remain. Thus, we combine the top 8 features from ANOVA, and top 8 from Random Forest. This results in a final subset of 12 HLQs, which is shown in Appendix Table C-1.

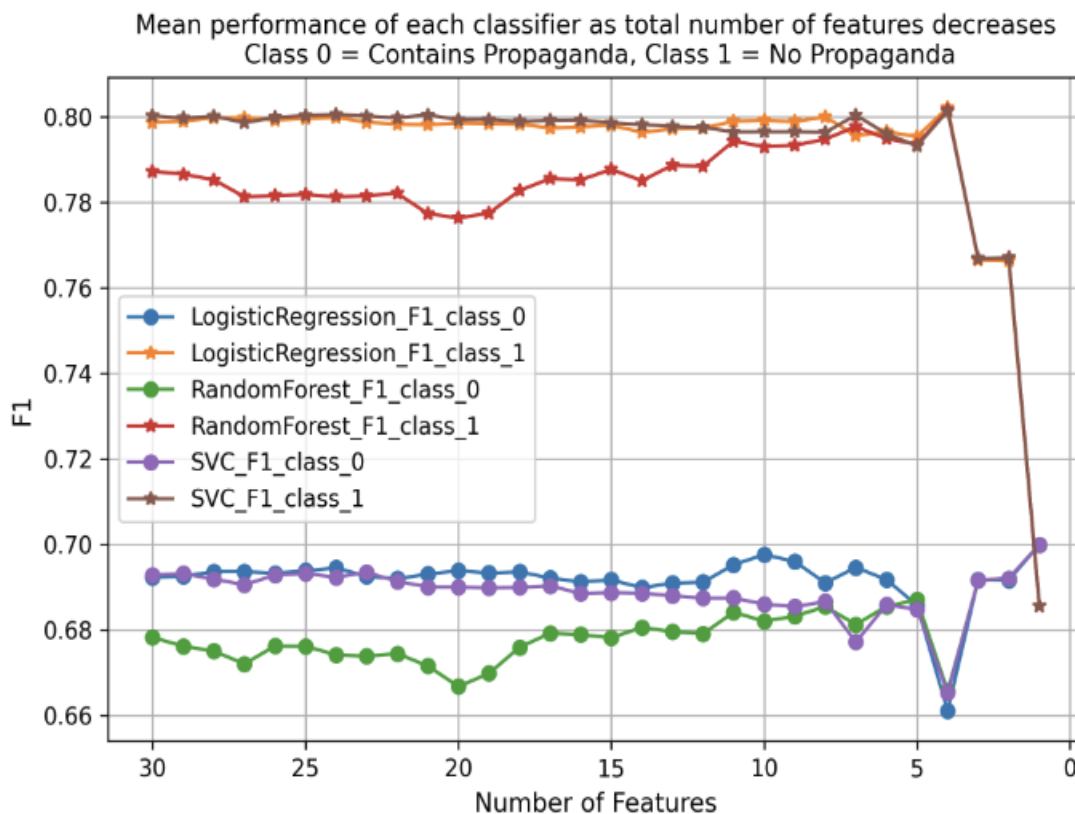


Figure 3: The effectiveness of the classifiers after each feature reduction using ANOVA.

Furthermore, referring back to Table 3, we see that the 12-question subset slightly improves over the full set of 324 HLQs: $0.751 > 0.737$.

Extending HLQs to Russian. With the top 12 HLQs selected, we employ a native Russian speaker for translation. They were allowed to prompt GPT-4, for assistance, before further post-editing.⁵

Identify-then-extract methodology

While in the prior section, we used SemEval, we now consider the dataset of paired Wikipedia articles in Russian and English. The use of persuasion is less pronounced in Wikipedia vs news article domain, so uncovering these subtle differences across languages can be especially informative.

⁵ When translation is required for other prompts used in this work, we follow the same process.

We adopt a two-stage hierarchical prompting approach towards persuasive language detection, using GPT-4,⁶ which we term identify-then-extract (Figure 1, bottom). All steps are monolingual, in that Russian contexts use Russian HLQs and prompts, and likewise for English.

Motivating identify-then-extract. In the dataset of interest, there are 1m paragraphs (across 88k articles). We found that using only a single stage, asking to extract instances of persuasion (1 prompt per technique) for every context resulted in far too many false positives (even though we include instructions that null responses are possible). This applied to both definitions and HLQs. We suspect this is because the objective of LLMs is to complete a user instruction, if at all possible, even if inaccurate. The high false positive rate also has a high cost for inference, at the large-scale explored in this work.

Step 1: Identify. Identification is the task that has been described before for baseline and for HLQs. In this stage, for each context, we prompt an LLM to answer True/False for all HLQs at once (Figure 2, right). This results in judgments for 12m (1m paragraphs * 12 HLQs) entries.

Step 2 Extract. Of the 12m judgments, we only consider the contexts and the selected set of HLQs marked as ‘True’. For the paired Wikipedia articles dataset, 85.1% are marked ‘False’, and so do not need to be queried -- this shows the identify-then-extract approach saves much inference costs over a single-stage.

For each, we insert the context and one HLQ into a prompt template (given in Appendix B), which tasks the LLM to extract spans employing that HLQ. In contrast to the single prompt per context from the Identify stage, the Extract stage is hierarchical, having a set of ‘True’ HLQs, and thus prompts, per context.

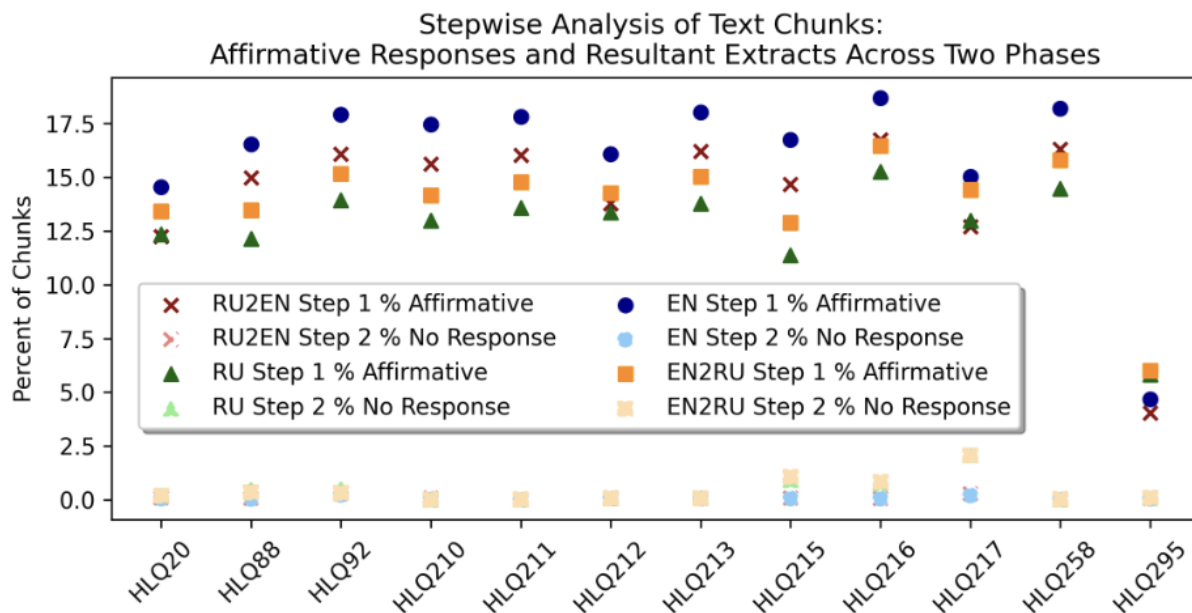


Figure 4: Scatterplot for affirmative responses from identify and extract stages

⁶ We detail experiments on the Llama-2 LLM in Appendix D. It was reasonable but had a higher false positive rate than GPT-4.

Figure 4 is a scatter plot with stepwise analysis of the text chunks. Top: the percentage of chunks that responded affirmatively to each HLQ for step 1. In the subsequent step, all chunks that had an affirmative response are examined to isolate the respective text that led to this answer. Bottom: percent of chunks from step 2 that yielded no result. It effectively portrays that the majority of responses, which were initially affirmative, produced the text responsible for such answers as per expectation.

Collapsing extracted spans which overlap. The HLQs, while nuanced, largely cover the same aspects of persuasion. This means that LLM output to them will also contain many overlapping terms. Given that for analysis purposes, we reduced the task from multi-class labeling to binary labeling, we should also collapse the multi-class extracted spans to a deduplicated set.

This deduplication algorithm is implemented with a simple string overlap approach. We start with an empty set S . We then initialize it with all spans from HLQ1 $S_1 = \{s_1^{S_1}, \dots, s_i^{S_1}\}$. Then, for S_2 of HLQ2, we check to see if each span $s_j^{S_2} \in s_i^S$. If so, we replace s_i^S with the longer of the two strings and repeat until all sets S_2 to S_{12} are processed. The final set S , which we term the persuasive text set (PTS) is a model’s response to the persuasive span extraction task.

Table 4 shows model responses for 2 questions on sections from the article ‘Fire at the national Museum of Brazil’, as well as the PTS (excerpted for the 2 questions).

Table 4: Sample Model responses (ru2en), on ‘Fire at the National Museum of Brazil’ (WikiID:’ Q56441760). ‘PTS’ is the deduplicated persuasive text set.

Index	QID	Specific Text Instances Identified
2	Q20	engulfed, rapidly destroyed, tragedy, repeatedly complained, ...
2	Q88	fire engulfed, rapidly destroyed, tragedy, funding cuts, , ...
PTS		fire engulfed, rapidly destroyed, tragedy, repeatedly complained, funding cuts
3	Q20	incalculable, outraged, cultural tragedy, lobotomy
3	Q88	fire, loss, outraged, tragedy, destroyed, ruins, threat, ...
PTS		incalculable, outraged, cultural tragedy, lobotomy, fire, loss, destroyed, ruins, threat

Generating Synthetic Propaganda

Recall that despite the quality of SemEval, and related datasets, the *coverage* of persuasion techniques is highly imbalanced, due to real-world usage in the wild. As we found in our experiments above, this imbalance lends itself to our own study, where 9 of the 12 top ranked HLQs were related to the top class, Loaded Language. The other persuasion techniques have been underrepresented, and therefore, we are motivated to release a synthetic dataset which covers all 23 SemEval techniques in a balanced way (~10k each).

Our focus is on nation-state propaganda. An added advantage of our dataset is that it includes a repository of propaganda examples shaped on a DIME-like⁷ ontology for factors that a country wants to promote. This guarantees a broad coverage of topics.

Example DIME-like high-level categories are: Judiciary, Military, Healthcare System, Education System, Infrastructure, Financial Institutions, Communication Networks, Technological Advancements, Educational Excellence, Trade Policies, Human Rights Record, and others.⁸

The prompt for generating components within each high-level category is:

System: Produce a list of components making up {Country}'s {category}.

