big data and cognitive computing

MDPI

Article

Large Language Models as Kuwaiti Annotators

Hana Alostad

Systems and Software Development Department, Institute for Scientific Research, P.O. Box 24885, Safat 13109, Kuwait; hostad@kisr.edu.kw

Abstract: Stance detection for low-resource languages, such as the Kuwaiti dialect, poses a significant challenge in natural language processing (NLP) due to the scarcity of annotated datasets and specialized tools. This study addresses these limitations by evaluating the effectiveness of open large language models (LLMs) in automating stance detection through zero-shot and few-shot prompt engineering, with a focus on the potential of open-source models to achieve performance levels comparable to those of closed-source alternatives. We also highlight the critical distinctions between zero- and few-shot learning, emphasizing their significance for addressing the challenges posed by low-resource languages. Our evaluation involved testing 11 LLMs on a manually labeled dataset of social media posts, including GPT-4o, Gemini Pro 1.5, Mistral-Large, Jais-30B, and AYA-23. As expected, closed-source models such as GPT-4o, Gemini Pro 1.5, and Mistral-Large demonstrated superior performance, achieving maximum F1 scores of 95.4%, 95.0%, and 93.2%, respectively, in few-shot scenarios with English as the prompt template language. However, open-source models such as Jais-30B and AYA-23 achieved competitive results, with maximum F1 scores of 93.0% and 93.1%, respectively, under the same conditions. Furthermore, statistical analysis using ANOVA and Tukey's HSD post hoc tests revealed no significant differences in overall performance among GPT-4o, Gemini Pro 1.5, Mistral-Large, Jais-30B, and AYA-23. This finding underscores the potential of open-source LLMs as cost-effective and privacy-preserving alternatives for low-resource language annotation. This is the first study comparing LLMs for stance detection in the Kuwaiti dialect. Our findings highlight the importance of prompt design and model consistency in improving the quality of annotations and pave the way for NLP solutions for under-represented Arabic dialects.

Keywords: data annotation; large language models; zero-shot; few-shot; Arabic; Kuwaiti dialect

check for updates

Academic Editor: Alberto Abelló

Received: 17 December 2024
Revised: 30 January 2025
Accepted: 3 February 2025
Published: 8 February 2025

Citation: Alostad, H. Large Language Models as Kuwaiti Annotators. *Big Data Cogn. Comput.* **2025**, *9*, 33. https://doi.org/10.3390/bdcc9020033

Copyright: © 2025 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Stance detection in natural language processing (NLP) is a technique aimed at identifying the attitude or position expressed by a writer towards a specific target or topic. This process typically classifies the stance into three categories—Favor, Against, or Neither—which allows for an understanding of opinions expressed in text [1,2]. The significance of stance detection has grown in various applications—particularly in social media analysis—and it can be valuable in several domains, including politics, social science, public opinion research, and public health, where it serves to gauge public sentiment on issues ranging from political opinions [3–6] to health-related topics, such as vaccination attitudes during the COVID-19 pandemic [7–9].

While there are an estimated 422 million Arabic speakers, encompassing various dialects [10], there remains a scarcity of research on Arabic stance detection [11–15]. Many researchers, particularly when considering low-resource language contexts, encounter

challenges related to obtaining the extensive, high-quality annotated datasets needed to train robust machine learning models [16]. Creating training datasets involves human annotation, which is labor-intensive, requires comprehensive training to ensure annotator reliability, and can incur substantial costs.

Researchers have shown a growing interest in automating data annotation tasks in recent years through the use of zero-shot and few-shot prompt engineering with large language models (LLMs); examples of such studies include [17–22]. These authors have proposed various approaches for the automation of stance data annotation for different English language datasets.

A few research publications have explored the use of large language models to automate Arabic data annotation [23,24]. Most of these studies have focused on datasets from certain Arabic countries, with only two explicitly addressing automated text annotation of datasets containing tweets written in Kuwaiti dialect [11,25]. Both studies implemented weakly supervised learning approaches and utilized only zero-shot learning with encoder-based LLMs to automate text annotation for the NLP tasks of stance detection and sentiment classification. This highlights the need for further research to fill the gap in this field and address the challenges of stance detection in low-resource languages, specifically Arabic and Kuwaiti dialects.

In this study, we aim to evaluate and assess the capabilities of closed- and open-source LLMs, including GPT-4o [26], Mistral-Large-latest [27], Gemini-pro [28], Jais-30B [29], BLOOMz [30], AYA-23 [31], AYA-101 [32], AceGPT-32B [33], and mT0 [30], to automate the stance annotation task in both zero- and few-shot settings, specifically considering the Kuwaiti Dialect—a variant of the Arabic language primarily used in conversations between Kuwaiti people. We also aim to investigate whether changing the experimental setup and prompts with free open-source LLMs can result in a performance as effective as that obtained with paid closed-source models for stance detection regarding the Kuwaiti dialect, such that they can be utilized in the future to create an automated dataset annotation system that works efficiently with various Kuwaiti datasets.

The key contributions of this research are the following:

1. This study represents the first attempt to investigate the efficiency of stance detection for the Kuwaiti Dialect using zero- and few-shot prompts on closed-source generative LLMs, in comparison with other available open-source generative LLMs.
2. The study results demonstrate that using few-shot prompts with open-source LLMs for stance detection can achieve a performance comparable to that of paid closed-source models when applied to the Kuwaiti dialect vaccine stance dataset, offering significant cost savings and data privacy benefits.
3. This study compares model performance, focusing on consistency and reliability. Using statistical methods such as one-way ANOVA and Tukey's HSD, we identified significant differences and determined which models are most stable and reliable.
4. The study results are valuable for expanding training datasets, enhancing automated labeling systems, and filling the gaps in NLP research across various domains—particularly with respect to the Kuwaiti dialect—underscoring the broad impact of the research.

The rest of the paper is organized as follows: Section 2 provides information on multi-lingual transformer-based large language models (LLMs) and reviews some of the previous and current research on the use of LLMs as annotators. Section 3 describes our methodology. We start by introducing the dataset used in our experiments. Then, we present the pre-processing steps, including how we selected the LLMs and designed the prompt templates. We conclude this section with details of our experiments using LLMs as Kuwaiti annotators. In Section 4, we present the results of our study, which includes

performance comparisons across different LLMs, an open versus closed LLM analysis, statistical significance tests, an examination of response consistency, and error analysis. Finally, in Section 5, we conclude the paper by summarizing our findings and discussing possible directions for future research.

2. Background

2.1. Large Language Models (LLMs)

Natural Language Processing (NLP) has been dramatically influenced by transformer architecture, primarily due to various crucial factors. One notable reason for its impact is its proficient capability to effectively comprehend extensive dependencies within textual data—a challenge that previous models generally failed to cover. This proficiency is achieved through the self-attention mechanisms used in transformers, which enable the model to focus on distinct segments of the input sequence simultaneously. Consequently, the model acquires the ability to grasp contextual information and dependencies across the entire sequence [34]. The architecture of a transformer is made up of encoder and decoder pipelines, where the encoder is used to process input sequences and the decoder is used to process output sequences [35]. In addition to encoder- and decoder-based LLMs, the use of the transformer architecture enables further models consisting of an encoder and a decoder. In this configuration, the encoder processes the input sequence and produces a fixed-length representation, which is subsequently utilized as the initial hidden state of the decoder. Subsequently, the decoder generates the output sequence based on this initial hidden state and the encoded information [36]. Models consisting of an encoder and a decoder are commonly used for text-to-text tasks such as machine translation, where the input sequence in one language is rendered into an output sequence in another language [36].

Bidirectional Encoder Representations from Transformers (BERT) is an example of an encoder-based large language model, which employs a masked input token training approach to evaluate bidirectional information [37]. While BERT was initially designed for English content, it has undergone fine-tuning to become a multi-lingual model which is suitable for monolingual and multi-lingual tasks [38].

Decoder-based models, on the other hand, include the well-known OpenAI GPT (Generative Pre-trained Transformer). This model utilizes auto-regressive language modeling to enable the generation of sequential text [39]. GPT is a multi-lingual model that also supports the Arabic language [40]. It is important to note that OpenAI models are closed-source and not free. Therefore, developers must pay to use or fine-tune GPT models based on the type of model used and the number of tokens used per request [41].

The Google Gemini closed-source models represent a class of decoder-based architectures refined for enhanced performance and structural efficiency. The Gemini API provides a variety of models tailored for distinct applications. The latest version—Gemini 1.5 Flash—is a multi-modal model that can process textual input alongside various audio and visual inputs such as natural images, charts, screenshots, PDFs, and videos, which is optimized for fast and versatile performance and can address a diverse variety of tasks. At the same time, Gemini 1.5 Pro is a mid-sized multi-modal model optimized for an extensive range of reasoning activities, including code and text generation, text modification, problem-solving, and data extraction and generation. Gemini 1.0 Pro is a natural language processing model that manages tasks such as multi-turn dialogues in text and code generation [28,42]. Gemini models are offered on a pay-as-you-go basis, with pricing structures and usage limits differing across tiers and varying with the used model [43,44].

Mistral is also a multi-lingual, decoder-based LLM developed by Mistral AI. The latest version, Mistral Large 2, is available under the Mistral Research License for non-commercial research purposes. This means that researchers can use and fine-tune the model for free.

However, for commercial use, a Mistral Commercial License must be purchased. Depending on the deployment option chosen, users must pay for the platform or set up on-premises infrastructure. These new licenses differ from the earlier versions of Mistral models, released under the free Apache 2.0 open-source license for everyone to customize and deploy where they want [27].

Core42's open models, designated as Jais, comprise a collection of decoder-based architecture models with up to 70 billion parameters, enhancing their performance and efficiency in various tasks including summarization, comprehension, and text generation. Jais models are bilingual and demonstrate fluency in both Arabic and English. The models have been specifically designed to grasp the linguistic nuances and cultural sensitivities inherent to the Arabic language [45].

The AceGPT model is explicitly designed for Arabic. It is built on the LLaMA2 architecture and is open-source. The model has been fine-tuned, using Reinforcement Learning from AI Feedback (RLAIF), to align with local values and culture through incorporating Arabic texts and culturally relevant instructions [33].

BLOOMz (BigScience Language Open-science Open-access Multilingual) refers to a group of decoder-based architecture models that have been pre-trained on multi-lingual language models. These models are capable of understanding and following human instructions in dozens of languages. To enhance their ability to follow instructions in multiple languages, the models underwent multi-task fine-tuning on top of BLOOMz [30].

The mT0 and mT0-mt models, built by BigScience using the T5 encoder–decoder transformer architecture, were designed for multi-lingual tasks. These models were trained on diverse, multi-lingual datasets to support various languages. Both models are open-source and can handle over 50 languages; mT0 models are fine-tuned on multi-lingual datasets with English prompts, while mT0-mt models are fine-tuned on multi-lingual datasets with English and machine-translated prompts [30].

Finally, the Aya-101 model is a massively multi-lingual open-source LLM that was a result of a global initiative led by Cohere For AI; in particular, Aya-101 was built by fine-tuning the 13B parameter mT5 model [32]. In addition, Aya-23 is a multi-lingual open-source LLM that builds on the Aya model. It covers 23 languages and is designed to balance breadth and depth. Unlike Aya-101, which covers 101 languages, Aya-23 allocates more capacity to fewer languages during pre-training, significantly improving tasks for those 23 languages [31]. A summary of these LLMs is presented in the Appendix A (Table A1).

2.2. Arabic Stance Detection with LLMs

The researchers in [14] conducted a preliminary study on the detection of authorities' stances toward rumors in Arabic tweets. They constructed and released the first Authority STance towards Rumors (AuSTR) dataset. Their work fine-tuned ARBERT [46] to identify stances in rumor-related discourse.

Additionally, in [47], the researchers developed a multi-lingual LLM which is specialized in analyzing news and social media content. Its fine-tuning for stance detection tasks showcased the potential of multi-lingual LLMs to adapt to domain-specific challenges, especially for under-represented languages such as Arabic [47].

