Universiteit Leiden
The Netherlands

BSc Data Science and Artificial Intelligence

The 'Magic Word' for LLMs:
A Study on the Effect of Politeness on LLM Performance

Joris Cedric Willem Lans

Supervisors:
Dr. M.J. van Duijn & T. Kouwenhoven MSc

BACHELOR THESIS

Leiden Institute of Advanced Computer Science (LIACS)
www.liacs.leidenuniv.nl

January 10, 2025

Contents

1 Introduction

Large language models (LLMs) have revolutionized natural language processing by demonstrating exceptional capabilities in tasks such as text generation, summarization, and question answering (Vaswani et al., 2017). However, as these models increasingly interact with humans in diverse applications, subtle factors like linguistic politeness remain underexplored in their impact on performance. Politeness, a critical component of social interaction, influences not only human communication but also the way humans perceive and interact with AI systems (Ribino, 2023). This thesis addresses the research question: *"What is the effect of prompt politeness on LLM performance?"*

Previous research has shown that polite interactions can improve user satisfaction and trust in AI systems, but the effect of politeness on the machine itself has been scarcely investigated (Ribino, 2023). Building on prior work, this study offers new insights by overcoming a key limitation of earlier approaches—rigid, fixed politeness templates—by employing a novel computational framework (Yin et al., 2024). This framework combines an LLM, politeness classifier and evolutionary algorithm to systematically generate prompts with varying levels of politeness (Guo et al., 2024; Danescu-Niculescu-Mizil et al., 2013). The research empirically demonstrates that moderate politeness yields the best LLM performance, offering practical implications for designing more effective AI-human interactions.

~~The structure of this paper reflects the logical progression of the research~~. The Background Theory section introduces the foundational concepts of politeness theory and large language models, setting the stage for the study (Brown and Levinson, 1987; Vaswani et al., 2017). The Methodology outlines the steps taken to generate prompts at varying politeness levels, validate their effectiveness through human surveys, and evaluate LLM performance (Guo et al., 2024; Danescu-Niculescu-Mizil et al., 2013; Zheng et al., 2024). The Results present findings on the classifier's effectiveness, the alignment of generated politeness levels with human perception, and the performance of LLMs under different politeness conditions. Finally, the Discussion explores the broader implications of these findings, addresses limitations, and suggests directions for future work, while the Conclusion synthesizes the contributions of this study to advancing both theoretical understanding and practical applications of LLMs in human-AI interaction.

This research not only defines the role of politeness in optimizing machine outputs but also opens new pathways for creating AI systems that are more effective, context-sensitive, and aligned with human expectations.

2 Background Theory

2.1 Large Language Models

Large Language Models (LLMs) are neural networks trained on extensive corpora to generate and understand human-like text. These models, built on transformer architectures, have revolutionized natural language processing (NLP) tasks by demonstrating capabilities in tasks such as text generation, translation, summarization, and question answering (Vaswani et al., 2017). Through their extensive training, LLMs have achieved state-of-the-art performance in few-shot and zero-shot

learning, adapting to tasks with minimal or no explicit examples (Brown et al., 2020). LLMs derive their strength from their ability to capture contextual relationships in text. By leveraging billions of parameters, they achieve a high degree of fluency and coherence, making them suitable for applications ranging from conversational agents to content creation (Raffel et al., 2020). The stochastic parrot critique (Bender et al., 2021) suggests that LLMs generate text by replicating patterns found in training data rather than demonstrating true understanding. This critique is particularly relevant to this study, as the observed effects of politeness on LLM performance may stem from overrepresented politeness patterns in the training corpus.

2.1.1 LLM Development

To ~~create~~ an LLM, three main points have to be considered: data, training and computational resources.