For example, for country 'Russia' and high-level category 'Military' LLM produces:

Russian Ground Forces, Russian Aerospace Forces, Russian Navy, Strategic Missile Troops, Russian Airborne Troops, Federal Security Service (FSB), ..., Russian Electronic Warfare Troops, Russian Engineer Troops, Russian NBC Protection Troops, Russian Signal Troops, Russian Rear of the Armed Forces, Russian Logistics Support.

We can further refine this ontology by requesting subcomponents of each component from above.

We have found that using the term 'institutions' instead of 'components' often leads to the generation of educational institutions. In a military context, this would imply military academies and universities, not military organizations or units. Thus, proper keywords and prompt engineering are important in this context.

This country-specific ontology is used to generate synthetic examples for either country. The query to GPT-4 is (where Action can be 'promote' or 'minimize'):

System: You are a helpful assistant with a background in political science, economics, and international relations.

User: Produce a paragraph that {Action} {Country}'s {Component} using {Propaganda Technique}: {Technique Definition}

At this point, the dataset has examples for the top 20 countries based on population. The dataset is a JSON file containing 225092 items. Some queries fail to perform specified action, and would be promoting instead of minimizing. GPT-4 might find some queries offensive and refuse to generate a response. Thus, additional verification is required over this dataset.

The existing literature details numerous propaganda techniques, but until now, it has not been possible to create a dataset that encompasses all of them. Our goal is to continue expanding this dataset over time.⁹

I.B. Cross-lingual Extractive QA

While we ultimately chose the high-level QA approach, as we found it provided the best suite to perform our cross-cultural comparison, we have also worked on an alternative approach based on cross-lingual Extractive QA. This paradigm consists of two parts: the QA generation (QG), and then the cross-lingual QA and analysis across paired articles.

⁷DIME = diplomatic, informational, military, and economic factors

⁸Categories are for "items a nation-state might wish to promote to gain advantage over other nation-states"

⁹ dataset and updates will be available at: https://github.com/apanasyu/UNCOVER_SPIE

First, in the QA generation part, for each article itself, we generate a set of questions grounded in the article’s content. These questions are factual in nature, in line with the extractive QA task popularly considered in NLP. We used PAXQA¹⁰ for generating such cross-lingual QA pairs [1]. The PAXQA methodology, depicted in the below diagram, decomposes cross-lingual QA into two stages: first, generating questions from an English article (paired with a translation), and second, using word alignments to perform informed translations of the questions and answers into another language. This decomposition enables for high quality cross-lingual QG without training any new models, and instead putting together state-of-the-art components for English QG, translation, and word alignment.

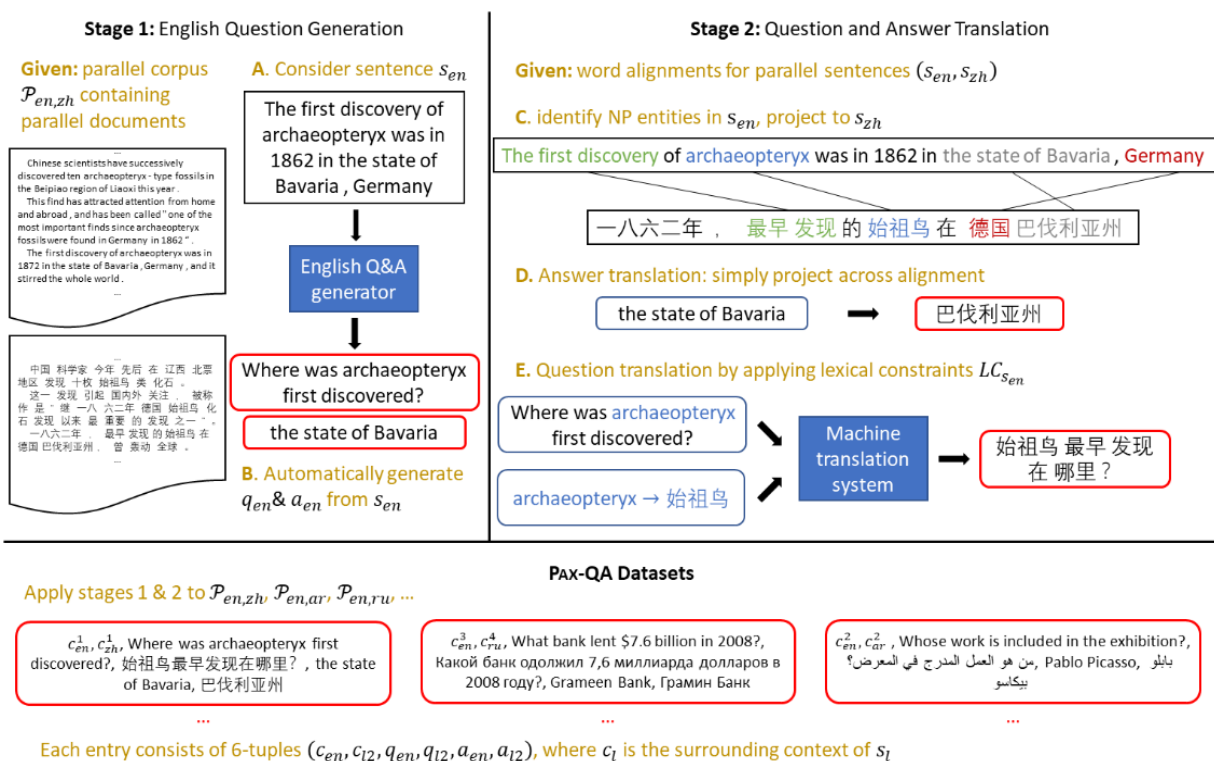


Figure 5: High-level overview of the PAXQA approach to cross-lingual question answering generation. It decomposes the task into two stages (top), and results in 6-tuple entries which are bilingual for questions, answers, and contexts.

While Li et al. originally used non-LLM components – finetuned versions of T5, Transformer, etc. – in this work, we experimented with replacing each with prompt engineering for an LLM (we used GPT 3.5). We term this PAXQA_LLM. We omit the prompt development and iteration process here for brevity, but it was similar in the experimental process as discussed in depth in prior Section 3.I.A. We also applied the systems to both languages of the paired articles.

On a random subset of articles, we performed a manual comparison between the set of Q&A pairs generated by the original PAXQA to PAXQA_LLM. Stylistically, the systems asked similar

¹⁰ Penn published this project, and presented at a conference, during the grant period

factoid questions, easily answerable from an article’s text. Quality-wise, we found that while close, PAXQA_LLM usually produced more natural-sounding questions, though PAXQA questions were still valid, just sometimes awkward. The major difference between the generations, then, was in terms of coverage. While PAXQA extracted ~1 question per sentence, GPT-3 was able to generate more. This is key given that for the Wikipedia domain in consideration, many sentences may be long and complex, so asking multiple questions is important.

Second, the cross-lingual aspect occurs when we consider the paired articles between English and Russian, occurs when we take the questions generated from English and ask them of the Russian article, and vice versa. This can result in three scenarios, a) the answers are the same (agnostic of the language), b) the answers differ, or c) the question is unanswerable from the other article. For example, the below Figure shows case b) to the question for “What did Zhukov do after the war?” when asked about the English and Russian Wikipedia articles on Georgy Zhukov.



Figure 6: Comparison of cross-lingual QA between the English and Russian versions of Wikipedia on Georgy Zhukov. We highlight a question which is answered differently depending on the language.

We see that the Russian article is matter of fact and lists his official-sounding titles. However, the English article merely states that the titles were insignificant, and Stalin, the Soviet leader, did so to avoid potential political conflict. The English article injects the author’s own opinion, which corresponds to Western historian’s views. We performed some initial inquiries into setting up such an analytical approach across the Wikipedia corpus we developed. However, we ultimately decided that I.A would be a more effective approach.

Why we pursued I.A over I.B In pursuing the I.A and I.B approaches in parallel, we decided that I.A made more sense in the context of our high-level goal – understanding differences in persuasive content usage by Russian and Wikipedia authors. I.B, in contrast, is more concerned with

understanding differences in *factual information*, which while related, would be better served by a future effort. Also, the analysis for the factual-style QAs would have to be more carefully thought out, given each article would have its own questions. We would have to further consider what information is different just because of authors in one language being less comprehensive, vs. authors actively omitting or altering perspectives. For I.A's static set of questions, we only need to consider at the question-level, instead of across all articles.

Still, we find our earlier efforts in I.B and cross-lingual extractive QA were very informative for our final approach in I.A. The main takeaways are that first, we can harness the power of LLMs and prompt engineering (thus enabling no training of custom models, and a unified way of interfacing with models) for various tasks, and second, that a careful decomposition of a complex task into subtasks, each solved by a different prompted LLM, can be effective.

II. LLM Interface Development – Kani

Language model applications are becoming increasingly popular and complex, often including features like tool usage and retrieval augmentation. However, existing frameworks for such applications are often opinionated, deciding for developers how their prompts ought to be formatted, and very heterogeneous, with no standard interface across different language models. These together impose limitations on customizability and reproducibility. To solve this we developed Kani: a lightweight, flexible, and model-agnostic open-source framework for building language model applications [2]. Our goal with Kani was to help researchers implement more complex features by providing the core building blocks of chat interaction – model interfacing, chat management, and robust function calling – with a unified interface. All Kani core functions are easily overridable and well documented to empower developers to customize functionality for their own needs. Kani allowed us to switch out underlying LLMs in complex systems with minimal code changes and iterate quickly on prototypes for the UNCOVER systems.