The StanceEval 2024 Shared Task [48], as part of the ArabicNLP 2024 program, utilized the MAWQIF [13] dataset, which consists of Arabic tweets annotated for stance, sentiment, and sarcasm. The shared task invited participants to develop models capable of detecting stances toward three topics: COVID-19 vaccines, digital transformation, and women's empowerment. The best-performing system, developed by AlexUNLP-BH [49], achieved a Macro F1 score of 84.38 through fine-tuning AraBERT variants, applying multi-task learning (MTL), and leveraging ensemble methods. The second- and third-place systems,

MGKM [50] and StanceCrafters [51], achieved Macro F1 scores of 82.06 and 81.68, respectively, using fine-tuning and MTL-based strategies [48].

Complementary to these efforts, specialized LLMs and benchmarking initiatives have emerged. The researchers in [24] introduced an Arabic-specific benchmark to evaluate LLMs, highlighting their performance across various NLP tasks including stance detection [24].

2.3. Kuwaiti Dialect NLP Research

Very few NLP studies have focused on the Kuwaiti dialect. Those studies in this line have explored various NLP tasks, including sentiment analysis, opinion mining, and stance detection. The research conducted by [52] represents the first contribution to natural language processing for Kuwaiti dialect. This study introduced an approach for opinion mining in microblogging, focusing on sentiment extraction from 340,000 manually labeled Twitter posts. The researchers experimented with machine learning classifiers, including J48 [53], ADTREE [54], Random Tree [55], and Support Vector Machines (SVM) [56], to classify sentiments into positive, negative, and neutral categories. The SVM classifier achieved a precision of 76%, a recall of 61%, and an F1 score of 69% for positive sentiments; in comparison, it attained a precision of 65%, a recall of 78%, and an F1 score of 71% for negative sentiments.

In the study [25], the researchers addressed the lack of resources for the Kuwaiti dialect (specifically for sentiment analysis) through proposing a weakly supervised transfer learning approach to automate dataset labeling. Their methodology involved collecting over 16,600 tweets spanning various themes and timeframes to mitigate content bias. The dataset comprised 7905 negative, 7902 positive, and 860 neutral tweets. The authors employed weak supervised learning, integrating rule-based methods in a zero-shot learning mode on pre-trained language models to assign sentiment labels. This approach reduced the reliance on extensive human annotation. For evaluation, the dataset was tested using various traditional machine learning classifiers and advanced deep learning language models, and was compared with fine-tuned Arabic BERT models, including AraBERT [57], ARBERT [46], MARBERT [46], Microsoft Multilingual Model (MiniLM) [58], and CAMeLBERT [59]. The ARBERT model achieved the highest accuracy at 89% on the testing dataset, demonstrating the effectiveness of the proposed weak supervised approach in creating reliable sentiment analysis resources for the Kuwaiti dialect.

The researchers in [11] introduced Q8VaxStance, a dataset labeling system created to detect stances toward vaccines in tweets written in the Kuwaiti dialect. This study also addressed the issue of linguistic resources for the Kuwaiti dialect, which are currently limited. Researchers collected 42,815 unlabeled tweets related to vaccines, posted between December 2020 and July 2022. To annotate the dataset, the study employed a combination of weak supervised learning and zero-shot learning techniques. A total of 52 experiments were conducted using various labeling function configurations, including keyword detection and zero-shot models. The results demonstrated that merging keyword detection with zero-shot learning significantly enhanced the model performance. This study's best-performing configurations achieved the highest Macro F1 score of 83%.

2.4. LLMs as Data Annotators

Data annotation is the process of labeling raw data with relevant information [60], which is an essential component of AI projects as it significantly influences the effectiveness of supervised machine learning models. Manual annotation can be time-consuming and expensive, often resulting in low-quality labels. To address these challenges, research has introduced automated annotation methods that employ LLMs as annotators, utilizing zero-

and few-shot in-context learning strategies. LLMs have attracted considerable attention due to their ability to perform as zero- and few-shot learners. Through specific prompting techniques, a diverse array of tasks can be achieved by offering these pre-trained models either descriptive instructions (referred to as "zero-shot") or a limited number of examples along with instructions (known as "few-shot"). Once provided with a prompt, the model can generate various outputs based on the given instructions, ranging from text and code to images, videos, or audio, depending on the specific model used [61–65]. Prompting techniques enable the rapid generation of outputs from pre-trained models for new tasks without fine-tuning. In contrast, fine-tuning a pre-trained model requires re-training it to learn a new task or domain, which demands significant computing resources and a large labeled dataset. This process is both time-consuming and expensive. Utilizing prompt engineering, we can substantially save time and resources. Several studies have explored the use of LLMs as annotators in various domains, such as medicine, finance, and cybersecurity. For instance, the researchers in [66] proposed combining LLMs with human expertise to generate ground-truth labels for efficient medical text annotation. In the financial domain, [67] investigated the potential of LLMs as efficient data annotators for the extraction of relations from financial documents. In the cybersecurity domain, [68] compared the effectiveness of LLMs in detecting phishing URLs when utilized with prompt engineering techniques versus when fine-tuned.

Most of the research in this field has focused on English datasets, with limited exploration and evaluation of LLMs as annotators for low-resource languages. For example, in [69], the researchers proposed a methodology to use LLMs as annotators for Named Entity Recognition (NER) tasks in low-resource language contexts, such as African languages. Similarly, in [23], the researchers evaluated the performance of LLMs (i.e., using GPT-4) for fine-grained propaganda detection from Arabic text. They developed the ArPro dataset, which consists of 8000 paragraphs from newspaper articles labeled at the text span level following a taxonomy of 23 propagandistic techniques.

Additionally, [24] proposed LAraBench to address the gap for Arabic NLP and speech processing tasks, utilizing and evaluating multi-lingual and Arabic LLM models and zero- and few-shot learning techniques across 61 publicly available datasets.

In the context of the Kuwaiti dialect, only two papers have explicitly addressed automated text annotation of Kuwaiti dialect datasets [11,25]. Both studies implemented weakly supervised learning approaches and utilized zero-shot learning based on encoder models to automate text annotation for the NLP tasks of stance detection and sentiment classification.

Based on the above, the field of Kuwaiti dialect NLP remains under-explored compared to Arabic MSA and other well-resourced Arabic dialects. The primary challenges include the limited availability of annotated datasets, the informal nature of the dialect, and the lack of dedicated tools or benchmarks. Future research should focus on expanding datasets and leveraging state-of-the-art techniques, such as large language models, to improve the adaptability and accuracy of NLP tools for the Kuwaiti dialect. Exploring other tasks such as machine translation, named entity recognition, and conversational AI in Kuwaiti Arabic could further enhance the field.

Consequently, there is a clear need for additional research efforts to bridge the existing research gaps in stance detection for Kuwaiti dialect. Addressing this gap, our research builds upon prior studies [11,25] through focusing on generative LLMs. We experiment with both zero- and few-shot prompting techniques, evaluating closed-source generative LLMs alongside open-source alternatives to assess their efficiency in stance detection for the Kuwaiti dialect.

3. Methodology

3.1. Dataset

We utilized the test dataset Q8Stance from the research paper [11], which consists of 519 human-labeled tweets that were collected using the Twitter academic API with a query that included keywords and hashtags related to COVID-19 vaccination. The tweets collected were written in the Kuwaiti dialect, and researchers ensured this by only including Arabic tweets with locations specific to Kuwait. The dataset was then given to Kuwaiti native speakers to annotate the tweets as either pro-vaccine or anti-vaccine. The final labeled dataset includes 265 tweets that are anti-vaccine and 254 tweets that are pro-vaccine. Examples of human-annotated tweets are presented in Table 1.

Table 1. Sample human annotated test data (Q8Stance).

Tweet in Kuwiati Dialect	Tweet in English	Label
المفروض مايخلطون بين المطعمين واللي مو مطعمين لازم يفصلونهم ويميزون المطعمين وبالنسبه للقرارات ما اعتبرها سذاجه اكثر حالات الوفاة انزلت خلال شهر بسبب التطعيم نعم للتطعيم نعم للتطعيم	They are not supposed to confuse those who are vaccinated with those who are not vaccinated. They must separate them and distinguish between the two vaccinated people. As for the decisions, I do not consider them naive. Most deaths occurred within a month due to vaccination. Yes to vaccination. Yes to vaccination.	Pro-vaccine
تم التطعيم اشكر وزارة الصحة وكل من يعمل فيها ولها من اجل صحة الكويت ، تنظيم راقي جدا	I have been vaccinated. I thank the Ministry of Health and everyone who works there and for it for the health of Kuwait, a very sophisticated organization	Pro-vaccine
الحل للوصول للمناعة المجتمعية الي من خلالها نقدر نرد حق حياتنا الطبيعية اهوا التطعيم اذا ماسجلت بعدك للتطعيم سجل، واذا عندك تساؤل أو تخوّف تقدر تسأل المختصين والكل مستعد يجاوبك نعم للتطعيم	The solution to achieving community immunity through which we can regain the right to our normal lives is vaccination. If you have not yet registered for vaccination, register, and if you have a question or fear, you can ask the specialists and everyone is ready to answer you yes to vaccination	Pro-vaccine
لا تحملوا المواطنين فشلكم الذريع لا للتطعيم الاجباري	Do not blame the citizens for your miserable failure, no to compulsory vaccination	Anti-vaccine
هل يعقل ونحن في دولة تملك دستور وبرلمان وديمقراطية وعلاوة على ذلك معروف عنها إنها ببلد الإنسانية ويجبر شعبها على التطعيم وعليه لا للتطعيم الاجباري	Is it possible that we are in a country that has a constitution, parliament, and democracy? Furthermore, it is known to be a country of humanity and its people are forced to be vaccinated! So, no to compulsory vaccination.	Anti-vaccine
مستمرين ساحة الإرادة لا للتطعيم الاجباري	Continuing the arena of will, no to compulsory vaccination	Anti-vaccine

3.2. Pre-Processing

We started by determining the selection criteria for the closed and open LLMs for use in our research experiments. We also designed the prompt templates to be used to

generate the zero- and few-shot prompts and then implemented them on the test dataset. The following sub-sections provide detailed information about each of these processes.

3.2.1. Selection of Large Language Models (LLMs)

To effectively manage our limited budget and computing resources, we established specific selection criteria for the LLMs we planned to evaluate. Our selection was based on four key factors: support for the Arabic language, model performance, parameter size, and model accessibility. These criteria guided our choices among the available LLMs, resulting in the selection of 11 models; namely, 3 proprietary closed-source and 8 open-source LLMS.

First, for the proprietary closed-source LLMs, we chose OpenAI's GPT-4o. OpenAI is a major player in generative artificial intelligence, where GPT-4o is an autoregressive, omni-multi-lingual model capable of processing various input types, such as text, audio, images, and video, generating outputs in any of these formats. This model was pre-trained using data from various sources up to October 2023 [26].

Next, we selected Google Gemini—a key competitor to OpenAI. Gemini is a set of multi-lingual multi-modal models trained jointly on image, audio, video, and text data. We used the Gemini-1.5-pro version in our experiments [28].

The third selected proprietary LLM was Mistral Large 2, which allows for usage and modification for research and non-commercial purposes. However, a Mistral Commercial License must be obtained for any commercial use [27]. We limited our selection of closed-source models to three models, due to our budgetary constraints and the insufficient support for the Arabic language in many other LLMs.

To select suitable open-source large language models (LLMs) that support Arabic, we evaluated several options and chose the following:

From BigScience, we selected the multi-lingual LLMs bloomz-7b1-mt, bloomz-7b1, mT0-xxl-mt, and mT0-xxl [30].

From Cohere, we opted for AYA-101 (13 billion parameters) and AYA-23 (35 billion parameters), which are known for their effectiveness in Arabic NLP tasks [31,32].

From Core42, we chose jais-30B, a bilingual model for Arabic and English, specifically using jais-30b-chat-v3 [29].