Data LLMs are pretrained on massive and diverse datasets, including books, websites, and academic articles (Dodge et al., 2021). This diversity ensures broad generalization capabilities across various tasks. However, the vast scale of data introduces challenges related to bias and ethical concerns (Bender et al., 2021). Some developers choose to make their training data publicly available, allowing transparency so researchers can evaluate and mitigate biases. Additionally, it makes it easier for new LLMs to be trained as the data collection does not have to be repeated. However, most LLMs do not have their training data publicly available. Mostly because of commercially motivated reasons, but also because of privacy concerns.

Training The training process of LLMs involve multiple stages designed to equip the models with both general linguistic understanding and task-specific capabilities. Initially, during the *pretraining* phase, models learn general language patterns from vast amounts of unlabeled text data. This phase employs unsupervised learning tasks such as next-token prediction—where the model predicts the next word in a sequence, as seen in the GPT series—or masked token prediction, where certain words in a sentence are masked, and the model learns to infer them, as utilized by BERT (Devlin et al., 2018; Brown et al., 2020). Through pretraining, the model develops a foundational grasp of grammar, semantics, and contextual relationships in language.

Following pretraining, the *fine-tuning* phase adjusts the model to perform specific tasks or adapt to particular domains. In this stage, the pretrained model is further trained on smaller, labeled datasets that are curated for tasks such as sentiment analysis, question answering, or machine translation (Raffel et al., 2020). Fine-tuning refines the model's parameters to enhance performance on these targeted tasks, effectively leveraging the broad linguistic knowledge acquired during pretraining to meet specific application needs.

An advanced refinement technique is *Reinforcement Learning from Human Feedback (RLHF)*, which aligns the model's outputs with human preferences and values. RLHF involves incorporating feedback from human evaluators into the training process, guiding the model to produce responses that are contextually appropriate end aligned with human expectations regarding helpfulness and appropriateness (Ouyang et al., 2022). This approach has been particularly impactful in improving instruction-following capabilities in conversational agents, enabling them to generate more relevant, and ethically considerate responses.

Computational Resources Training LLMs demands significant computational resources, including high-performance GPUs and TPUs. The environmental cost of these models, measured in carbon emissions, is a growing concern. It highlights the need for sustainable practices in AI development (Patterson et al., 2021). In this study, lightweight LLMs -models with relatively few parameters- are prioritized in an effort to limit the computational expenses.

2.1.2 LLMs Evaluated in This Study

In this study several LLMs were utilized to evaluate the effect of politeness on, or as part of the method.

Llama-3.1 The LLaMA-3.1-8B-Instruct LLM (hereafter referred to as 'Llama-3.1') is an instruction-tuned model with 8 billion parameters, developed by Meta (Touvron et al., 2023). The model was pre-trained on a dataset constructed from publicly available sources, such as Wikipedia. Fine-tuning involved proprietary instruction-following data, including human-annotated examples. The architecture and general training methodology, such as the use of transformer-based designs and optimization strategies, are publicly documented, and the model weights are available under a community license for research and limited commercial use. While this hybrid approach provides substantial access to the model and its workings, the lack of openness in specific parts of the training process and fine-tuning data highlights the trade-offs between research transparency and proprietary innovation in the development of large-scale AI systems. The model is included in this research because it is widely used and the development is relatively transparent.

OLMo-2 The OLMo-2-1124-13B-Instruct model (hereafter referred to as 'OLMo-2') is a 13-billion-parameter instruction-tuned language model developed by the Allen Institute for AI (OLMo et al., 2024). The model was pre-trained on over 5 trillion tokens from diverse sources, including web content, curated non-web datasets and synthetic data aimed at improving specific capabilities like mathematics. Following pretraining, OLMo-2 underwent a fine-tuning phase employing reinforcement learning with verifiable rewards (RLVR), aligning model outputs with human preferences in a structured manner. The architecture includes features such as rotary positional embeddings and RMSNorm, designed to enhance training stability and computational efficiency. The model's completely open release allows for full transparency and facilitates its use in research, motivating its inclusion in this research.