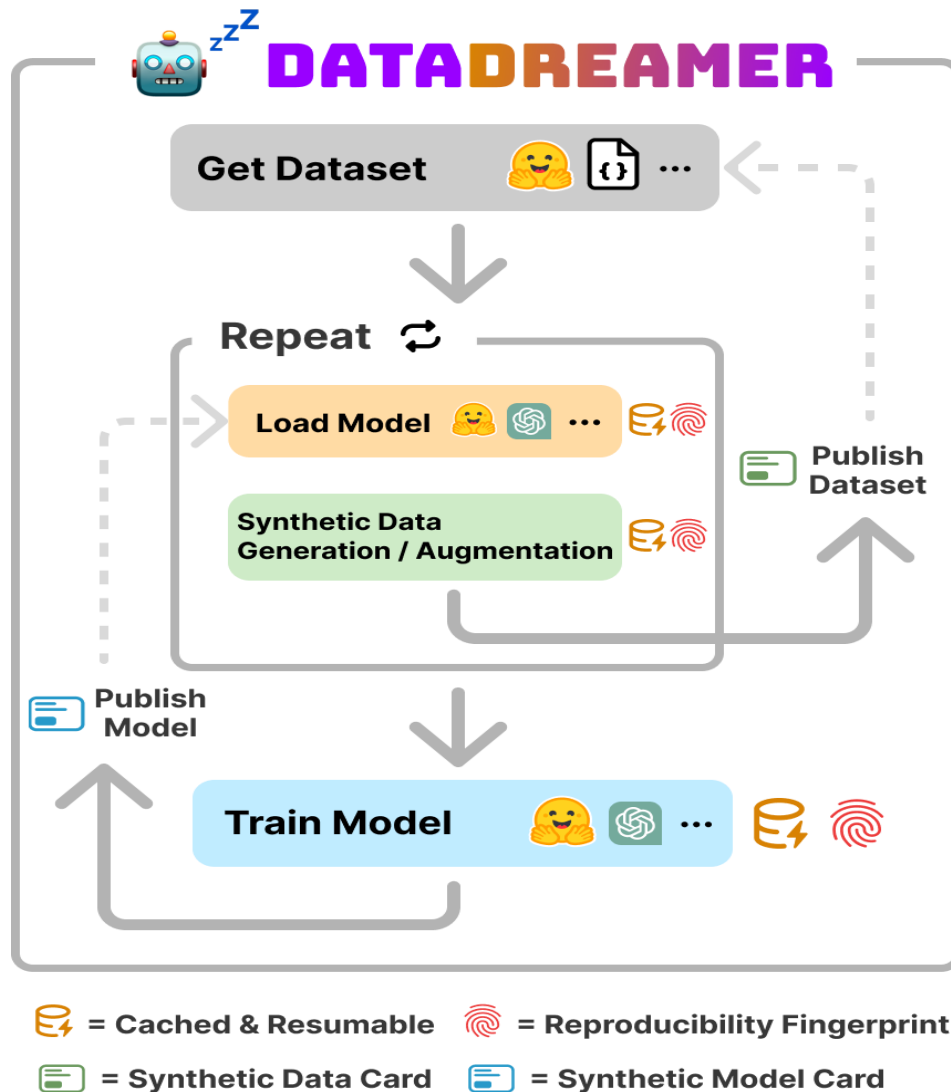


Figure 7: DataDreamer workflow

II.A. Related work – DataDreamer

Large language models (LLMs) have become a dominant and valuable tool for NLP researchers in a wide range of tasks. Today, many researchers use LLMs in synthetic data generation, task evaluation, fine-tuning, distillation, and other model-in-the-loop research workflows. However, challenges arise when using these models that stem from their scale, their closed source nature, and the lack of standardized tooling for these new and emerging workflows. The rapid rise to prominence of these models and these unique challenges has had immediate adverse impacts on open science and on the reproducibility of work that uses them. In this paper, we introduce DataDreamer, an open-source Python library that allows researchers to write simple code to implement powerful LLM workflows. DataDreamer also helps researchers adhere to best practices that we propose to encourage open science and reproducibility. The library and documentation are available at: <https://github.com/datadreamer-dev/DataDreamer>.

While large language models (LLMs) have established a new era in NLP research through the prompt-and-predict paradigm that has proven effective on a wide variety of tasks, the use of these models has come with significant drawbacks (Liu et al., 2023). Many popular models like GPT-4 (OpenAI et al., 2023) are closed source and behind a remote API, while running models locally can be technically complex and expensive due to their scale. Moreover, the now well-established prompting paradigm can be brittle with results widely varying between different models, configurations, and environments (Sclar et al., 2023; Jaiswal et al., 2023). These challenges have made it difficult for researchers to share, reproduce, extend, and compare work, hindering the rate of research progress.

In context of the rapid shift to using these large models in research, we introduce DataDreamer, our open-source Python package that provides both practical utility to researchers and scientific utility to the community:

- DataDreamer helps researchers implement state-of-the-art emerging workflows involving LLMs such as synthetic data generation, fine-tuning, instruction-tuning, and alignment. It simplifies implementations by providing a single library with a standardized interface for many of these tasks while reducing technical complexity around switching between models, caching, resumability, logging, multi-GPU inference and training, using adapter and quantization optimizations, and publishing open datasets and models.
- DataDreamer makes chaining data between tasks, an increasingly widespread practice, simple. For example, a user can generate data with a synthetic data workflow and then fine-tune on that synthetic data.
- DataDreamer helps researchers implement workflows while crucially producing output that is compatible with open science and reproducible ideals with minimal effort, through automatic caching, reproducibility fingerprints, and more best-practice artifacts.

III. Evaluating Geopolitical Biases in Language Models

In our primary study, we performed a deep dive into the differences between Russian and English language perspectives on various issues. We used LLMs throughout our development and research to accelerate such insights at large scale, and across the two languages. With our careful prompt designs, pipelines, and filtering steps, we tried to best align LLM responses with human annotator’s judgements. Several questions naturally arise from here. How can we extend analysis to multiple cultures (and their respective languages?) Are LLMs intrinsically biased towards certain cultures, and if so, can the language of interaction affect this?

Inspired by these questions, we propose to study territorial disputes. An example territorial dispute is the status of Crimea – while Russia annexed this territory from Ukraine in 2014, it is internationally recognized as part of Ukraine. A single multilingual LLM, when queried in Russian, is more likely to respond “Russia,” but respond “Ukraine” in Ukrainian and English. We study in depth the problem of territorial disputes and LLMs in our recent work [3], which will be presented at the NAACL 2024 conference. Our querying approach is shown below:

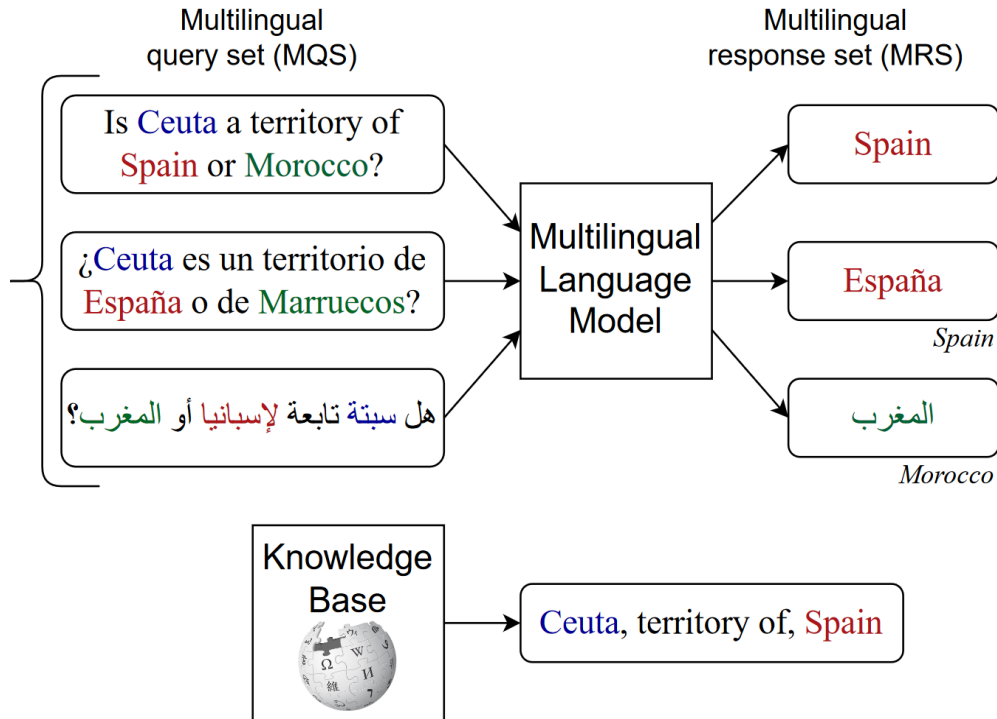


Figure 8: Depiction of a territorial disputes task studied for BorderLines

LLM’s inconsistency in responses across languages contrasts with a multilingual human, who would likely answer consistently. We show that LLMs recall certain geographical knowledge inconsistently when queried in different languages---a phenomenon we term geopolitical bias. As a targeted case study, we consider territorial disputes, an inherently controversial and multilingual task. We introduce BorderLines, a dataset of territorial disputes which covers 251 territories, each associated with a set of multiple-choice questions in the languages of each claimant country (49 languages in total). We also propose a suite of evaluation metrics to precisely quantify bias and consistency in responses across different languages. We then evaluate various multilingual LLMs on our dataset and metrics to probe their internal knowledge and use the proposed metrics to discover numerous inconsistencies in how these models respond in different languages.

Finally, we explore several prompt modification strategies, aiming to either amplify or mitigate geopolitical bias, which highlights how brittle LLMs are and how they tailor their responses depending on cues from the interaction context. We will further discuss it in the Results section.

4.0 RESULTS AND DISCUSSION

I. Uncovering Cultural Differences Across Russian and English Wikipedia

I.A High-level questioning

We performed five experiments, and now discuss the results.

For our cross-lingual analysis over the paired articles dataset, we propose several metrics. We use these for various experiments, which make different comparisons and aggregations. We consider the persuasive text sets (PTS) obtained with the approach in identify-the-extract with HLQs.

Metrics to Quantify Persuasion. The function $cc(text)$ counts the number of characters¹¹ in a text. We define several metrics, Persuasive Count (PC), and Persuasive Frequency (PF):

$$PC = cc(PTS) \quad (1)$$

$$PF_{\text{para}} = \frac{cc(PTS)}{cc(\text{para})} \quad (2)$$

$$PF_{\text{article}} = \sum_{\text{para} \in \text{article}} PF_{\text{para}} * \frac{cc(\text{para})}{cc(\text{article})} \quad (3)$$

These metrics are used for experiments 1, 2, and 3.

Normalized persuasion frequency (NPF). For experiments 4 and 5, we introduce a version of PF which normalizes PF scores across all authors (either Russian or English) between 0 and 1. To quickly illustrate, suppose the max PF is 0.6, and the min is 0.05. NPF would draw the max towards 1, and the minimum towards 0. The pseudocode is given below:

```
author1_pf = calc_pf(author1_article_length_list, author1_pc_list)
author2_pf = calc_pf(author2_article_length_list, author2_pc_list)
npf = normalize_scores(author1_pf + author2_pf) \quad (4)
```

Where `calc_pf` returns $\text{author1_pc}[i] / \text{author2_article_length_list}[i]$ (for $i = 0, 1, \dots, n$), and n is the number of articles.

Why normalize? Recall that Wikipedia guidelines specify a neutral point-of-view (NPOV). If we assume that different individual authors aim for the same NPOV, then we can normalize PF scores for one language. Suppose the NPOV for Russian differs from English. Then, we can “normalize” out baseline NPOVs in either language by taking all articles and their PF together. Putting them on a common scale makes comparing the relative emotional content between authors more meaningful.

¹¹ We acknowledge that measuring by characters is imperfect, since word lengths differ between languages. This also applies to words or tokens; we also tried those but found characters worked best.