We also included AceGPT-v2-32B [33,70], a bilingual model based on LLaMA 2 and trained on Arabic and English datasets. To ensure cultural alignment, the model incorporated reinforcement learning from an AI Feedback reward model using localized data based on GPT-4 preferences. The appendix (Table A1) presents a summary of the selected LLMs.

3.2.2. Design of Prompt Templates

We developed 24 prompt templates covering zero- and few-shot learning contexts. Our initial step involved creating three primary zero-shot prompt templates in the Modern Standard Arabic (MSA) language, describing the task and specifying the expected output labels; in the third template, we added the role of the LLM as an "expert text annotator" as extra information. Later, these prompt templates were used with the tweets/posts written in the Kuwaiti dialect; we then translated the zero-shot prompt templates from Arabic (MSA) into English templates. In this case, we named the template as a mixed language, as it will use English for the prompt instructions and the Kuwaiti Arabic dialect for the content of the post. Figure 1 illustrates the three main Arabic and English (Mixed) prompt templates.

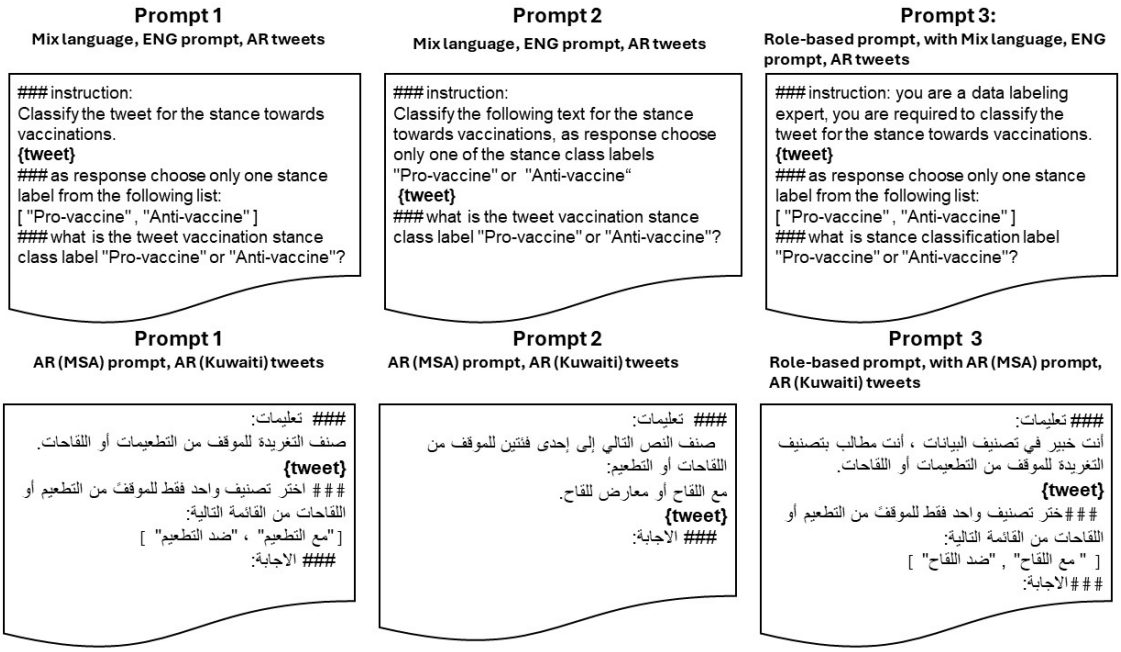


Figure 1. Arabic and English prompt templates.

Next, to create the few-shot prompt templates, we first needed to select six tweets as examples that were not part of the Q8Stance test dataset. We accomplished this by manually selecting examples from the training dataset and making sure that they covered both pro- and anti-vaccine stances. These examples provided context for the task and were used to develop 2-, 4-, and 6-shot prompt templates. We limited the number of examples to six to avoid exceeding the maximum context size limit of the language models and incurring additional costs, especially with respect to the paid closed models.

3.3. LLMs as Kuwaiti Annotator Experiments

The steps we followed to execute the experiments are detailed below.

1. Dataset Preparation:
   - From the Q8Stance test dataset, retrieve text from each tweet.
   - For the few-shot prompts, also retrieve six example tweets covering pro- and anti-vaccine stances.
2. Prompt Template Generation:
   - Generate zero-shot prompts by applying the zero-shot prompt templates on each row from the Q8Stance test dataset. These prompts serve as the input for LLM inference, guiding it to complete the given annotation task.
   - Generate few-shot prompts using example tweets through applying the few-shot prompt templates combined with the few-shot examples on each row from the Q8Stance test dataset.
3. For each generated prompt, call the LLM API to execute the instructions in the prompt.
4. Store the completion results from the LLM.
5. Post-processing Steps: After receiving the completion results from the LLM, we apply post-processing steps to enable comparison with the human-labeled test dataset. This involves filtering keywords from the results and mapping them to related labels. This step is necessary as the LLMs sometimes do not follow the exact instructions. Instead of returning only the label, they produce results that are not directly comparable to the human-annotated labels. For example, they may return labels while adding extra text or return variations in Arabic labels.

6. Store the final LLM results.
7. To evaluate the performance of the LLMs, we compared the final results of the LLM with human-annotated labels from the Q8VaxStance test dataset. For the evaluation metrics, we utilized the Macro F1 formula. Additionally, we conducted a one-way ANOVA and Tukey's HSD post hoc tests in order to determine whether the experimental results were statistically significant. We chose one-way ANOVA as it is a robust statistical method for comparing the means of multiple groups, allowing us to determine whether there are statistically significant differences among them. In this study, the groups represent the performances of different models (or conditions), while the dependent variable is the performance metric, specifically the Macro F1 score. One-way ANOVA enabled us to test the null hypothesis that all model performances were equal, thereby revealing any significant differences among the groups. After identifying significant differences using ANOVA, we applied Tukey's HSD (Honestly Significant Difference) test as a post hoc analysis to determine which specific groups (models) differ from each other. To further analyze the consistency of the responses generated by the LLMs, we assessed the variability in their performance across different prompts and settings. A model that produces consistent results will exhibit low variance in its Macro F1 scores across various scenarios, while a model with high variance would be considered less reliable. We calculated the variance [71] in Macro F1 scores across different prompts for each LLM model using the variance formula variance Formula (1):

$$\sigma^2 = \frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2, \tag{1}$$

where

- σ^2 is the variance;
- N is the total number of Macro F1 scores (or observations);
- x_i represents each individual Macro F1 score;
- μ is the mean of the Macro F1 scores.

After variance calculation, we checked the variability across models, and models with lower variance in their Macro F1 scores were considered more consistent.

4. Results

We implemented the experiments using the Python programming language, with most of the execution carried out on Google Colab Pro+. However, for the Jais-30B, AYA 23, and AceGPT32 models, we utilized an hourly GPU instance from RunPod [72], as these models required more than 40 GB of GPU VRAM for loading and inference. We applied the steps in Figure 2 for each selected LLM model. To compare the performance of the LLMs, we focused on different scenarios based on various criteria, such as the following:

- Prompt Type: zero- versus few-shot setting;
- Prompt Language: Arabic (MSA) prompts versus English prompts;
- LLM source: closed- versus open-source LLMs.

Furthermore, to ensure a comprehensive performance analysis, we conducted a one-way ANOVA and Tukey's HSD post hoc tests to determine whether the observed differences in the experimental results were statistically significant. Additionally, we assessed the LLM response consistency to evaluate their variability in performance.

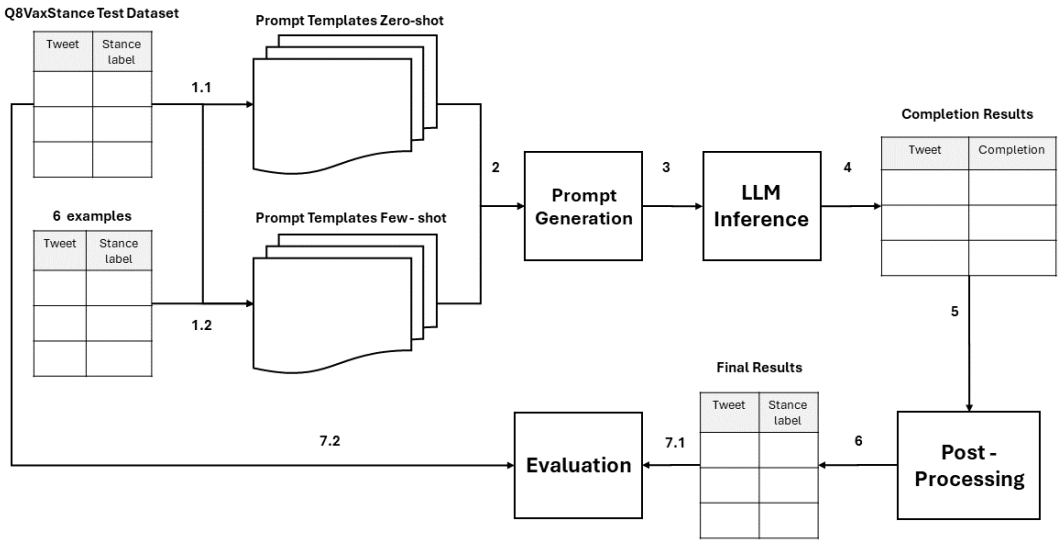


Figure 2. LLMs as Kuwaiti annotators: experimental steps.

4.1. Performance Comparison Across LLMs

Examining the overall performance of the LLMs, as illustrated in Table 2 and Figure 3, and the performance comparison of the LLMs by prompt language (Arabic vs. Mixed) plus prompt type (zero- vs. few-shot), as illustrated in Figure 4, uncovered several key findings:

The best-performing large language models (LLMs) were GPT-4o and Gemini-pro 1.5, both achieving a mean Macro F1 score of 0.94. These models consistently maintained high Macro F1 scores across various testing conditions. They excel when handling mixed-language prompts, such as English instructions combined with tweets in the Kuwaiti dialect and few-shot examples.

GPT-4o achieved the highest Macro F1 score of 0.954 when using two-shot examples (Mix-2FS), while Gemini-pro 1.5 reached a high Macro F1 score of 0.950 with four-shot examples (Mix-4FS). Both LLMs performed strongly in zero-shot settings when using Arabic or English prompts. GPT-4o scored 0.937 with an Arabic prompt template (AR-ZS) and 0.939 with a mixed prompt template (Mix-ZS). In comparison, Gemini-pro 1.5 scored 0.932 with the Arabic prompt template (AR-ZS) and 0.945 with the mixed prompt template (Mix-ZS), demonstrating its solid capabilities in zero-shot settings.

Mistral-Large-Latest was the next best-performing large language model (LLM), which achieved a mean Macro F1 score of 0.92. This score is only slightly lower than that of GPT-4o and Gemini-pro 1.5. Mistral-Large-latest excelled in few-shot prompt template settings across both languages, achieving performance scores of 0.924 with the Arabic six few-shot templates (AR-6FS) and 0.932 with the mixed language templates (Mix-4FS). Additionally, the model demonstrated solid performance with zero-shot template settings across both languages, with scores of 0.911 for Arabic templates (AR-ZS) and 0.917 for mixed language templates (Mix-ZS).

The third best-performing language model was the open-source bilingual Jais-30B, which achieved a mean Macro F1 score of 0.90. It performed excellently in few-shot prompting for both languages, with scores of 0.916 when using Arabic prompts (AR-6FS) and 0.930 when using mixed language prompts (Mix-6FS). However, its performance in zero-shot settings was lower, with scores of 0.867 in Arabic (AR-ZS) and 0.884 in mixed prompts (Mix-ZS). This suggests that the model requires more context or examples to understand and perform the given tasks effectively.