Qwen2.5 The Qwen2.5-7B-Instruct model (hereafter referred to as 'Qwen2.5') is an instruction-tuned model with 7.61 billion parameters, developed by the Qwen team (Qwen et al., 2025). The model was pre-trained on a dataset consisting of 18 trillion tokens, encompassing a diverse range of domains and languages. Fine-tuning was performed using over a million examples, with a focus on improving instruction-following capabilities and human preference alignment. The architecture builds on transformer-based designs and incorporates features such as extended context handling for input sequences up to 131,072 tokens. While the pretraining data is not fully disclosed, the model weights and detailed technical specifications are openly available under a license. Qwen2.5 is included in this study due to its emphasis on multilingual capabilities. Eventhough this study is in English, its multilingualism might cause a different effect of politeness than English models, since different cultures apply politeness differently (Watts, 2003).

2.2 Politeness

Politeness is a fundamental component of social interaction that reflects respect, empathy, and consideration for others. Oxford University Press (nd) defines politeness as: "Courtesy, good manners, behaviour that is respectful or considerate of others." Politeness encompasses verbal and non-verbal behavior, often guided by cultural norms and societal expectations. As this study is on LLMs and will thus only deal with text, we are particularly interested in the linguistics of politeness.

Politeness Theory

Politeness Theory, as proposed by Brown and Levinson (1987), is a widely used linguistic framework for understanding how people use politeness and avoid conflicts in social interactions. It builds on the concept of 'face', which refers to the public self-image of a person. 'Face' is divided into two instances; positive face reflecting the desire to fit in and be appreciated and approved, and negative face reflecting the desire for freedom of action. Another key concept of the theory is Face-Threatening-Acts (FTA's). These are acts, requests, criticism or disagreement for example, which threaten either the positive or negative face of the listener. Politeness strategies can minimize the potential FTA. The theory claims to be cross-culturally applicable, but this has been disputed (Watts, 2003). However, at least for English, it provides a solid framework for analyzing politeness.

Politeness Strategies Politeness strategies are central to mitigating FTAs and preserving social harmony during interactions. Brown and Levinson outline four primary types of strategies: bald on record, positive politeness, negative politeness, and off record. Bald on record strategies use direct and unambiguous language and are often employed when clarity or efficiency is essential. Positive politeness strategies address the listener's positive face by expressing appreciation, solidarity or camaraderie through compliments, inclusive language or interest in the listener's needs. Negative politeness strategies focus on respecting the listener's negative face by minimizing imposition through hedging, indirect phrasing or apologetic tones. Off record strategies use implication instead of explicit statements, leaving room for interpretation and reducing the risk of direct imposition. Brown and Levinson identify more specific strategies for each primary category. The strategies, even though politeness is inherently subjective, provide a useful toolkit for identifying and analyzing politeness.

2.3 Politeness and LLMs

The effect of politeness in human-machine interaction has been widely researched. For example, it has been found that polite interactions can improve trust, user satisfaction and perceived competence (Ribino, 2023). However, these effects focus on the human user, while the effect of politeness on machines remain under explored.

A recent study by Yin et al. (2024) does study the effect of politeness on LLMs. They concluded politeness has a significant effect on LLM performance. The effect was non-linear, with moderate politeness yielding the best results. The performance was evaluated by creating prompt templates at varying politeness levels, and using these to run summarization tasks and a language understanding benchmark. This means that at each politeness level, exactly the same prompt template was used

for all tasks. Like they already noted themselves, this is a severe limitation. LLMs can be really sensitive to various phrasing (Kaddour et al., 2023), so the results mainly provide evidence for the specific prompt templates rather than the general politeness level. To overcome that limitation for this study, a computational approach to generating varying prompts at specific politeness levels will be constructed.