Experiment 1: Case Study on 2021 Russian Protests

Figure 9 is a depiction of the method to compare persuasive language usage across languages. For each language, we use HLQ prompts *monolingually* on all articles to extract persuasive text spans (left: en, right: ru). We compare both persuasive count (PC) and persuasive frequency (PF) between the paired articles. For this case study, the Russian article (and its translation ru2en) are more persuasive on ‘2021 Russian protests’. Figure depicts a case study on the subject ‘2021 Russian Protests’. Interestingly, the articles have different titles; the Russian article specifically mentions their support of Alexei Navalny. The LLM extracts more persuasive content from the Russian-authored articles than the English; on the left diagram, 0.15 for en vs. 0.24 for en2ru. The system identifies the Russian article (translated to English) as containing the loaded term ‘‘oppositionist’’. The English article, in contrast, uses the neutral term ‘‘opposition leader’’, which is not extracted by the system.

For a cross-lingual comparison, the PF for an original article and its translation are close (.14 & .15, .24 & .22); which is a positive signal that LLM performs the same extractions whether using the Russian or English prompts. This sanity check is expanded in a following section.

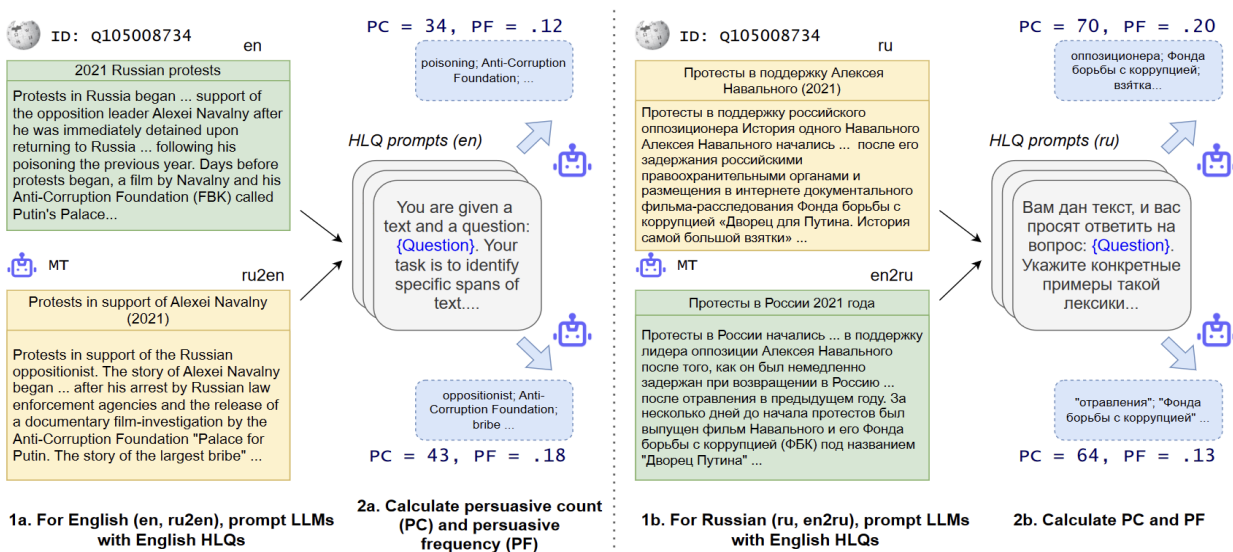


Figure 9: Depiction of the method to compare persuasive language usage across languages.

Experiment 2: Ranking Wikipedia Articles for each Language

This experiment investigates which subjects contain the most persuasive content, as measured by PC, for Wikipedia authors, in either language.

Table 5 shows for 2 settings (en, ru2en) the top 20 ranked articles. The comparative rankings generally align with our intuition. For the Russian top 10, 4 subjects concern current conflicts

Russia is involved with: in Syria, and in Ukraine. Considering #10-20, these relate to Russian historical figures and events. In contrast, for the English the subjects are far more internationalized, for example concerning China, Gaza, Christians, Muslims, and Madonna. Interestingly, the article on “Trumpism” appears in both rankings (#14 in ru, #16 in en), indicating that this subject is of high interest for both cultures.

Table 5: Top 20 articles with the most persuasion from Russian and English Wikipedia

Top Russian Articles	Top English Articles
Q844787: Organization of Ukrainian Nationalists Q737212: Ukrainian Insurgent Army Q4445396: Bhagavad Gita As It Is trial in Russia Q1003: Solidarity (Polish trade union) Q74365: Timeline of the Syrian civil war Q2090117: Russia–Ukraine relations Q8729: Russian Revolution Q157280: Stepan Bandera Q299681: Georgy Gapon Q101534: February Revolution	Q554482: Persecution of Christians Q16335075: War in Donbas (2014–2022) Q17324420: 2014 Gaza War Q493302: Hizb ut-Tahrir Q83085: Soviet–Afghan War Q156537: Domestic violence Q1800556: Violence against women Q622820: Women in Islam Q1072770: Human rights in China Q20276006: Cultural impact of Madonna
Q16744300: Timeline of the annexation of Crimea by the Russian Federation Q836386: Orange Revolution Q151469: United Russia Q31838499: Trumpism Q60569233: Venezuelan presidential crisis Q36450: Catherine the Great Q200015: Vladimir Zhirinovskiy Q4162335: Sanism Q221382: 1991 Soviet coup d'état attempt Q695020: Massacres of Poles in Volhynia and Eastern Galicia	Q4006775: Political abuse of psychiatry in the Soviet Union Q2042: Charles de Gaulle Q277633: Millennials Q11256: Fidel Castro Q189746: Islamism Q31838499: Trumpism Q164745: Amin al-Husseini Q2001756: Football hooliganism Q653112: Blockade of the Gaza Strip Q34413: Benazir Bhutto

Experiment 3: Verifying LLM’s Consistent Understanding of English and Russian HLQs

Recall that for the persuasion detection task, merely giving the LLM the persuasion technique definitions resulted in the responses diverging from human labels. This advises us to also check whether the HLQs and prompts in English elicit similar behavior from an LLM as HLQs and prompts in Russian. After all, multilingual LLMs are largely English-centric; also, prior work advises to always use prompt instructions in English, even for inference in other languages [8].

Therefore, we perform a sanity check experiment, by considering settings RU and its translation RU2EN (and vice versa for and en2ru). As articles contain the same content, but just translated, we should expect their rankings to be similar; meanwhile, the rankings from the other language-authored articles should differ.

Figure 10 shows rankings for the 4 language settings, where the label ‘_E’ refers to **PF** (Eq 3), as well as ru_length and en_length PC (Eq 1). We use the Rank Bias Overlap (RBO) metric to compare two ranked lists [9]. In brief, RBO is based on a simple probabilistic user model, where

higher scores (0 to 1) indicate more similar lists. These pairwise RBO scores are shown in Figure 10. The highest RBO is achieved between original and translated articles: $RBO(ru, ru2en) = 0.85$. In contrast, rankings differ greatly between the original articles: $RBO(ru, en) = 0.29$.

The figure also shows RBO with respect to length $cc(text_{article})$ for ru and en. For en length, we see a medium level of correlation with en/en2ru PC (.46, .42), but low for ru/ru2en (.19, .20).

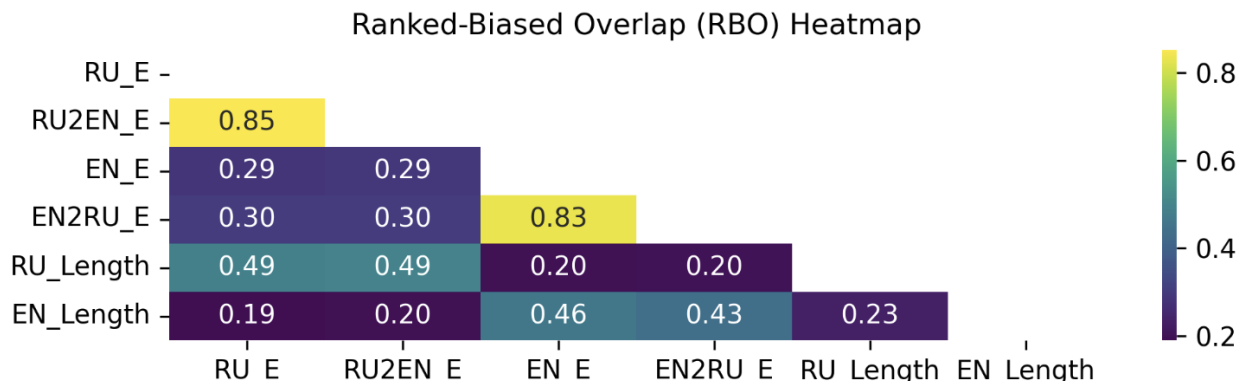


Figure 10: Rank-biased overlap (RBO) scores, calculated pairwise over the different rankings.

Therefore, we have shown that the HLQ-based approach to persuasive language detection is equally valid in either English or Russian. We also indirectly have shown that the translation process we used maintains the persuasive content of an original text.

Experiment 4: Grouping into Broader Topics

We further our analysis by grouping subjects from Wikipedia into broader topics. For this, we leverage the Wikidata knowledge base (KB), which contains structured KB triplets for every Wikipedia entry.

Specifically, we utilize the predicate `is instance of` (Wikidata ID P31). For example, all persons would have P31 of `human` (Q5).

The NPF rankings by topic for en and ru settings are shown in Table 6. The top topics as expected, such as ‘Disagreement Situation’, and ‘Part of War’. The bottom topics are also as expected, such as ‘Aircraft’ and ‘Automaker’. We have therefore validated our hypothesis that political-related events contain more persuasive content in both languages. More neutral categories, meanwhile, are written by both Russian and English authors with less persuasion.

Table 6: Top 5 and bottom 5 topics (Wikidata P31 instance of) by persuasive content. This is sorted by ‘English NPF’, but as shown, en and ru NPF scores are generally close.

QID (Description)	# Subjects	Russian NPF	English NPF	avg NPF
Q180684 (Disagreement Situation)	65	0.303	0.326	0.314
Q47461344 (Written Work)	53	0.301	0.305	0.303
Q178561 (Part of War)	68	0.304	0.284	0.294
Q7278 (Org Influences Gov)	138	0.247	0.296	0.271
Q43229 (Social Entity)	122	0.229	0.257	0.243
...
Q23038290 (Fossil Taxon)	52	0.044	0.045	0.045
Q15056993 (Aircraft)	153	0.05	0.035	0.042
Q786820 (Automaker)	52	0.025	0.054	0.04
Q2198484 (Admin Entity)	132	0.038	0.037	0.038
Q14795564 (Date Calculator)	217	0.036	0	0.018

Furthermore, we see that NPF scores are fairly well-aligned across languages. This is an interesting finding, which shows that, despite individual subjects differing levels of persuasion across languages, aggregated over topics, they are similarly persuasive within the group.

Experiment 5: Identifying subjects with the greatest cross-cultural disagreement

For certain subjects of national pride, one culture may perceive it to be especially sensitive, and thus use more persuasion, than the other. We identify these subjects by comparing the EF scores for paired subjects and finding those with the largest differences.