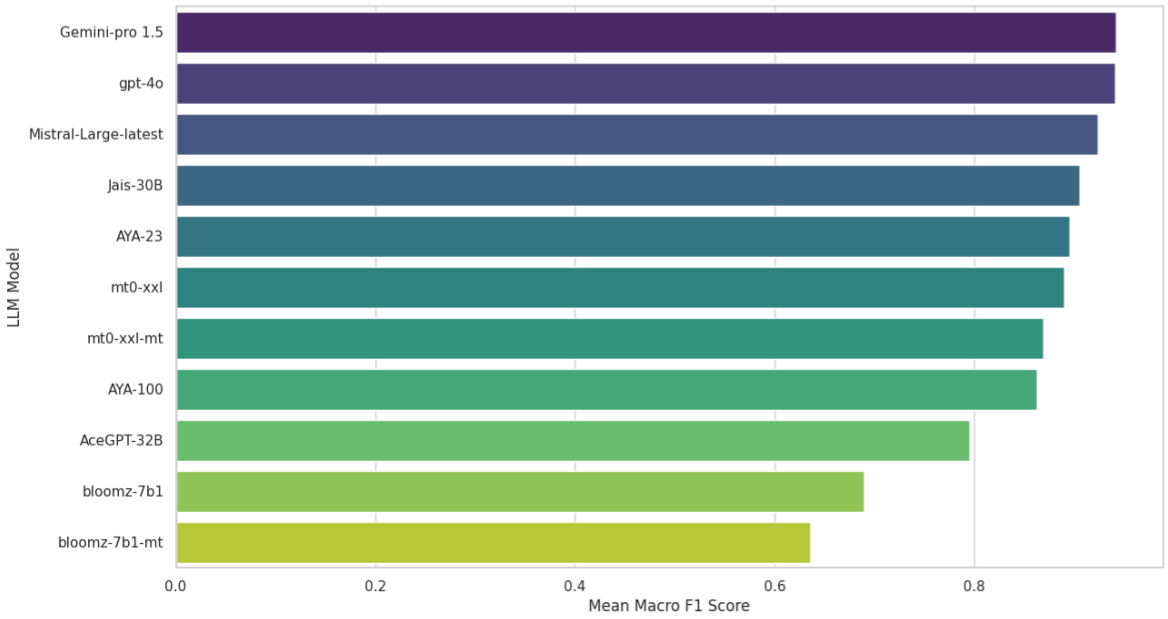


Figure 3. Performance comparison between mean (Macro F1) across LLMs.

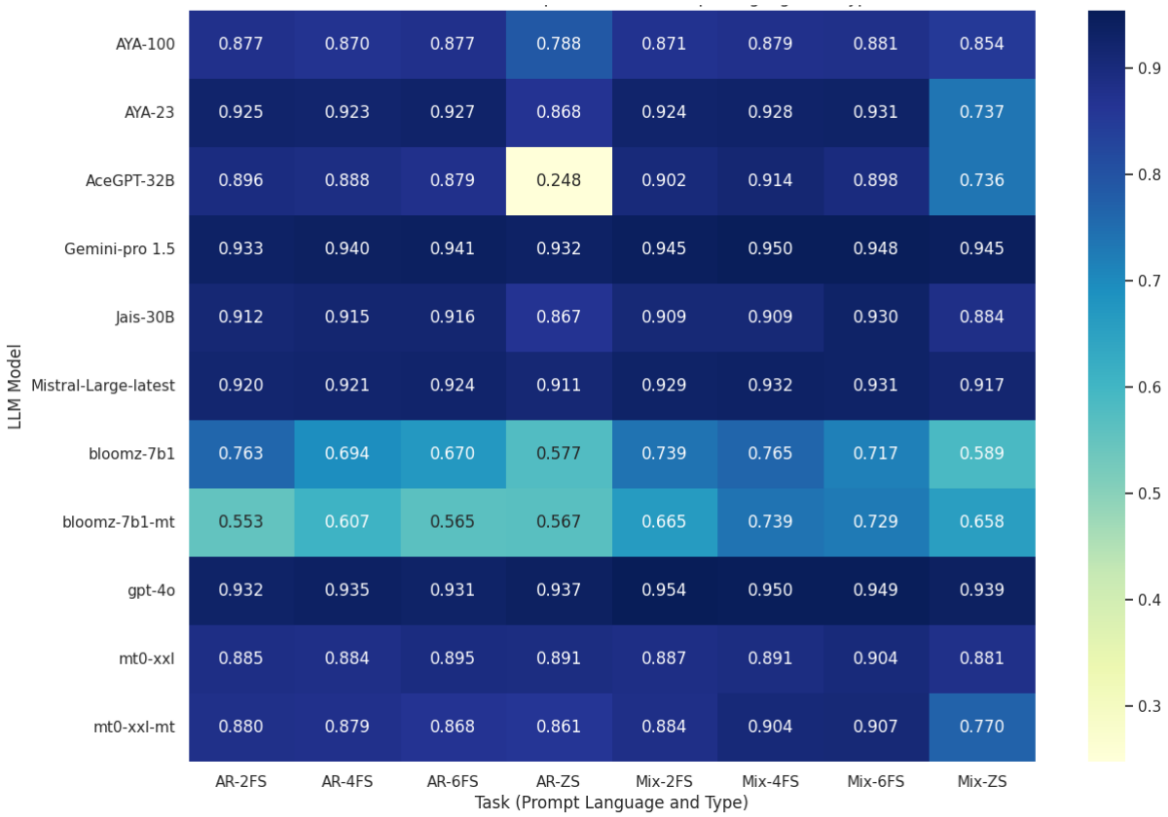


Figure 4. Performance comparison across LLMs based on prompt language and prompt type.

The models AYA-23, mT0-xxl, mT0-xxl-mt, and AYA-100 performed well across most prompt templates, achieving mean Macro F1 scores of 0.895, 0.889, 0.869, and 0.862, respectively. These models also showed their best performance in six few-shot settings; for instance, when using mixed language templates in the Mix-6FS templates setting, AYA-23, mT0-xxl, mT0-xxl-mt, and AYA-100 achieved Macro F1 scores of 0.931, 0.904, 0.907, and 0.881, respectively. However, their performance tended to decrease when using zero-shot prompts.

Table 2. Comparison of mean (Macro F1) scores between LLMs.

LLM	Mean Macro F1 Score
Gemini-pro 1.5 *	0.94 *
GPT-4o *	0.94 *
Mistral-Large-latest	0.92
Jais-30B	0.90
AYA-23	0.89
mt0-xxl	0.88
mt0-xxl-mt	0.87
AYA-100	0.86
AceGPT-32B	0.79
bloomz-7b1	0.69
bloomz-7b1-mt	0.63

*: indicates best-performing LLMs.

The AceGPT-32B model exhibited inconsistency in its performance. For example, it performed well in mixed-language few-shot settings, achieving a score of 0.914 (Mix-4FS); however, its zero-shot performance was notably poor, particularly with Arabic language prompts, where it only reached a Macro F1 score of 0.248. Analyzing the results of this language model revealed that this low score is primarily due to hallucinations: AceGPT-32B either generated unrelated text or failed to produce any results. We attempted several changes to the LLM configuration, but none of these adjustments improved the outcomes.

The worst-performing LLMs, as illustrated in Figures 3 and 4, were bloomz-7b1 and bloomz-7b1-mt. Bloomz-7b1-mt achieved the lowest mean Macro F1 score of 0.635, while bloomz-7b1 scored 0.689. Both of these open-source LLMs generally struggled, especially in zero-shot settings. In this context, bloomz-7b1-mt achieved Macro F1 scores of 0.567 when using Arabic template with Zero-shot (AR-ZS) and 0.658 in mixed language prompt template (Mix-ZS), whereas bloomz-7b1 achieved 0.577 and 0.589 when using AR-ZS and Mix-ZS prompt template setting, respectively. Even in few-shot scenarios, their performance remained below average, compared to other models. Both of these LLMs have a parameter size of 7 billion, which explains the significant difference in their performance.

Finally, we compared the best annotation results of our proposed method, which utilizes open-source LLMs as annotators, with the Q8VaxStance Dataset Labeling System [11]. The best-performing open-source LLM using our approach was Jais-30B, which achieved a Macro F1 score of 0.93 with few-shot prompts that included six examples, using English as the instruction language. In zero-shot scenarios, Jais-30B achieved scores of 0.88 and 0.86 using English and Arabic as the instruction languages, respectively. In contrast, the Q8VaxStance system reported a Macro F1 score of 0.83 when applying weak supervised learning through keyword and hashtag detection labeling functions alongside zero-shot model labeling functions. These results demonstrate that our approach outperformed the Q8VaxStance system in both zero- and few-shot contexts. We can expect even better performance when these two approaches are combined.

4.2. Open vs. Closed LLMs

The comparison between open- and closed-source LLMs, as illustrated in Table 2, revealed distinct performance differences based on their mean Macro F1 scores. Among the closed-source LLMs, GPT-4o and Gemini Pro 1.5 achieved the highest mean Macro F1 score of 0.94, and Mistral-Large-latest achieved a score of 0.92. These results highlight the strong performance of closed-source LLMs. On the open-source side, Jais-30B had the strongest performance, achieving a mean Macro F1 score of 0.90—comparable to that of the best closed LLMs—and AYA-23 also showed competitive results with a score of

0.89. Overall, the results from Figure 4 indicate that using few-shot prompts with six examples on open-source LLMs—such as Jais-30B and AYA-23—for stance detection in the Kuwaiti dialect can achieve a performance close to that of paid closed-source LLMs such as GPT-4o, Gemini Pro 1.5, and the latest version of Mistral Large. Furthermore, this suggests that open-source LLMs with fewer parameters can achieve performance similar to larger closed-source LLMs. Additionally, incorporating human feedback into the process can further enhance the annotation performance.

4.3. Statistical Significance Tests

We conducted a one-way ANOVA test to determine whether there were statistically significant differences in Macro F1 scores among the tested LLMs. The ANOVA results indicated a highly significant difference in performance across the LLMs, with a p-value of $2.81943943 \times 10^{-33}$. This value suggests that the performance, as measured by the Macro F1 score, varied significantly between at least some of the LLMs tested.

Furthermore, we performed a post hoc analysis by applying Tukey's Honest Significant Difference (HSD) test to identify which specific LLMs differed in performance. This test compared each pair of LLMs to assess whether the differences in their Macro F1 scores were statistically significant.

As illustrated in Figure 5 and Table A5, the Tukey's HSD test results revealed several statistically significant differences between the LLMs. Notably, the BigScience models (i.e., bloomz-7b1 and bloomz-7b1-mt) significantly under-performed when compared to all other LLMs. Additionally, we found that the AceGPT-32B model demonstrated significantly lower performance than all other LLMs, except for mT0-xxl-mt and AYA-100, where there was no significant statistical difference. These differences were all statistically significant, with p-values less than 0.05.

While some models showed apparent performance differences, other comparisons did not yield statistically significant results. For example, there were no significant differences between GPT-4o and models such as Gemini Pro 1.5, Jais 30B, mT0-xxl, mT0-xxl-mt, AYA-100, AYA-23, and Mistral-Large. This result suggests that these models offer comparable performance in terms of Macro F1 scores. We observed that the mT0-xxl and mT0-xxl-mt versions of the mT0 model performed equally well, showing no statistically significant difference in their performance. Similarly, there was no significant difference between AYA-100 and AYA-23, as both LLMs from Cohere For AI demonstrated comparable performance.

We also applied Tukey's HSD test to compare the results obtained with zero-shot (ZS) and few-shot (2FS, 4FS, 6FS) prompt templates. The results, as shown in Figure 6 and Table A2, indicate statistically significant differences between the results obtained using zero-shot prompts and those from 2FS, 4FS, and 6FS, which is an expected outcome. However, we found no overall statistical significance among the 2FS, 4FS, and 6FS results. Next, we applied Tukey's HSD test to compare the results of prompt templates 1, 2, and 3. Based on the results, as shown in Figure 7 and Table A3, we found no statistically significant differences between the different prompt templates. Finally, we applied Tukey's HSD test to compare the results of Arabic and Mixed (English instructions, Kuwaiti dialect examples, and tweet text) language prompt templates. As shown in Figure 8 and Table A4, the results indicated no overall statistically significant differences between the different prompt templates.