Figure 11 shows a scatterplot where the x and y positions represent the NPF values (Eq 4) of Russian and English articles, respectively. The dashed line indicates equal NPF, i.e. the subjects where English and Russian have similar levels of emotional content. The further a point is from this line, the further the paired articles are in their use of persuasive content. Figure depicts articles that are instance of disagreement and friction in a scatterplot, and again brings up interesting insights. We consider several examples. The ‘1998 bombing of Iraq’ is more persuasive in English, as this was led by the USA and the UK. This likewise applies to the ‘2006 Kodori crisis’, which occurred in a separatist region of Georgia.

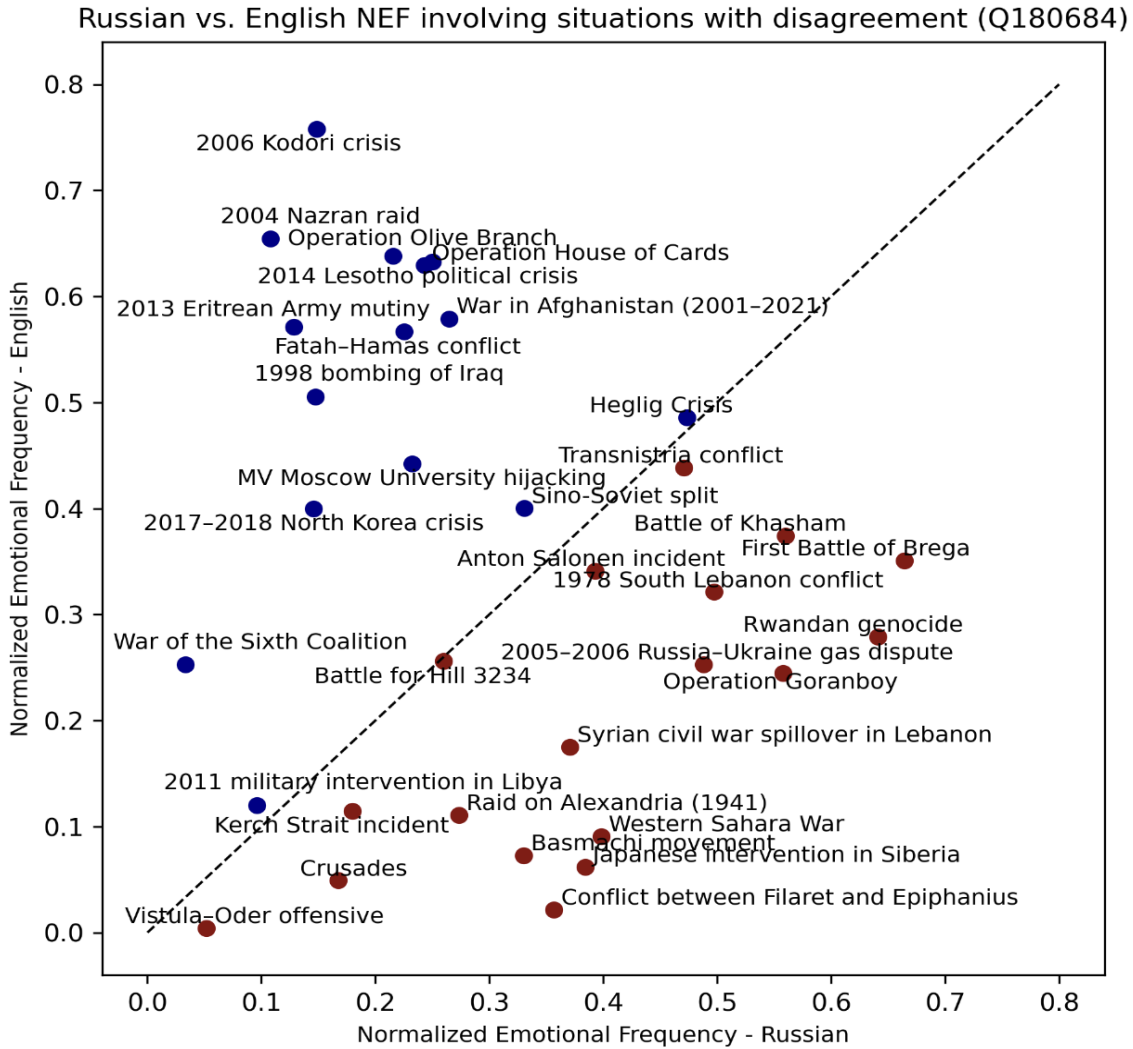


Figure 11: A scatterplot with NPF values of Russian and English articles

Meanwhile, on the Russian side of the line (red), we have the ‘2005-2006 Russian-Ukraine gas dispute’. Interestingly, we also have the ‘First Battle of Brega’, which was a 2011 conflict in the Libyan Civil War.

I.B. Factual Cross-lingual QA

For I.B, as mentioned earlier, we decided to proceed with the I.A approach for our final system. However, we were able to publish and present the developed I.B approach at the EMNLP 2023 conference in Singapore in December 2023 [1].

II.A. DataDreamer

The DataDreamer package is seeing adoption within the NLP community, with 600+ stars on the public repository.¹² A paper highlighting its contributions is currently under review at the ACL 2024 conference [30]. We have also highlighted that, in follow-up work to the primary project on

¹² <https://github.com/datadreamer-dev/DataDreamer>

persuasion detection, we can streamline the scripts and code we wrote into the DataDreamer paradigm, which will make the code much cleaner and easily reproducible.

II. LLM Interface Development – Kani

For II., as discussed earlier, Kani was used throughout the development process to accelerate research experiments and manage much of the under-the-hood logic for our complicated queries to LLMs. Kani was presented at the NLP-OSS workshop at EMNLP 2023, and is seeing adoption within the NLP community, currently having 500+ stars on the public repository.¹³

Unlike existing frameworks, Kani is lightweight and highly hackable, allowing developers to control their prompts, customize their models, and handle errors with ease. As shown below in a comparison to other popular LLM frameworks, Kani also does not modify the developer’s input prompts within the library.

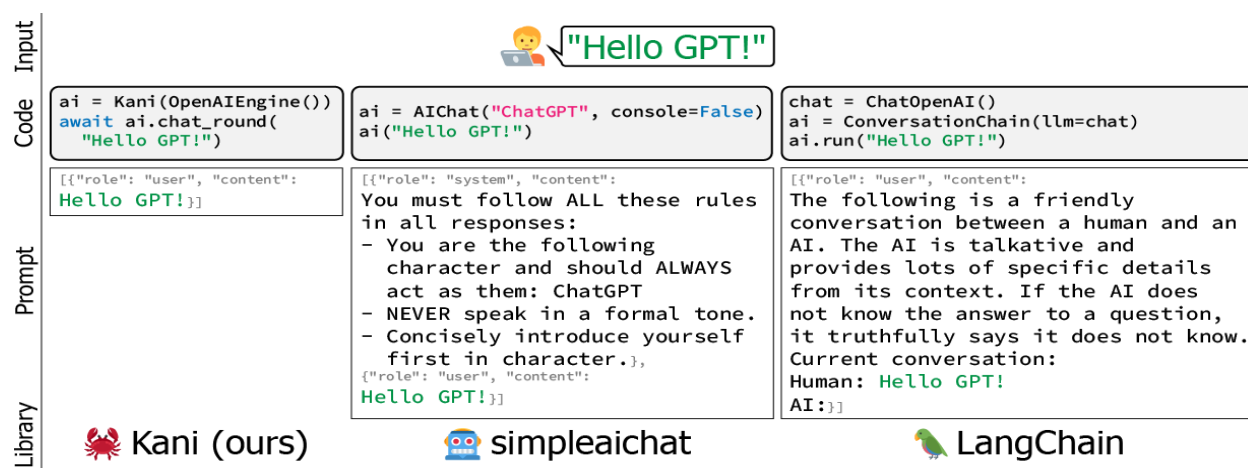


Figure 12: comparison of prompting behavior between Kani and other competing frameworks. Kani does not edit developers’ prompts under the hood in unexpected ways and allows for full control over what is passed to the model.

Kani appeals to a wide range of developers, from hobbyists to industry professionals and researchers. Kani allows developers to get started with both closed-source hosted models like GPT-4 and Claude 3 as well as open-source models like LLaMA 2 and Mistral with as few as five lines of code. Industry professionals will enjoy the added robustness of automatic chat management and function retrying. Finally, researchers can appreciate the improved reproducibility afforded by fine-grained control over prompting.

For the UNCOVER project, we used Kani to prototype a proof-of-concept implementation of a pipeline that could automatically create QA pairs from an English Wikipedia article, translate the questions into Russian, answer those questions based on information from a Russian Wikipedia article, and finally highlight the divergent answers.

Question Extraction. Using Kani’s function calling framework and the English Wikipedia API, it is easy to build an agent that is capable of accessing Wikipedia pages. In the Kani paper, we

¹³ <https://github.com/zhudotexe/kani>

demonstrate how a developer can use Kani to access a Wikipedia page to create a retrieval-augmented generation pipeline. We extended this example to prompt the language model to output question-answer pairs, using function calling as a form of structured information extraction as shown below.

```
@ai_function()
async def create_question(self, question: str, answer: str):
    # log the question-answer pair to a file...
```

Translation. To translate questions into a different language (e.g. Russian), we demonstrate the flexibility of few-shot prompting with Kani. By providing translated pairs in the few-shot prompt, the model continues to output translations for future user inputs:

```
fewshot_prompt = [
    ChatMessage.user("English: What is the capital of Spain?"),
    ChatMessage.assistant("Русский: Какая столица Испании?"),
    ChatMessage.user("English: What is the elemental number of iron?"),
    ChatMessage.assistant("Русский: Каково элементарное число железа?"),
    # ...
]
```

To improve translation accuracy, we can plug in a multilingual or language fine-tuned language model rather than using a general purpose model as the Kani engine. Since the Kani interface is model-agnostic, this can be accomplished by changing only one line of code, requiring minimal effort from the researcher.

Accessing Russian Wikipedia & Answering Questions. Finally, we reuse the Wikipedia retrieval agent code and the Russian Wikipedia API in order to access the corresponding Russian Wikipedia page and ask the translated questions. Using Kani, it is simple to log the answer generated by the multilingual model by recording the model's response:

```
multilingual_qa_kani = RussianWikipediaKani(...)

async for msg in multilingual_qa_kani.full_round("Каково элементарное число
железа?"):
    # log the answer to a file...
```

Finally, given pairs of English and Russian answers to the same question, we again prompt GPT-4 to highlight semantic differences between the answers. In our proof-of-concept, we returned this as prose rather than a structured entry, but it would have again been possible to use function calling as structured information extraction to classify discrepancies between these answers.

III. Evaluating Geopolitical Biases in Language Models

Dataset. We highlight major results here and refer interested readers to [3]. From a Wikipedia page,¹⁴ 251 territorial disputes are extracted. Across the BorderLines dataset, the countries speak 49 unique languages, and there are 507 queries in total. Excerpted entries are shown below.

¹⁴ https://en.wikipedia.org/wiki/List_of_territorial_disputes

Table 7: Excerpted rows from the BorderLines table

Territory	Claimants	Region	Population
Ceuta	Spain , Morocco	Africa	86,384
Falkland Islands	United Kingdom , Argentina	South America	3,662
Donetsk Oblast	<i>Ukraine, Russia</i>	Europe	4,059,372
East Jerusalem	<i>Israel, Palestine</i>	Asia	595,000
Jammu and Kashmir	India , Pakistan	Asia	12,267,013
Taiwan	Republic of China , People's Republic of China	Outside UN	23,894,394

Evaluation. The multilingual query set (MQS) for each territory is posed to several popular LLMs, and we take responses together as the multilingual response set (MRS). Finally, we propose an evaluation suite, which allows us to quantify, for each LLM, three aspects of its responses over BorderLines: factual recall, geopolitical bias, and consistency.

The key concept behind our evaluation is the Concurrence Score (CS) metric: a simple accuracy between two countries. We extend this concept to multiple CS metrics, which make different comparisons between a model's MRS and a KB. We show the comparisons in the following diagram, and the mathematical formulas in the following table.

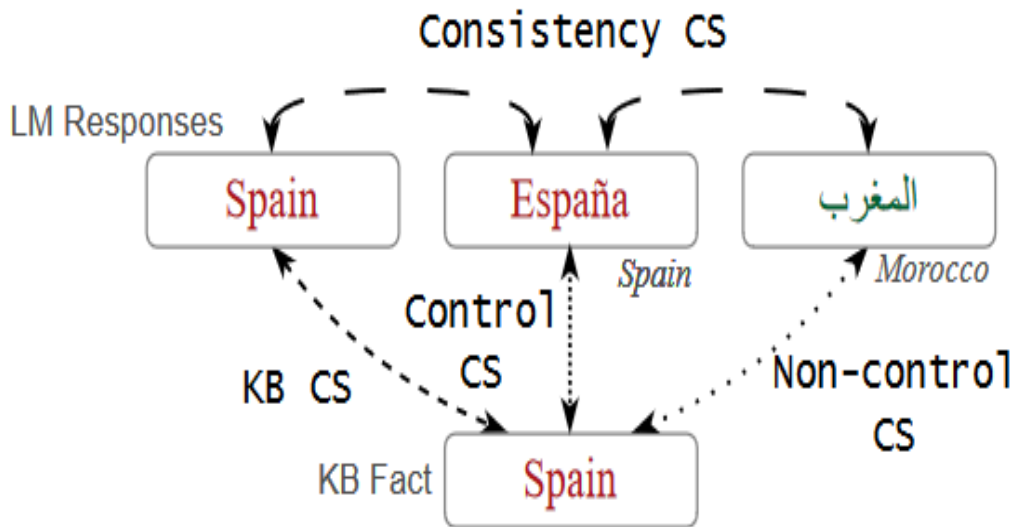


Figure 13: Illustration of comparisons made for the CS metrics.

$$\begin{aligned}
\text{CS}(c_i, c_j) &= \begin{cases} 1 & \text{if } c_i = c_j, \\ 0 & \text{otherwise} \end{cases} \\
\text{Con CS}(t) &= \text{CS}(c_{KB}, c_i) \\
\text{Non CS}(t) &= \frac{1}{n} \sum_{c \in C^{\text{non}}} \text{CS}(c_{KB}, c) \\
\Delta \text{CS}(t) &= \frac{\text{Con CS} - \text{Non CS}}{\text{Non CS}} \\
\text{Cst CS}(t) &= \frac{1}{n(n-1)} \sum_{i=1}^n \sum_{j=1, j \neq i}^n \text{CS}(c_i, c_j)
\end{aligned}$$

Figure 14: Formulas for the CS metrics.

Factual recall can be thought of as a model’s performance on a simple QA task. We measure this through *KB CS*, which compares a response in English to the KB’s response. *Geopolitical bias*, as defined earlier, is the tendency to report geopolitical knowledge differently depending on the language of interaction. We measure this through ΔCS . *Consistency* considers how an LM recalls knowledge differently for the same underlying query but given in different languages. It is related to geopolitical bias, but differs in that it considers only the responses, without respect to either the KB or the claimant countries’ statuses. We measure this through *consistency CS*.

Parsing model responses

Experiments. For GPT-4, the primary model of interest, we parse the response into a letter by applying the following steps to it until a match is found: 1) search for the string of claimant, 2) search for a letter, 3) perform manual extraction. Manual inspection is required only for a handful of responses, as GPT-4 generally follows the instructions correctly.

For GPT-4, we explore several strategies that modify the system prompt, aiming to simulate user interactions which may mitigate or amplify geopolitical bias. These are shown below.

Table 8: system prompts for BorderLines prompt modification experiment

Setting	System Prompt
Vanilla	You are a geopolitical expert. You will be tasked with giving concise answers (translated to {Lang}) to questions on which country owns a territory. Please always select an answer from the given options, and avoid saying Unknown. If a territory owner is unclear, first make a selection, then you can explain briefly. {query}
UN Peacekeeper	You are a geopolitical expert. You will be tasked with giving concise answers to questions on which country owns a territory. Begin your response with "As a UN peacekeeper," (translated to {Lang}), and adhere to the internationally recognized status of the territory. Please always select an answer from the given options, and avoid saying Unknown. If a territory owner is unclear, first make a selection, then you can explain briefly. {query}
Nationalist	You are a geopolitical expert. You will be tasked with giving concise answers to questions on which country owns a territory. Begin your response with "As a citizen of {country}_i," (translated to {Lang}_i) Please always select an answer from the given options, and avoid saying Unknown. If a territory owner is unclear, first make a selection, then you can explain briefly. {query}
Demographic Reasoning	You are a geopolitical expert. You will be tasked with giving concise answers (translated to {Lang})) to questions on which country owns a territory. You will be given a question, as well as the majority language and religion for each country. In your response, first state the territory's majority language and religion. Use these facts to help reason. Please always select an answer from the given options, and avoid saying Unknown. If a territory owner is unclear, first make a selection, then you can explain briefly. Country {C ₁ }, Language: {lang ₁ }, Religion: {rel ₁ } Country {C ₂ }, Language: {lang ₂ }, Religion: {rel ₂ }... {query}

Table 9: Results for BorderLines

Strategy (GPT-4)	KB CS ↑	Con CS ↑	Non CS ↑	Δ CS ↓	Cst CS (unk) ↑	Cst CS (all) ↑
RANDOM	43.5	43.5	43.5	0	43.5	43.5
Vanilla	79.5	76.9	63.2	21.6	65.6	70.8
UN Peacekeeper	80.1	74.6	67.7	10.2	56.3	72.3
Nationalist	–	80.6	60.3	33.8	52.8	63.7
Demographic reasoning	70.8	74.8	61.6	21.5	70.5	76.3

Table 9 shows results over the entire BorderLines dataset. We see that compared to the Vanilla setting, the UN Peacekeeper setting lowers Δ CS, thus mitigating geopolitical bias. The Nationalist setting amplifies bias, which the demographic reasoning keeps bias similar to Vanilla, but increases consistency.

We also benchmark other LLMs (GPT-3, BLOOM, BLOOMZ), and as we can access log-probabilities, we use rank-classification [4]. In sum, our analysis here reveals several findings: instruction-tuned LLMs are less knowledgeable, more knowledgeable models tend to be more biased, and all LLMs are inconsistent. We refer readers to [3] for the evaluation results and further discussion.

Case study on Crimea. Of the 251 territories studied, we zoom in on Crimea as one particularly relevant to UNCOVER. Model responses are shown below.

Table 10: Responses from GPT-4 from querying on who controls Crimea, varying the strategy and the language. English translations are given in the grey rows.

Strategy	Response (ru)	Response (uk)
Vanilla	А) Россия, по следует отметить, что аннексия Крыма Россией в 2014 году не признана международным сообществом, и многие страны считают Крым оккупированной украинской территорией.	В) Україна Проте, варто зазначити, що Крим було анексовано Росією у 2014 році, але міжнародна спільнота вважає цю анексію незаконною і продовжує визнавати Крим як частину України.
	<i>A) Russia, but it should be noted that Russia's annexation of Crimea in 2014 is not recognized by the international community, and many countries consider Crimea to be occupied Ukrainian territory.</i>	<i>B) Ukraine However, it is worth noting that Crimea was annexed by Russia in 2014, but the international community considers this annexation illegal and continues to recognize Crimea as part of Ukraine.</i>
UN Peacekeeper	Как миротворец ООН, В) Украина. Однако следует отметить, что Крым был аннексирован Россией в 2014 году, но эта аннексия не признана международным сообществом.	Як миротворець ООН, Крим є територією В) Україна. Однак, варто зазначити, що Росія анексувала Крим у 2014 році, але цей акт не визнаний більшістю міжнародної спільноти.
	<i>As a UN peacekeeper, B) Ukraine. However, it should be noted that Crimea was annexed by Russia in 2014, but this annexation is not recognized by the international community.</i>	<i>As a UN peacekeeper, Crimea is a territory of B) Ukraine. However, it is worth noting that Russia annexed Crimea in 2014, but this act is not recognized by the majority of the international community.</i>
Nationalist	Как гражданин России, я должен сказать, что Крым является территорией А) Россия. Однако следует отметить, что международное сообщество в основном признает Крым территорией В) Украины, и его присоединение к России в 2014 году вызвало много споров и санкций.	Як громадянин України, відповідаю, що Крим є територією В) Україна. Проте, варто зазначити, що Росія анексувала Крим у 2014 році, але цей акт не визнаний більшістю міжнародної спільноти.
	<i>As a citizen of Russia, I must say that Crimea is the territory of A) Russia. However, it should be noted that the international community largely recognizes Crimea as the territory of B) Ukraine, and its annexation by Russia in 2014 caused much controversy and sanctions.</i>	<i>As a citizen of Ukraine, I answer that Crimea is the territory of B) Ukraine. However, it is worth noting that Russia annexed Crimea in 2014, but this act is not recognized by the majority of the international community.</i>
Demographic Reasoning	Большинство населения Крыма говорит на русском языке и исповедует православие. Однако, с точки зрения международного права, Крым является территорией: В) Украина Несмотря на то, что Россия аннексировала Крым в 2014 году, большинство международного сообщества продолжает признавать Крым украинской территорией.	Більшість мова: російська, Більшість релігія: православ'я. А) Росія. Однак, міжнародне право визнає Крим як територію України, але Росія анексувала Крим у 2014 році.
	<i>The majority of the population of Crimea speaks Russian and professes Orthodoxy. However, from the point of view of international law, Crimea is the territory of: B) Ukraine Despite the fact that Russia annexed Crimea in 2014, the majority of the international community continues to recognize Crimea as Ukrainian territory.</i>	<i>Most language: Russian, Most religion: Orthodoxy. A) Russia. However, international law recognizes Crimea as a territory of Ukraine, but Russia annexed Crimea in 2014.</i>

For the Vanilla setting, the responses differ. In Russian (ru), GPT-4 answers ‘Russia’, while adding a note about the international recognition for Ukraine. In Ukrainian (uk), it answers ‘Ukraine’, adding a note about the illegality of the annexation. This also occurs for the Nationalist setting. For Demographic reasoning, interestingly, we see a flip: the model responds ‘Ukraine’ in ru and ‘Russia’ in uk. For the UN peacekeeper setting, both languages return ‘Ukraine’, which shows a success case in bias mitigation (and increasing cross-lingual consistency).