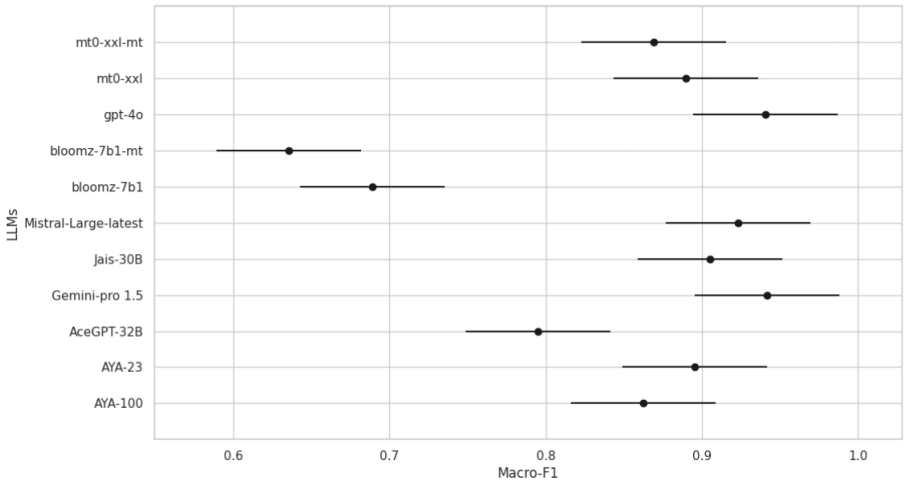


Figure 5. Tukey's HSD test results: Pairwise comparisons of models.

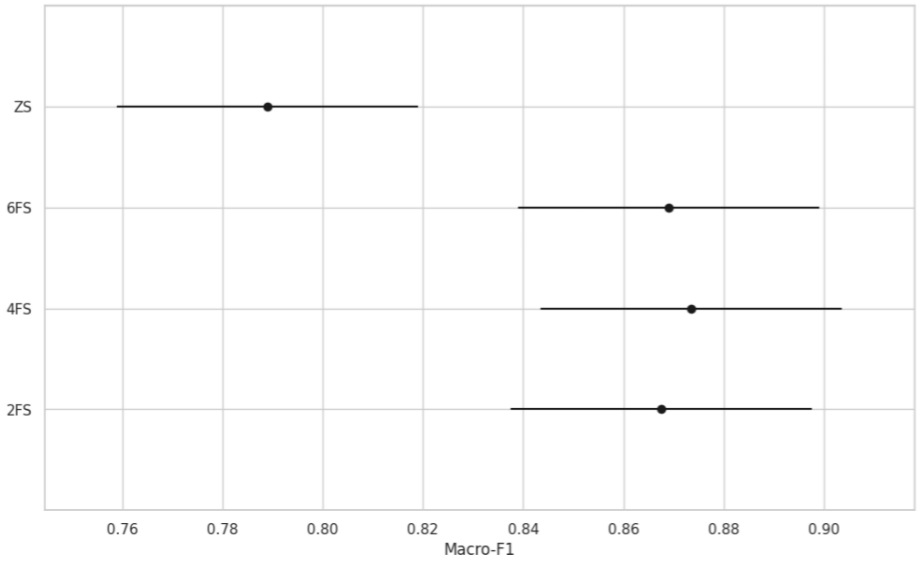


Figure 6. Tukey's HSD test results: Pairwise comparisons of zero-shot and few-shot prompt templates.

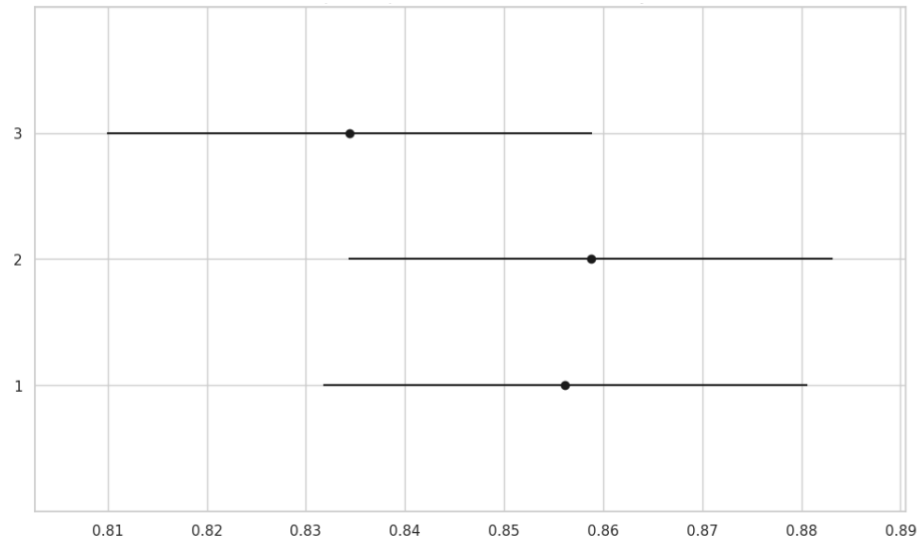


Figure 7. Tukey's HSD test results: Pairwise comparisons of prompt templates 1, 2, and 3.

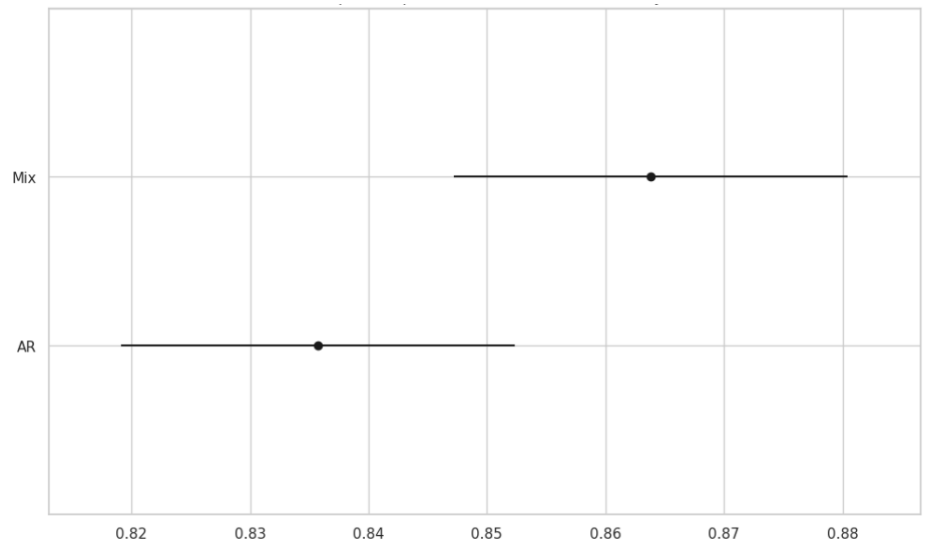


Figure 8. Tukey's HSD test results: Pairwise comparisons of Arabic and Mixed language prompt templates (English instructions, Kuwaiti dialect examples, and tweet text).

4.4. Response Consistency of LLMs

To evaluate the consistency of the responses generated by the LLMs, we analyzed the variance in their Macro F1 scores, as shown in Table 3 and Figure 9. Most models showed low variability, with variances clustered below 0.01. Based on this observation, we established 0.01 as a threshold for acceptable variance. Models with variances below this threshold were identified as consistent, while those exceeding it were considered inconsistent and exhibited higher variability; therefore, they are less reliable. Analyzing the consistency of responses based on the variance in Macro F1 scores reveals several key insights:

Models such as mT0-xxl and GPT-4o exhibited the lowest variance in Macro F1 scores (0.00014 and 0.0016, respectively), indicating a high level of consistency in performance across various prompts and settings. This suggests that these models are reliable in producing stable and predictable outputs, regardless of variations in input.

Similarly, Gemini-pro 1.5, Jais-30B, and Mistral-Large-latest also demonstrated low variance (0.00023, 0.00044, and 0.00057, respectively), reflecting strong consistency in their responses.

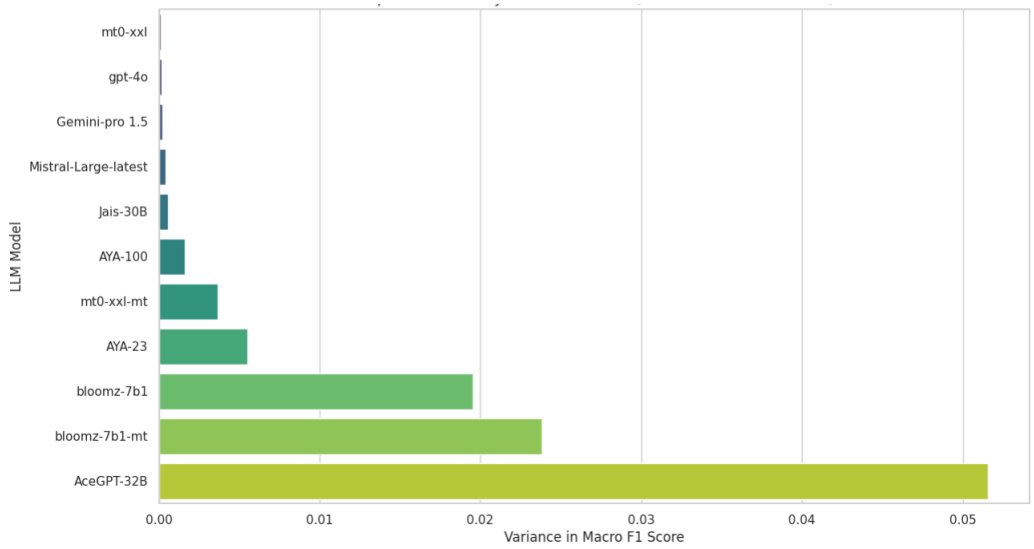


Figure 9. LLM response consistency analysis using variance in Macro F1 scores.

Table 3. Comparison of different LLMs based on consistency.

LLM	Macro F1
mT0-xxl	0.00014
GPT-4o	0.00016
Gemini-pro 1.5	0.00023
Mistral-Large-latest	0.00044
Jais-30B	0.00057
AYA-100	0.00160
mT0-xxl-mt	0.00368
AYA-23	0.00551
bloomz-7b1	0.01955
bloomz-7b1-mt	0.02384
AceGPT-32B	0.05155

In contrast, the AYA-100 and mT0-xxl-mt models showed slightly higher variance, with scores ranging from 0.00160 to 0.00368. While still relatively consistent, these models displayed some variability in their performance, indicating that specific prompt settings impact their outputs more than the top-performing models. AYA-23 exhibited a variance of 0.00551, suggesting that it is less consistent than its counterparts and may require assistance when using particular prompts or settings.

Models such as bloomz-7b1 and bloomz-7b1-mt had significantly higher variances (0.01955 and 0.02384, respectively), indicating less reliable performance. Their responses were more susceptible to changes in prompts and settings, leading to inconsistencies.

Finally, the AceGPT-32B model recorded the highest variance (0.05155), signifying its relatively low consistency among all the analyzed models. This suggests that its performance can fluctuate widely based on the prompt or setting, making it less predictable and reliable, primarily when used with English prompts.

4.5. Error Analysis

In the error analysis, we focused on the AceGPT-32B model as it demonstrated the lowest consistency among all the analyzed models. Our error analysis, based on Tables 4–6, highlights important factors that impact the accuracy and reliability of stance classification when using the AceGPT-32B model.

Table 4. Effects of prompt language, zero-shot, and few-shot prompts on AceGPT LLM performance.

Method	Macro F1 Arabic Templates	Macro F1 English Templates
ZS	**0.248**	**0.736**
2FS	0.896	0.902
4FS	0.888	0.914
6FS	0.879	0.898

Note: bold text represents the lowest Macro F1 values.

Table 5. Impact of zero-shot prompt selection on AceGPT LLM performance.

Prompt Template Language	Prompt Template Number	Macro F1 Score
AR	1	**0.354**
AR	2	**0.095**
AR	3	**0.294**
Mix	1	0.854
Mix	2	0.801
Mix	3	**0.552**

Note: bold text represents the lowest Macro F1 values.

Table 6. Error examples when using zero-shot templates with AceGPT LLM.

Prompt Template	Tweet Text	Predicted Label	Human Label	Error Type
Arabic 2	قناة atv ترفع شعار نعم للتطعيم English Translation: ATV channel raises the slogan "Yes to vaccination".	صنف المنشورين الآ English Translation: Categorize the posts now	Pro-vaccine	Hallucination
Mixed 3	في مثل هذا اليوم بدأ فيروس كورونا بالانتشار المحلي في الكويت الحل بتطعيم اكثرمن % من الشعب نعم للتطعيم والله المستعان English Translation: On this day, the Corona virus began to spread locally in Kuwait. The solution is to vaccinate more than % of the people. Yes to vaccination. May God help us.	## إعداد: - @Mohamed Al-Masry Mohamed Elmas English Translation: ## Prepared by: @Mohamed Al-Masry - Mohamed Elmas	Pro-vaccine	Hallucination
Arabic 2	المفروض مايخلطون بين المطعمين واللي مو مطعمين لازم يفصلونهم ويميزون المطعمين وبالنسبه للقرارات ما اعتبرها سذاجه اكثر حالات الوفاة انزلت خلال شهر بسبب التطعيم نعم للتطعيم نعم للتطعيم English Translation: They are not supposed to confuse those who are vaccinated with those who are not vaccinated. They must separate them and distinguish between the two vaccinated people. As for the decisions, I do not consider them naive. Most deaths occurred within a month due to vaccination. Yes to vaccination. Yes to vaccination.	في هذا الجزء، سنقوم بتدريب English Translation: In this part, we will train	Pro-vaccine	Hallucination
Arabic 1	هل يعقل ونحن في دولة تملك دستور وبرلمان وديمقراطية وعلاوة على ذلك معروف عنها إنها ببلد الإنسانية ويجبر شعبها على التطعيم وعليه لا للتطعيم الاجباري English Translation: Is it possible that we are in a country that has a constitution, parliament, and democracy? Furthermore, it is known to be a country of humanity and its people are forced to be vaccinated! So, no to compulsory vaccination.	مع التطعيم English Translation: Pro-vaccine	Anti-vaccine	Misclassification
Mixed 3	لا تحملوا المواطنين فشلكم الذريع لا للتطعيم الاجباري English Translation: Do not blame the citizens for your miserable failure, no to compulsory vaccination	Pro-vaccine	Anti-vaccine	Misclassification
Mixed 3	تم التطعيم اشكر وزارة الصحة وكل من يعمل فيها ولها من اجل صحة الكويت ، تنظيم راقي جدا English Translation: I have been vaccinated. I thank the Ministry of Health and everyone who works in it and for it for the health of Kuwait, a very sophisticated organization	Anti-vaccine	Pro-vaccine	Misclassification

4.5.1. Effect of Prompt Language and Few-Shot Examples

The choice of prompt language had a significant impact on model performance. As shown in Table 4, AceGPT-32B achieved a Macro F1 score of 0.248 for Arabic zero-shot (ZS) templates, while the English zero-shot templates performed much better, scoring 0.736. This indicates that the model has difficulty in detecting the stance when instructions are provided in Arabic. However, incorporating few-shot examples improved the accuracy for both languages. With two-shot (2FS) prompts, the Macro F1 score increased to 0.896 in Arabic and 0.902 in English. The highest performance was reached with four-shot (4FS) prompts in English, achieving a score of 0.914, which suggests that providing more examples helps the model to classify stances more accurately.

4.5.2. Impact of Prompt Selection on Performance

The design of the prompt templates significantly influenced the model's accuracy. As illustrated in Table 5, various zero-shot prompts resulted in markedly different Macro F1 scores for AceGPT-32B. Notably, Arabic Zero-Shot Prompt 2 had the lowest performance,

achieving a Macro F1 score of only 0.095. This indicates that this template was ineffective in guiding the model's responses.

4.5.3. Hallucinations and Unrelated Outputs

In AceGPT-32 B's error analysis, we identified differences between Arabic and English prompts. Out of 519 test samples (265 pro-vaccine, 254 anti-vaccine), Arabic zero-shot prompts (1, 2, and 3) had high hallucination rates, with Arabic Template 2 being the worst (457 hallucinations out of 480 total errors). This suggests the model frequently generated unrelated outputs instead of correctly identifying a stance. The following is a sample of errors from Table 6:

- Arabic Zero-shot Prompt 2: A tweet simply stating that "ATV channel supported vaccination" was processed incorrectly, and the model returned an irrelevant classification instruction ("Categorize the posts now") instead of the correct stance label.
- Mixed Zero-shot Prompt 3: A tweet about COVID-19 spreading in Kuwait and calling for more vaccinations was misinterpreted, with the model inserting an unrelated username ("## Prepared by: @Mohamed Al-Masry"), which was not part of the original tweet.
- Arabic Zero-shot Prompt 2: A pro-vaccine tweet discussing a decrease in COVID-19-related deaths due to vaccination was mistakenly transformed into a statement about training ("In this part, we will train"), indicating failure in stance detection.

These hallucinations indicate that the AceGPT-32B model fails to focus on the task and instead generates unrelated or procedural responses.

4.5.4. Misclassifications

Table 6 provides several examples where AceGPT-32B did not assign the correct stance label. Below are some sample errors:

- Arabic Zero-shot Prompt 1 Misclassification: A tweet expressing skepticism about vaccine mandates ("Is it possible that we are in a country with a constitution and democracy, yet its people are forced to be vaccinated? No to compulsory vaccination".) was incorrectly classified as pro-vaccine.
- Mixed Zero-shot Prompt 3 Misclassification: A tweet that blamed authorities for vaccination failures ("Do not blame the citizens for your miserable failure, no to compulsory vaccination".) was mistakenly labeled as pro-vaccine instead of anti-vaccine.
- Mixed Zero-shot Prompt 3 Misclassification: A tweet thanking the Ministry of Health for vaccinations ("I have been vaccinated. I thank the Ministry of Health and everyone who works in it and for it for the health of Kuwait, a very sophisticated organization") was wrongly labeled as anti-vaccine, demonstrating the model's struggle with positive statements about vaccination policies.

Misclassification rates varied significantly across zero-shot prompt templates:

- Mixed Zero-shot Prompt 3 had the highest misclassification rate (166 pro-vaccine, 29 anti-vaccine), indicating that pro-vaccine tweets were more frequently misclassified in the English prompt setting.
- Arabic Zero-shot Prompt 3 had the highest misclassification rate (51 total), with more errors in anti-vaccine tweets (30 anti-vaccine misclassified vs. 21 pro-vaccine misclassified).
- Arabic Zero-shot Prompt 1 also showed more misclassification in anti-vaccine tweets (30 anti-vaccine vs. 8 pro-vaccine).
- Arabic Zero-shot Prompt 2 had the lowest misclassification rate (23 total) but misclassified more pro-vaccine tweets (17 pro-vaccine vs. 6 anti-vaccine).

These results indicate that the language and phrasing of prompts are critical in determining the performance of LLMs.

5. Conclusions

In this study, we investigated the performance of various large language models (LLMs) in automating stance detection for the Kuwaiti dialect using both zero- and few-shot learning techniques. Our findings indicated that closed-source LLMs, such as GPT-4o, Gemini-pro 1.5, and Mistral-Large, consistently achieved high performance across different prompt settings, with mean Macro F1 scores of 94%, 94%, and 92%, respectively.

In contrast, open-source models such as Jais-30B and AYA-23 demonstrated competitive results, particularly in few-shot settings, with mean Macro F1 scores of 90% and 89%. These results position them as strong candidates for future research focused on low-resource language annotation. The study also revealed that models such as AceGPT-32B and Bloomz-7b1 struggled in zero-shot settings, especially with mixed-language prompts, highlighting the importance of context and training data in these scenarios.

The ANOVA results indicated a highly significant difference in performance among the large language models (LLMs), with a p-value of $2.81943943 \times 10^{-33}$. This value suggests that the performance—as measured by the Macro F1 score—varied significantly between at least some of the LLMs tested. Tukey's HSD test revealed several statistically significant differences among the LLMs. Notably, BigScience's models (i.e., bloomz-7b1 and bloomz-7b1-mt) significantly under-performed when compared to all other LLMs. However, we found no significant differences between GPT-4o and other models such as Gemini Pro 1.5, Jais 30B, mT0-xxl, mT0-xxl-mt, AYA-100, AYA-23, and Mistral-Large, suggesting that these models offer comparable performance in terms of Macro F1 scores.

Overall, the competitive performance of these models reassures us about the potential of open-source LLMs. Fine-tuning and human-in-the-loop involvement can lead to their further improvement, reaching performance levels similar to those of top-performing closed-source models. Utilizing open-source LLMs for Kuwaiti dialect annotation offers advantages such as cost reduction, enhanced data privacy, and the quicker annotation of large datasets, thereby enabling more NLP research projects focused on Arabic and the Kuwaiti dialect in particular.

In future work, we aim to experiment with fine-tuning the top-performing open-source models to further improve their performance, as well as exploring their application in other low-resource dialects and additional NLP tasks.

6. Limitations

This study has several limitations. First, due to the lack of available human-labeled datasets for the Kuwaiti dialect, we utilized the only online stance detection dataset currently accessible. This dataset is small and specifically focused on vaccine-related stances, which may limit its applicability to other topics or dialects. Therefore, this research serves as a foundation for testing and generating more general stance detection datasets for the Kuwaiti dialect.

Second, we only examined zero- and few-shot learning settings, excluding fine-tuning models, which the researcher intends to address in future work.

Third, the results were heavily influenced by the design and phrasing of the prompt templates; thus, using different phrasing or languages may yield different outcomes.

Furthermore, budget and computational constraints limited our ability to test a broader range of LLMs and those with larger sizes. The costs associated with closed-source models such as GPT-4 and Gemini restricted us from conducting additional experiments. Finally,

this research does not address scalability challenges relating to real-world deployment, which will be explored in future work.

Funding: This research received no external funding.

Data Availability Statement: The original data presented in the study are openly available in GitHub at https://github.com/hanaalostad/Q8Stance, accessed on 7 February 2024.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

NLP	Natural Language Processing
LLMs	Large Language Models
GPT	Generative Pre-trained Transformer
BLOOM	BigScience Language Open-science Open-access Multilingual
MSA	Modern Standard Arabic

Appendix A

Table A1. Summary of generative large language models.

LLM	Language Support	Source Type	Model Architecture	Model Size
GPT-4o	Multi-lingual	Closed	Decoder	Not officially declared
Gemini-pro 1.5	Multi-lingual	Closed	Decoder	Not officially declared
Mistral-Large-latest	Multi-lingual	Closed	Decoder	123 Billion
AYA-23	Multi-lingual	Open	Decoder	35 Billion
AceGPT-32B	Bilingual	Open	Decoder	32 Billion
Jais-30B	Bilingual	Open	Decoder	30 Billion
AYA-100	Multi-lingual	Open	Decoder	13 Billion
mt0-xxl	Multi-lingual	Open	Encoder–Decoder	13 Billion
mt0-xxl-mt	Multi-lingual	Open	Encoder–Decoder	13 Billion
bloomz-7b1	Multi-lingual	Open	Decoder	7 Billion
bloomz-7b1-mt	Multi-lingual	Open	Decoder	7 Billion

Table A2. Tukey's HSD test results: Pairwise comparisons of zero-shot and few-shot prompt templates.

Group 1	Group 2	p-adj	Reject
2FS	4FS	0.9938	False
2FS	6FS	0.9999	False
2FS	ZS	0.0047	True
4FS	6FS	0.9973	False
4FS	ZS	0.0019	True
6FS	ZS	0.0038	True

Table A3. Tukey's HSD test results: Pairwise comparisons of prompt templates 1, 2, and 3.

Group 1	Group 2	p-adj	Reject
1	2	0.9915	False
1	3	0.5464	False
2	3	0.4697	False

Table A4. Tukey's HSD test results: Pairwise comparisons of Arabic and Mixed language prompt templates.