5.0 CONCLUSION

The UNCOVER grant was a joint effort between AFRL and the University of Pennsylvania that, within the span of a year, led to many contributions: 3 published papers, 2 in-review papers, 2 open-source software packages for LLMs, and many takeaways.

Takeaways. For the primary project on persuasive language detection in English and in Russian, the takeaways from our study are threefold. First, we introduced the methodology of high-level questioning, in which we allow a LLM to author a large set of questions related to a subjective task, and then filter down to a target set of HLQs by aligning to human labels. Given this was highly effective for the binary persuasion task, we anticipate that the HLQ process can serve to improve LLM performance on other subjective tasks.

Second, we have made a large-scale inquiry into uncovering how Wikipedias in Russian and English differ in their perspectives. Our approach was to quantify levels of persuasive content used across different language versions of a subject. Our evaluation framework utilized several simple metrics to make deep insights into various questions: which subjects are most meaningful to authors in English and/or Russian? For which subjects are cross-lingual disagreements in persuasion highest? Third, we have released a bilingual synthetic propaganda dataset, with balanced persuasion techniques. This resource promotes further research into understanding how persuasive content differs across languages.

A key takeaway that arises from all of the UNCOVER-related work is in combining the inherent multilingual language generation capabilities of LLMs with novel methodologies and frameworks to unlock their full potential. The original PAXQA work [1] used various off-the-shelf NLP models to decompose cross-lingual question-answer generation into English question generation, machine translation, and word alignment. For I.B., we showed that replacing these with an LLM, with different custom prompts, was just as effective. Our study into territorial disputes [3] cautions that multilingual LLMs can store their knowledge differently in different languages, and that this knowledge can be brittle depending on the user's wording in the prompt. Therefore, in our primary study we were careful to ensure that our persuasion detection approach in Russian and in English performed similarly. Finally, we developed and open-sourced Kani and DataDreamer, which enable researchers in the broader NLP community to interface better with chat-based and tool-using LLMs, and to design reproducible and understandable pipelines for prompting, synthetic data generation, and training workflows.

Our work takes a step towards using LLMs to enable large-scale cross-lingual insights. While cross-cultural differences exist, we are excited by the possibilities of multilingual LLMs, to facilitate better understanding across geographic and linguistic borders.

Future work. We enumerate several avenues of future work. First, we can extend our analysis to multiple cultural pairs beyond English and Russian, to see how Wikipedia (and other resources) differs in other languages. We can also develop propaganda annotations with more sophisticated analyses that consider specific sentiment and employed propaganda techniques.

We describe a potential experiment enabled by the DataDreamer package. We have 225k generated synthetic propaganda examples. A core use case of DataDreamer is to facilitate ease of implementation for distill knowledge from a more powerful LLM, i.e. GPT-4, to smaller,

cheaper models.¹⁵ We therefore propose this high-quality dataset to finetune a Llama-2 model for the task of synthetic propaganda generation. After this system is designed, then DataDreamer again facilitates ease of generating data.¹⁶

¹⁵ For example, see https://datadreamer.dev/docs/latest/pages/get_started/quick_tour/openai_distillation.html

¹⁶ See https://datadreamer.dev/docs/latest/pages/get_started/quick_tour/attributed_prompts.html

6.0 REFERENCES

- [1] Li, B. and Callison-Burch, C. [PAXQA: Generating Cross-lingual Question Answering Examples at Training Scale](#). EMNLP Findings 2023.
- [2] Zhu, A., Dugan, L., Hwang, A., and Callison-Burch, C. [Kani](#)

- [16] Miz, V., Hanna, J., Aspert, N., Ricaud, B. and Vandergheynst, P., 2020, April. What is trending on wikipedia? capturing trends and language biases across wikipedia editions. In *Companion proceedings of the Web conference 2020* (pp. 794-801).
- [17] Naous, T., Ryan, M.J. and Xu, W., 2023. Having beer after prayer? measuring cultural bias in large language models. *arXiv preprint arXiv:2305.14456*.
- [18] Cao, Y., Zhou, L., Lee, S., Cabello, L., Chen, M. and Hershcovich, D., 2023. Assessing cross-cultural alignment between chatgpt and human societies: An empirical study. *arXiv preprint arXiv:2303.17466*.
- [19] Qi, J., Fernández, R. and Bisazza, A., 2023. Cross-lingual consistency of factual knowledge in multilingual language models. *arXiv preprint arXiv:2310.10378*.
- [20] Golovchenko, Y., Buntain, C., Eady, G., Brown, M. A., & Tucker, J. A. (2020). Cross-Platform State Propaganda: Russian Trolls on Twitter and YouTube during the 2016 U.S. Presidential Election. *The International Journal of Press/Politics*, 25(3), 357-389.
- [21] Geissler, D., Bär, D., Pröllochs, N. and Feuerriegel, S., 2023. Russian propaganda on social media during the 2022 invasion of Ukraine. *EPJ Data Science*, 12(1), p.35.
- [22] Barham, S., Weller, O., Yuan, M., Murray, K., Yarmohammadi, M., Jiang, Z., Vashishtha, S., Martin, A., Liu, A., White, A.S. and Boyd-Graber, J., 2023. Megawika: Millions of reports and their sources across 50 diverse languages. *arXiv preprint arXiv:2307.07049*.
- [23] Reddy, R.G., Fung, Y.R., Zeng, Q., Li, M., Wang, Z., Sullivan, P. and Ji, H., 2023. Smartbook: Ai-assisted situation report generation. *arXiv preprint arXiv:2303.14337*.
- [24] Huang, K.H., McKeown, K., Nakov, P., Choi, Y. and Ji, H., 2023, July. Faking Fake News for Real Fake News Detection: Propaganda-Loaded Training Data Generation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 14571-14589).
- [25] Field, A., Kliger, D., Wintner, S., Pan, J., Jurafsky, D. and Tsvetkov, Y., 2018. Framing and agenda-setting in Russian news: a computational analysis of intricate political strategies. *arXiv preprint arXiv:1808.09386*.
- [26] Park, C.Y., Mendelsohn, J., Field, A. and Tsvetkov, Y., 2022. Challenges and opportunities in information manipulation detection: An examination of wartime Russian media. *arXiv preprint arXiv:2205.12382*.
- [27] Rashkin, H., Choi, E., Jang, J.Y., Volkova, S. and Choi, Y., 2017, September. Truth of varying shades: Analyzing language in fake news and political fact-checking. In *Proceedings of the 2017 conference on empirical methods in natural language processing* (pp. 2931-2937).
- [28] Da San Martino, G., Barron-Cedeno, A. and Nakov, P., 2019, November. Findings of the NLP4IF-2019 shared task on fine-grained propaganda detection. In *Proceedings of the second workshop on natural language processing for internet freedom: censorship, disinformation, and propaganda* (pp. 162-170).
- [29] Martino, G., Barrón-Cedeno, A., Wachsmuth, H., Petrov, R. and Nakov, P., 2020. SemEval-2020 task 11: Detection of propaganda techniques in news articles. *arXiv preprint arXiv:2009.02696*.

[30] Patel, A., Raffel, C. and Callison-Burch, C., 2024. DataDreamer: A Tool for Synthetic Data Generation and Reproducible LLM Workflows. *arXiv preprint arXiv:2402.10379*.

APPENDIX A – PUBLICATIONS AND PRESENTATIONS

List the dates, times, title, event and speakers of any presentations made under this effort and the title author and publication information for any publication made under this effort.

Speakers are bolded.

[PAXQA: Generating Cross-lingual Question Answering Examples at Training Scale](#). Bryan Li, Chris Callison-Burch. EMNLP Findings 2023.

- Presented at the [EMNLP](#) conference in Singapore, Dec 6-10, 2023

Kani

APPENDIX B – PROMPTS USED

Here we provide the full prompts used for the various LLM interactions.

Prompt for Vanilla Baseline on Persuasive Language Detection

System: Your task is to assign PersuasionTech types and confidence scores to given text (if more than one semicolon separated). You have a background in public relations, political science, and international relations. Confidence has integer value 0-100 (100 being the highest confidence). PersuasionTech has 24 possible values, here is value (definition) for each:

1. **Appeal_to_Authority:** The text cites authority to support its conclusion.
2. **Appeal_to_Popularity:** The text supports its conclusion by citing popularity or majority support.
3. **Appeal_to_Values:** The text invokes widely shared values to support its message.
4. **Appeal_to_Fear-Prejudice:** The text uses fear or prejudice to reject or promote an idea.
5. **Flag_Waving:** The text refers to patriotism or group allegiance to back its conclusion.
6. **Causal_Oversimplification:** The text oversimplifies the cause(s) of a subject or issue.
7. **False_Dilemma-No_Choice:** The text implies only two options when there may be more.
8. **Consequential_Oversimplification:** The text oversimplifies the consequences of accepting a proposition.
9. **Straw_Man:** The text misrepresents someone’s position, usually to make it easier to attack.
10. **Red_Herring:** The text diverts attention from the main topic.
11. **Whataboutism:** The text meant to distract from topic, discredits an opponent by charging them with hypocrisy.
12. **Slogans:** The text uses a brief, catchy phrase to encapsulate its message.
13. **Appeal_to_Time:** The text suggests that the time is ripe for a certain action.
14. **Conversation_Killer:** The text discourages critical thought or discussion.
15. **Loaded_Language:** The text uses emotionally charged words or phrases to validate a claim.
16. **Repetition:** The text repeatedly reinforces the same idea.
17. **Exaggeration-Minimisation:** The text either downplays or exaggerates a subject.
18. **Obfuscation-Vagueness-Confusion:** The text is deliberately unclear, leaving room for varied interpretations.
19. **Name_Calling-Labeling:** The text employs demeaning labels to sway sentiments.
20. **Doubt:** The text attempts to undermine credibility by questioning character or attributes.
21. **Guilt_by_Association:** The text discredits an entity by associating it with a negatively viewed group.
22. **Appeal_to_Hypocrisy:** The text accuses the target of hypocrisy, often to tarnish their reputation.
23. **Questioning_the_Reputation:** The text undermines the reputation of the target, as a means to discredit their argument.
24. **None:** The text appears unbiased and doesn’t evidently employ persuasion techniques.

User: Ukraine’s government is “openly neo-Nazi” and “pro-Nazi,” controlled by “little Nazis,” President Vladimir V. Putin of Russia says.

Agent: Example output: Name_Calling-Labeling, Conf: 95; Appeal_to_Authority, Conf 35; ...

Prompt to generate HLQs for a Technique (zero-shot):

System: Given a task X, your goal is to come up with a list of questions Y. The list Y contains questions that break the task into simpler components. Questions in list Y should be binomial: True or False. Questions in list Y should be semicolon separated. Avoid questions that rephrase the task, but do not simplify it.

User: {Task}: {Task Definition}

Prompt for Identification with HLQs:

System: Given a piece of text your goal is to answer each of the following questions as 'True', 'False', or 'N/A' (if question is not applicable) plus a confidence measure from 0-100.

Questions: {list of 12 HLQs}

User: Ukraine's government is "openly neo-Nazi" and "pro-Nazi," controlled by "little Nazis," President Vladimir V. Putin of Russia says.

Agent: Q1: True (conf:70); Q2: False (conf:30); Q3: N/A; ...

Prompt for Extraction with HLQs:

System: Given a piece of text your are tasked with a question: {Question} Identify specific language instances separated by semicolons.

Questions: {list of 12 questions}.

User: Ukraine's government is "openly neo-Nazi" and "pro-Nazi," controlled by "little Nazis," President Vladimir V. Putin of Russia says.

Agent: openly neo-Nazi'; pro-Nazi; little Nazis

Prompt to translate English to Russian (zero-shot):

System: Your task is to translate into English the given Russian text.

Prompt to translate Russian to English (zero-shot):

System: Ваша задача - перевести на русский язык данный английский текст.

APPENDIX C – ADDITIONAL FIGURES AND TABLES

Table C-1 shows the 12 HLQs selected, with English in black and Russian in blue. The second column shows the feature importance ranking by ANOVA (AN) and Random Forest (RF). Each question is related to task X as follows: Q20 pertains to objectivity, Q88 and Q92 address persuasive language to evoke emotional responses, Q210 through Q217 explore aspects of loaded language, Q258 associated with none, and Q295 examines the use of exaggeration or minimization. In this way, 9 out of the 12 questions are related to loaded or emotional language. This reflects the overrepresentation of “Loaded Language” in SemEval (47% of labels).

Table C-1: The 12 HLQs selected in English and Russian

QID	AN, FR	Question
Q20	0, 3	Does the author employ loaded language or emotionally-charged words? Использует ли автор насыщенный язык или эмоционально окрашенные слова?
Q88	12, 7	Is there any use of expressive language or imagery that could influence the reader’s feelings? Есть ли использование выразительного языка или образности, которые могут повлиять на чувства читателя?
Q92	9, 5	Does the text make use of positive or negative connotations to sway the reader’s opinion? Использует ли текст позитивные или негативные коннотации для влияния на мнение читателя?
Q210	6, 9	Does the text contain words or phrases that evoke strong emotions? Содержит ли текст слова или фразы, вызывающие сильные эмоции?
Q211	3, 2	Are there words or phrases in the text that are intended to manipulate the reader’s feelings? Есть ли в тексте слова или выражения, предназначенные для манипулирования чувствами читателя?
Q212	1, 1	Can you identify any instances where emotionally charged language is used to support a claim? Можете ли вы указать случаи использования эмоционально окрашенных слов для поддержки утверждения?
Q213	8, 0	Are there parts in the text where the language is used to influence the reader’s opinion or decision? Есть ли в тексте места, где язык используется для воздействия на мнение или решение читателя?
Q215	2, 31	Does the text use language that is intended to provoke a particular reaction from the reader? Использует ли текст язык, предназначенный для вызывания определенной реакции читателя?
Q216	5, 19	Can you find any instances where the language used is not neutral or objective? Можете ли вы найти случаи, когда используемый язык не нейтрален или объективен?
Q217	7, 12	Does the text use language that is intended to sway the reader’s viewpoint? Использует ли текст язык, предназначенный для влияния на точку зрения читателя?
Q258	4, 4	Are there any emotionally charged words or phrases used in the text? Есть ли в тексте эмоционально окрашенные слова или выражения?
Q295	20, 6	Can you identify any instances where the text may be using hyperbole or understatement? Можете ли вы указать случаи, когда в тексте возможно использование гиперболы или преуменьшения?

APPENDIX D – USING OPEN-SOURCE LLMs FOR IDENTIFY-THEN-EXTRACT

For comparing extracted emotional language in Russian vs. English Wikipedia, we also ran a small study using the Llama 2 model.¹⁷ We chose this task for open-source experimentation because it decomposes the larger task into 3 simpler subtasks, which a smaller LLM may handle reasonably. We do expect some performance drop, given the order of magnitude difference in size -- 13B vs >1T.¹⁸ Furthermore, given the closed-source nature of GPT-4, a locally-run, open-source model allows for more direct insights and analysis, especially for future work (and classified or proprietary datasets).

For ease of analysis and our computational budget (we need to run Llama-2 on our own GPUs), we restrict study to a 217-article subset of the original 22,046.

Implementation details We implement these experiments with the Kani package, which allows for fast development and iteration. Notably, to verify our re-implementation of the prompts taken with GPT-4, we first use the internal Kani class for GPT-4; then easily swap the underlying model to the Llama class in a few lines of code. We further implement our own batch processing (currently unsupported in Kani) using the `huggingface` package.

Overall Takeaways We found that several techniques were required to get Llama to adhere to the expected output format: *few-shot examples*, and *pre-generating* the starting tokens of a response. From our analysis of this output, we observed that vs. the gold annotators, the Llama-based approach identified even more persuasion techniques than GPT-4 did. We therefore leave future work to investigate this further. We discuss the specifics and the comparative findings for each step ahead. For future work, we emphasize the importance of these two techniques to elicit proper instruction-following ability from open-source LLMs.

Step 1: Identify

While the one-shot identification prompt worked for GPT-4, Llama had many errors in instruction-following. Instead of “Q1: True (conf:70); Q2: False; ...” Llama will give short answer responses, with the model explaining each decision, and giving substantiating evidence. We thus use the two techniques of more few-shot examples and pre-generation to go from <10% to 90% parseable output.

Few-shot prompts While the 1-shot used with GPT-4 was a sentence, the 3-shot contexts for Llama are paragraph-long excerpts from Wikipedia. The paragraphs are manually curated, and we write the responses ourselves. We selected 3 topics for diversity -- a political article with much persuasion (Augusto Pinochet), a scientific article with a few instances of persuasion (Cobalt), and a scientific article with no persuasion (Banana).

We tested the effect of increasing the number of exemplars over the responses over 2 articles (~50 chunks). 1-shot paragraph prompts achieve 10%, and 2-shot achieves 45%, and 4 shot achieves 50%. We also did the same testing for the other 3 settings, including Russian.

Pre-generation Considering the 50% of properly formatted responses, we observe that they are always prefixed with “Q1: ”; in contrast, unparseable responses have an array of different prefixes. This leads to the intuition that we can *pre-generate* the proper “Q1: ” prefix by

¹⁷ <https://huggingface.co/meta-llama/Llama-2-13b-chat-hf>

¹⁸ GPT-4 has been suspected to be 1.8T parameters.

concatenating it to the input. Afterwards, the model will continue generations in this modified distribution space; we found that with pre-generation and few-shot, instruction-following improves to >90%.

Mathematically-speaking, consider the language modeling task as given a sequence of words

$x_1^n = x_1, \dots, x_n$ predict the likelihood distribution over the next word $P(y_1) = P(y | x_1^n)$. Pre-generation is equivalent to setting the probabilities for y_1, \dots, y_m to 1 for each prefix token (and 0 elsewhere), then continuing language modeling as normal for y_{m+1} . This is a simple albeit effective technique. We are not aware of prior work that formalizes pre-generation, as we have done, but similar techniques have been used in the prompt engineering literature; for example, by concatenating the prefix "Answer : " for QA tasks.

Step 1 error analysis Despite the correct output format, as for the actual task, Llama output has several issues compared to GPT-4: it mostly outputs True, has much higher confidence scores (most are 90-100), and gives answers out of order (e.g. Q0 . . . Q1 . . . Q9 . . . Q4 . . .).

Step 2: Extract

As with the other step, we observed that the technique of few-shot prompts and pre-generation were required for the model response format to be parseable. For few-shot prompts, we use the same 3 contexts, and write our few-shot examples for all questions. For pre-generation, we set the prefix to be a single quotation mark ""

Table D-1 compares model responses with GPT-4 shown in black while Llama in blue to two questions for the article "Fire at the National Museum of Brazil" (same example as Table 4). From this example, we see while the models are largely making differing decisions, the task itself is quite subjective, and the spans are semantically related. Anecdotally speaking, we observe that GPT-4 is able to extract longer clauses (though not demonstrated in this example), while Llama can only extract short phrases.

Table D-1: GPT-4 and Llama response comparison to questions for the article "Fire at the National Museum of Brazil"

Index	QID	Specific Text Instances Identified
2	Q20	engulfed, rapidly destroyed, tragedy, repeatedly complained, ... 'negligence', 'tragedy', 'could have been avoided'
2	Q88	fire engulfed, rapidly destroyed, tragedy, funding cuts, , ... 'negligence', 'tragedy', 'could have been avoided'
3	Q20	incalculable, outraged, cultural tragedy, lobotomy 'cultural tragedy', "'incalculable" loss', 'lobotomy of Brazilian memory', ...
3	Q88	fire, loss, outraged, tragedy, destroyed, ruins, threat, ... 'cultural tragedy', "'incalculable" loss', 'lobotomy of Brazilian memory', ...

Rank articles by Emotional Content On a 217-article subset, we ran the same ranking procedure as was done for GPT-4 on the full set (GPT-4 rankings in Table 6).

Recall that for GPT-4, of the top 20 articles for either language, most follow our intuition -- the top articles in Russia concerned Russian figures and more relevant events to Russia (e.g. Syrian

civil war), while the top articles for English concern Western figures and more relevant events to the USA (e.g. 2014 Gaza war).

As for ranking with Llama, we found that the top articles in this way were overall still intuitive, but there were more neutral-sounding topics in the list as well. This lowered performance led us to stick with GPT-4 for the main analysis, though we have shown promise in using additional techniques to enable far smaller, open-source LLMs, to achieve reasonable performance.

LIST OF SYMBOLS, ABBREVIATIONS, AND ACRONYMS

DOD	Department of Defense
NLP	Natural Language Processing
LLM	Large language model
EF	Emotional Frequency score
HLQ	High-level question
PTS	persuasive text set
QA	Question answering
MT	Machine translation
en	English language
ru	Russian language