Group 1	Group 2	p-adj	Reject
AR	Mixed	0.0968	False

Table A5. Tukey's HSD test results: Pairwise comparisons of LLMs.

Group 1	Group 2	p-adj	Reject	Group 1	Group 2	p-adj	Reject
AceGPT-32B	GPT-4o	0.0000	True	AceGPT-32B	Gemini-pro	0.0000	True
AceGPT-32B	Jais-30B	0.0065	True	AceGPT-32B	bloomz-7b1	0.0112	True
AceGPT-32B	bloomz-7b1-mt	0.0000	True	AceGPT-32B	mT0-xxl	0.0399	True
AceGPT-32B	mT0-xxl-mt	0.2534	False	AceGPT-32B	AYA-100	0.3995	False
AceGPT-32B	AYA-23	0.0213	True	AceGPT-32B	Mistral-Large	0.0005	True
GPT-4o	Gemini-pro	1.0000	False	GPT-4o	Jais-30B	0.9769	False
GPT-4o	bloomz-7b1	0.0000	True	GPT-4o	bloomz-7b1-mt	0.0000	True
GPT-4o	mT0-xxl	0.7875	False	GPT-4o	mT0-xxl-mt	0.3050	False
GPT-4o	AYA-100	0.1825	False	GPT-4o	AYA-23	0.8866	False
GPT-4o	Mistral-Large	0.9999	False	Gemini-pro	Jais-30B	0.9723	False
Gemini-pro	bloomz-7b1	0.0000	True	Gemini-pro	bloomz-7b1-mt	0.0000	True
Gemini-pro	mT0-xxl	0.7686	False	Gemini-pro	mT0-xxl-mt	0.2869	False
Gemini-pro	AYA-100	0.1697	False	Gemini-pro	AYA-23	0.8730	False
Gemini-pro	Mistral-Large	0.9999	False	Jais-30B	bloomz-7b1	0.0000	True
Jais-30B	bloomz-7b1-mt	0.0000	True	Jais-30B	mT0-xxl	1.0000	False
Jais-30B	mT0-xxl-mt	0.9737	False	Jais-30B	AYA-100	0.9149	False
Jais-30B	AYA-23	1.0000	False	Jais-30B	Mistral-Large	0.9999	False
bloomz-7b1	bloomz-7b1-mt	0.7258	False	bloomz-7b1	mT0-xxl	0.0000	True
bloomz-7b1	mT0-xxl-mt	0.0000	True	bloomz-7b1	AYA-100	0.0000	True
bloomz-7b1	AYA-23	0.0000	True	bloomz-7b1	Mistral-Large	0.0000	True
bloomz-7b1-mt	mT0-xxl	0.0000	True	bloomz-7b1-mt	mT0-xxl-mt	0.0000	True
bloomz-7b1-mt	AYA-100	0.0000	True	bloomz-7b1-mt	AYA-23	0.0000	True
bloomz-7b1-mt	Mistral-Large	0.0000	True	mT0-xxl	mT0-xxl-mt	0.9997	False
mT0-xxl	AYA-100	0.9967	False	mT0-xxl	AYA-23	1.0000	False
mT0-xxl	Mistral-Large	0.9849	False	mT0-xxl-mt	AYA-100	1.0000	False
mT0-xxl-mt	AYA-23	0.9978	False	mT0-xxl-mt	Mistral-Large	0.7204	False
AYA-100	AYA-23	0.9854	False	AYA-100	Mistral-Large	0.5494	False
AYA-23	Mistral-Large	0.9965	False				

References

1. Küçük, D.; Can, F. Stance Detection: A Survey. *ACM Comput. Surv.* **2020**, *53*, 1–37 . [CrossRef]
2. Shyu, M.L.; Yan, Y.; Chen, J. Efficient Large-Scale Stance Detection in Tweets. *Int. J. Multimed. Data Eng. Manag.* **2018**, *9*, 1–16. [CrossRef]
3. Burnham, M. Stance detection: A practical guide to classifying political beliefs in text. *Political Sci. Res. Methods* **2024**, 1–18. [CrossRef]
4. Kuo, K.H.; Wang, M.H.; Kao, H.Y.; Dai, Y.C. Advancing Stance Detection of Political Fan Pages: A Multimodal Approach. In Proceedings of the Companion Proceedings of the ACM Web Conference 2024, WWW '24, New York, NY, USA, 13–17 May 2024; pp. 702–705. [CrossRef]
5. Mets, M.; Karjus, A.; Ibrus, I.; Schich, M. Automated stance detection in complex topics and small languages: The challenging case of immigration in polarizing news media. *PLoS ONE* **2024**, *19*, e0302380. [CrossRef]
6. Lee, Y.; Ozer, M.; Corman, S.R.; Davulcu, H. Identifying Behavioral Factors Leading to Differential Polarization Effects of Adversarial Botnets. *SIGAPP Appl. Comput. Rev.* **2023**, *23*, 44–56. [CrossRef]
7. Lee, Y.; Alostad, H.; Davulcu, H. Quantifying Variations in Controversial Discussions within Kuwaiti Social Networks. *Big Data Cogn. Comput.* **2024**, *8*, 60. [CrossRef]
8. Hua, Y.; Jiang, H.; Lin, S.; Yang, J.; Plasek, J.M.; Bates, D.W.; Zhou, L. Using Twitter Data to Understand Public Perceptions of Approved Versus Off-Label Use for COVID-19-related Medications. *J. Am. Med. Inform. Assoc.* **2022**, *29*, 1668–1678. [CrossRef] [PubMed]
9. Cascini, F.; Pantović, A.; Al-Ajlouni, Y.A.; Failla, G.; Puleo, V.; Melnyk, A.; Lontano, A.; Ricciardi, W. Social Media and Attitudes Towards a COVID-19 Vaccination: A Systematic Review of the Literature. *Eclinicalmedicine* **2022**, *48*, 101454. [CrossRef]
10. List of Countries and Territories Where Arabic Is an Official Language. Available online: https://en.wikipedia.org/wiki/List_of_countries_and_territories_where_Arabic_is_an_official_language (accessed on 16 December 2024).
11. Alostad, H.; Dawiek, S.; Davulcu, H. Q8VaxStance: Dataset Labeling System for Stance Detection towards Vaccines in Kuwaiti Dialect. *Big Data Cogn. Comput.* **2023**, *7*, 151. [CrossRef]
12. Alhindi, T.; Alabdulkarim, A.; Alshehri, A.; Abdul-Mageed, M.; Nakov, P. AraStance: A Multi-Country and Multi-Domain Dataset of Arabic Stance Detection for Fact Checking. In *Proceedings of the Fourth Workshop on NLP for Internet Freedom: Censorship, Disinformation, and Propaganda*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2021; pp. 57–65. [CrossRef]
13. Alturayeif, N.S.; Luqman, H.A.; Ahmed, M.A.K. MAWQIF: A Multi-label Arabic Dataset for Target-specific Stance Detection. In *Proceedings of the Seventh Arabic Natural Language Processing Workshop (WANLP)*; Association for Computational Linguistics: Abu Dhabi, United Arab Emirates, 2022; pp. 174–184.

14. Haouari, F.; Elsayed, T. Detecting stance of authorities towards rumors in Arabic tweets: A preliminary study . In *Proceedings of the 45th European Conference on Information Retrieval*; Springer: Dublin, Ireland, 2023; pp. 430–438.
15. Hamad, O.; Hamdi, A.; Hamdi, S.; Shaban, K. StEduCov: An Explored and Benchmarked Dataset on Stance Detection in Tweets towards Online Education during COVID-19 Pandemic. *Big Data Cogn. Comput.* **2022**, *6*, 88. [CrossRef]
16. Hardalov, M.; Arora, A.; Nakov, P.; Augenstein, I. Few-Shot Cross-Lingual Stance Detection With Sentiment-Based Pre-Training. *Proc. AAAI Conf. Artif. Intell.* **2022**, *36*, 10729–10737. [CrossRef]
17. Kim, H.; Mitra, K.; Li Chen, R.; Rahman, S.; Zhang, D. MEGAnno+: A Human-LLM Collaborative Annotation System. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations, St. Julians, Malta, 17–22 March* 2024; Aletras, N., De Clercq, O., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2024; pp. 168–176.
18. He, X.; Lin, Z.; Gong, Y.; Jin, A.L.; Zhang, H.; Lin, C.; Jiao, J.; Yiu, S.M.; Duan, N.; Chen, W. AnnoLLM: Making Large Language Models to Be Better Crowdsourced Annotators. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 6: Industry Track), Mexico City, Mexico, 16–21 June 2024*; Yang, Y., Davani, A., Sil, A., Kumar, A., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2024; pp. 165–190. [CrossRef]
19. Zhang, R.; Li, Y.; Ma, Y.; Zhou, M.; Zou, L. LLMaAA: Making Large Language Models as Active Annotators. In *Proceedings of the Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, 6–10 December 2023*; Bouamor, H., Pino, J., Bali, K., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2023; pp. 13088–13103. [CrossRef]
20. Liu, R.; Lin, Z.; Tan, Y.; Wang, W. Enhancing zero-shot and few-shot stance detection with commonsense knowledge graph. In *Proceedings of the Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2021; pp. 3152–3157.
21. Liang, B.; Zhu, Q.; Li, X.; Yang, M.; Gui, L.; He, Y.; Xu, R. Jointcl: A joint contrastive learning framework for zero-shot stance detection. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Dublin, Ireland, 22–27 May* 2022; Association for Computational Linguistics: Stroudsburg, PA, USA, 2022; Volume 1, pp. 81–91.
22. Liew, X.Y.; Hameed, N.; Clos, J.; Fischer, J.E. Predicting Stance to Detect Misinformation in Few-shot Learning. In Proceedings of the First International Symposium on Trustworthy Autonomous Systems, TAS '23, New York, NY, USA, 11–12 July 2023; [CrossRef]
23. Hasanain, M.; Ahmad, F.; Alam, F. Can GPT-4 Identify Propaganda? Annotation and Detection of Propaganda Spans in News Articles. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024), Torino, Italy, 20–25 May 2024*; Calzolari, N., Kan, M.Y., Hoste, V., Lenci, A., Sakti, S., Xue, N., Eds.; ELRA and ICCL: Torino, Italy, 2024; pp. 2724–2744.
24. Abdelali, A.; Mubarak, H.; Chowdhury, S.; Hasanain, M.; Mousi, B.; Boughorbel, S.; Abdaljalil, S.; El Kheir, Y.; Izham, D.; Dalvi, F.; et al. LAraBench: Benchmarking Arabic AI with Large Language Models. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers), St. Julian's, Malta, 17–22 March 2024*; Graham, Y., Purver, M., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2024; pp. 487–520.
25. Husain, F.; Alostad, H.; Omar, H. Bridging the Kuwaiti Dialect Gap in Natural Language Processing. *IEEE Access* **2024**, *12*, 27709–27722. [CrossRef]
26. OpenAI. GPT-4o System Card. Available online: https://openai.com/index/gpt-4o-system-card/ (accessed on 22 September 2024).
27. AI Models. Premier Models. Available online: https://mistral.ai/technology/#models (accessed on 16 December 2024).
28. Team, G.; Anil, R.; Borgeaud, S.; Wu, Y.; Alayrac, J.B.; Yu, J.; Soricut, R.; Schalkwyk, J.; Dai, A.M.; Hauth, A.; et al. Gemini: A family of highly capable multimodal models. *arXiv* **2023**, arXiv:2312.11805.
29. Sengupta, N.; Sahu, S.K.; Jia, B.; Katipomu, S.; Li, H.; Koto, F.; Marshall, W.; Gosal, G.; Liu, C.; Chen, Z.; et al. Jais and Jais-chat: Arabic-Centric Foundation and Instruction-Tuned Open Generative Large Language Models. *arXiv* **2023**, arXiv:2308.16149.
30. Muennighoff, N.; Wang, T.; Sutawika, L.; Roberts, A.; Biderman, S.; Scao, T.L.; Bari, M.S.; Shen, S.; Yong, Z.X.; Schoelkopf, H.; et al. Crosslingual Generalization through Multitask Finetuning . In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Toronto, ON, Canada, 9–14 July 2023; Association for Computational Linguistics: Stroudsburg, PA, USA, 2023; pp. 15991–16111.
31. Aryabumi, V.; Dang, J.; Talupuru, D.; Dash, S.; Cairuz, D.; Lin, H.; Venkitesh, B.; Smith, M.; Campos, J.A.; Tan, Y.C.; et al. Aya 23: Open Weight Releases to Further Multilingual Progress. *arXiv* **2024**, arXiv:2405.15032.
32. Üstün, A.; Aryabumi, V.; Yong, Z.X.; Ko, W.Y.; D'souza, D.; Onilude, G.; Bhandari, N.; Singh, S.; Ooi, H.L.; Kayid, A.; et al. Aya model: An instruction finetuned open-access multilingual language model. *arXiv* **2024**, arXiv:2402.07827.
33. Huang, H.; Yu, F.; Zhu, J.; Sun, X.; Cheng, H.; Song, D.; Chen, Z.; Alharthi, A.; An, B.; He, J.; et al. AceGPT, Localizing Large Language Models in Arabic. *arXiv* **2024**, arXiv:2309.12053.

34. Singh, S.K.; Mahmood, A. The NLP Cookbook: Modern Recipes for Transformer Based Deep Learning Architectures. *IEEE Access* **2021**, *9*, 68675–68702. [CrossRef]
35. Li, K.; Liu, Z.; He, T.; Huang, H.; Peng, F.; Povey, D.; Khudanpur, S. An Empirical Study of Transformer-Based Neural Language Model Adaptation. In Proceedings of the ICASSP 2020—2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–8 May 2020; pp. 7934–7938. [CrossRef]
36. Xie, H.; Qin, Z.; Tao, X.; Letaief, K.B. Task-oriented multi-user semantic communications. *IEEE J. Sel. Areas Commun.* **2022**, *40*, 2584–2597. [CrossRef]
37. Choi, H.; Kim, J.; Joe, S.; Gwon, Y. Evaluation of BERT and ALBERT Sentence Embedding Performance on Downstream NLP Tasks. In Proceedings of the 2020 25th International Conference on Pattern Recognition (ICPR), Milan, Italy, 10–15 January 2021; pp. 5482–5487. [CrossRef]
38. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. BERT Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv* **2019**, arXiv:1810.04805.
39. Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; Sutskever, I. Language models are unsupervised multitask learners. *OpenAI Blog* **2019**, *1*, 9.
40. Beheitt, M.E.G.; Ben Haj Hmida, M. Automatic Arabic Poem Generation with GPT-2. In Proceedings of the 14th International Conference on Agents and Artificial Intelligence, Online, 3–5 February 2022; Science and Technology Publications: Setúbal, Portugal, 2022, pp. 366–374. [CrossRef]
41. Steele, J.L. To GPT or not GPT? Empowering our students to learn with AI. *Comput. Educ. Artif. Intell.* **2023**, *5*, 100160. [CrossRef]
42. Google AI. Gemini Models. Available online: https://ai.google.dev/gemini-api/docs/models/gemini (accessed on 16 December 2024).
43. Google AI. Pricing Models. Available online: https://ai.google.dev/pricing (accessed on 16 December 2024).
44. Google AI. Billing. Available online: https://ai.google.dev/gemini-api/docs/billing (accessed on 16 December 2024).
45. Core42. Core42's Bilingual AI for Arabic Speakers. Available online: https://www.core42.ai/jais.html (accessed on 16 December 2024).
46. Abdul-Mageed, M.; Elmadany, A.; Nagoudi, E.M.B. ARBERT & MARBERT: Deep Bidirectional Transformers for Arabic. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Online, 1–6 August 2021*; Zong, C., Xia, F., Li, W., Navigli, R., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2021; pp. 7088–7105. [CrossRef]
47. Kmainasi, M.; Shahroor, A.; Hasanain, M.; Laskar, S.; Hassan, N.; Alam, F. LlamaLens: Specialized Multilingual LLM for Analyzing News and Social Media Content. *arXiv* **2024**, arXiv:2410.15308.
48. Alturayeif, N.; Luqman, H.; Alyafeai, Z.; Yamani, A. StanceEval 2024: The First Arabic Stance Detection Shared Task. In Proceedings of the Second Arabic Natural Language Processing Conference, Bangkok, Thailand, 16 August 2024; pp. 774–782.
49. Badran, M.; Hamdy, M.; Torki, M.; El-Makky, N. AlexUNLP-BH at StanceEval2024: Multiple Contrastive Losses Ensemble Strategy with Multi-Task Learning For Stance Detection in Arabic. In *Proceedings of the Second Arabic Natural Language Processing Conference, Bangkok, Thailand, 16 August 2024*; Habash, N., Bouamor, H., Eskander, R., Tomeh, N., Abu Farha, I., Abdelali, A., Touileb, S., Hamed, I., Onaizan, Y., Alhafni, B., et al., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2024; pp. 823–827. [CrossRef]
50. Alghaslan, M.; Almutairy, K. MGKM at StanceEval2024 Fine-Tuning Large Language Models for Arabic Stance Detection. In *Proceedings of the Second Arabic Natural Language Processing Conference, Bangkok, Thailand, 16 August 2024*; Habash, N., Bouamor, H., Eskander, R., Tomeh, N., Abu Farha, I., Abdelali, A., Touileb, S., Hamed, I., Onaizan, Y., Alhafni, B., et al., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2024; pp. 816–822. [CrossRef]
51. Hasanaath, A.; Alansari, A. StanceCrafters at StanceEval2024: Multi-task Stance Detection using BERT Ensemble with Attention Based Aggregation. In *Proceedings of the Second Arabic Natural Language Processing Conference, Bangkok, Thailand, 16 August 2024*; Habash, N., Bouamor, H., Eskander, R., Tomeh, N., Abu Farha, I., Abdelali, A., Touileb, S., Hamed, I., Onaizan, Y., Alhafni, B., et al., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2024; pp. 811–815. [CrossRef]
52. Salamah, J.B.; Elkhlifi, A. Microblogging opinion mining approach for kuwaiti dialect. In Proceedings of the the International Conference on Computing Technology and Information Management (ICCTIM), Dubai, United Arab Emirates, 9 April 2014; Society of Digital Information and Wireless Communication (SDIWC); p. 388.
53. Quinlan, J.R. *C4.5: Programs for Machine Learning*; Morgan Kaufmann Publishers Inc.: San Francisco, CA, USA, 1993.
54. Freund, Y.; Mason, L. The Alternating Decision Tree Learning Algorithm. In Proceedings of the Sixteenth International Conference on Machine Learning, ICML '99, San Francisco, CA, USA, 27–30 June 1999; pp. 124–133.
55. Breiman, L. Random Forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]
56. Boser, B.E.; Guyon, I.M.; Vapnik, V.N. A training algorithm for optimal margin classifiers. In Proceedings of the Fifth Annual Workshop on Computational Learning Theory, Pittsburgh, PA, USA, 27–29 July 1992; pp. 144–152.

57. Antoun, W.; Baly, F.; Hajj, H. AraBERT: Transformer-based Model for Arabic Language Understanding. In *Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection, Marseille, France, 11–16 May 2020*; Al-Khalifa, H., Magdy, W., Darwish, K., Elsayed, T., Mubarak, H., Eds.; European Language Resource Association: Paris, France, 2020; pp. 9–15.
58. Wang, W.; Wei, F.; Dong, L.; Bao, H.; Yang, N.; Zhou, M. MINILM: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. In Proceedings of the 34th International Conference on Neural Information Processing Systems, NIPS '20, Vancouver, BC, Canada, 6–12 December 2020.
59. Inoue, G.; Alhafni, B.; Baimukan, N.; Bouamor, H.; Habash, N. The Interplay of Variant, Size, and Task Type in Arabic Pre-trained Language Models. In *Proceedings of the Sixth Arabic Natural Language Processing Workshop, Kyiv, Ukraine (Virtual), 19 April 2021*; Habash, N., Bouamor, H., Hajj, H., Magdy, W., Zaghouani, W., Bougares, F., Tomeh, N., Abu Farha, I., Touileb, S., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2021; pp. 92–104.
60. Tan, Z.; Li, D.; Wang, S.; Beigi, A.; Jiang, B.; Bhattacharjee, A.; Karami, M.; Li, J.; Cheng, L.; Liu, H. Large Language Models for Data Annotation and Synthesis: A Survey. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, Miami, FL, USA, 12–16 November 2024*; Al-Onaizan, Y., Bansal, M., Chen, Y.N., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2024; pp. 930–957. [CrossRef]
61. Kojima, T.; Gu, S.S.; Reid, M.; Matsuo, Y.; Iwasawa, Y. Large language models are zero-shot reasoners. In Proceedings of the 36th International Conference on Neural Information Processing Systems, NIPS '22, Red Hook, NY, USA, 28 November–9 December 2022.
62. Li, Y.; Zhang, J. Semi-supervised Meta-learning for Cross-domain Few-shot Intent Classification. In *Proceedings of the 1st Workshop on Meta Learning and Its Applications to Natural Language Processing, Online, 5 August 2021*; Lee, H.Y., Mohtarami, M., Li, S.W., Jin, D., Korpusik, M., Dong, S., Vu, N.T., Hakkani-Tur, D., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2021; pp. 67–75. [CrossRef]
63. Wei, J.; Bosma, M.; Zhao, V.Y.; Guu, K.; Yu, A.W.; Lester, B.; Du, N.; Dai, A.M.; Le, Q.V. Finetuned Language Models are Zero-Shot Learners. In Proceedings of the the Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, 25–29 April 2022.
64. Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language Models are Few-Shot Learners. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*; Curran Associates, Inc.: Red Hook, NY, USA, 2020; pp. 1877–1901.
65. Gao, T.; Fisch, A.; Chen, D. Making Pre-trained Language Models Better Few-shot Learners. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Online, 5 August 2021*; Zong, C., Xia, F., Li, W., Navigli, R., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2021; pp. 3816–3830. [CrossRef]
66. Goel, A.; Gueta, A.; Gilon, O.; Liu, C.; Erell, S.; Nguyen, L.H.; Hao, X.; Jaber, B.; Reddy, S.; Kartha, R.; et al. Llms accelerate annotation for medical information extraction. In Proceedings of the Machine Learning for Health (ML4H), PMLR, New Orleans, LA, USA, 10 December 2023; ML Research Press: Cambridge, MA, USA, 2023; Volume 225, pp. 82–100.
67. Aguda, T.D.; Siddagangappa, S.; Kochkina, E.; Kaur, S.; Wang, D.; Smiley, C. Large Language Models as Financial Data Annotators: A Study on Effectiveness and Efficiency. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024), Torino, Italy, 20–25 May 2024*; Calzolari, N., Kan, M.Y., Hoste, V., Lenci, A., Sakti, S., Xue, N., Eds.; ELRA and ICCL: Torino, Italy, 2024; pp. 10124–10145.
68. Trad, F.S.; Chehab, A. Prompt Engineering or Fine-Tuning? A Case Study on Phishing Detection with Large Language Models. *Mach. Learn. Knowl. Extr.* **2024**, *6*, 367–384. [CrossRef]
69. Kholodna, N.; Julka, S.; Khodadadi, M.; Gumus, M.N.; Granitzer, M. LLMs in the loop: Leveraging large language model annotations for active learning in low-resource languages. In Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Vilnius, Lithuania, 9–13 September 2024; Springer: Berlin/Heidelberg, Germany, 2024; pp. 397–412.
70. FreedomIntelligence. FreedomIntelligence/AceGPT-v2-32B. Available online: https://huggingface.co/FreedomIntelligence/AceGPT-v2-32B (accessed on 16 December 2024).
71. Wikipedia. Variance. Available online: https://en.wikipedia.org/wiki/Variance (accessed on 16 December 2024).
72. RunPod. RunPod—The Cloud Built for AI. Available online: https://www.runpod.io/ (accessed on 16 December 2024).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.