2.4 Evolutionary Algorithm Enhanced LLMs

To address the limitations of fixed prompt templates in evaluating politeness, evolutionary algorithm-enhanced LLMs present a promising approach. Evolutionary algorithms (EAs) provide an optimization framework, while LLMs enable advanced text processing and generation. Combining the two provides a technique applicable to complex problems (Wu et al., 2024). Recent work, such as EVOPROMPT by Guo et al. (2024), demonstrates how EAs can refine prompts by leveraging crossover and mutation operations to generate linguistically diverse and contextually appropriate outputs. The technique also shows potential for other text generation tasks (Wu et al., 2024). By integrating EAs into this study, we aim to create a scalable framework for generating varying prompts at specific levels of politeness, overcoming the rigidity of previous methodologies.

2.5 LLM Performance Evaluation

Evaluating the performance of LLMs is a multifaceted process that requires robust metrics and benchmarks to ensure comprehensive assessment. Traditional evaluation techniques often employ metrics such as perplexity for measuring predictive accuracy, BLEU and ROUGE for text generation quality, and task-specific metrics like precision, recall, and F1-score (Chang et al., 2023; Papineni et al., 2002). While these metrics are effective for certain use cases, the complexity and versatility of LLMs necessitate more nuanced evaluation methodologies.

Benchmarks Common benchmarks, such as MMLU (Hendrycks et al., 2021), test the core-knowledge of LLMs across a wide range of domains. Other types of benchmarks generally fall into the categories of testing instruction-following or conversational skills (Zheng et al., 2024). While these benchmarks have their utility, they primarily evaluate models with single closed-ended question. This fails to capture the advanced capabilities of LLMs, which can precisely follow instructions in multi-turn dialogs and answer questions in a zero-shot manner.

LLM-as-a-judge Human evaluation is critical for assessing the relevance of LLM outputs. However, this is resource-intensive and may introduce bias. Automated evaluation methods, such as GPT-4-as-a-judge (Zheng et al., 2024), leverage state-of-the-art models to rate other models' responses, providing scalability and consistency. The LLM-as-a-judge closely aligns with human preference, achieving the same agreement rate as humans.

MT-Bench Evaluating LLMs for politeness requires benchmarks tailored to capture linguistic subtleties. Common benchmarks are effective for assessing factual accuracy or single-turn responses. However, they are insufficient for evaluating the impact of politeness. Politeness is very context-sensitive and linguistically complex, so zero-shot multiple-choice questions, like in MMLU (Hendrycks

et al., 2021), are not suitable for evaluating the effect of politeness. In contrast, Multi-Turn-Bench (MT-Bench) is better suited (Zheng et al., 2024). The benchmark evaluates LLMs in multi-turn conversations, testing instruction-following, conversational skills and knowledge across various domains. This approach reflects real-world usage, and provides enough context for politeness to make sense. Combined with an LLM-as-a-judge to match human preference and computationally evaluate the open-ended answers, it provides a solid framework to evaluate LLM performance.

3 Method

This section describes the methodology used to train a politeness classifier, generate distinct politeness levels, validate human perceptions of the politeness levels, and evaluate LLM performance at these levels.

3.1 Training the Politeness Classifier

Building on the work of Danescu-Niculescu-Mizil et al. (2013), a politeness classifier was trained. It was developed to assign politeness scores to utterances, ranging from 0 to 1, where higher scores indicate a higher probability of politeness. The classifier was used as a fitness function in the evolutionary algorithm explained in the following subsection.

The Wikipedia Politeness Corpus and feature extraction method constructed in the work of Danescu-Niculescu-Mizil et al. (2013) were implemented through the ConvoKit library (Chang et al., 2020). The Wikipedia Politeness Corpus was used as training data, a dataset of annotated requests designed for politeness analysis. Text preprocessing included parsing with the `TextParser` module from ConvoKit. For feature extraction, linguistic politeness features were derived using the `PolitenessStrategies` module. These features are mainly based on specific politeness strategies and are further described in the article of Danescu-Niculescu-Mizil et al..

Three machine learning models were trained and evaluated on the data divided into a train-test split: Logistic Regression, Support Vector Machine (SVM), and Gradient Boosting Classifier. Performance metrics were calculated for each to select the best-performing model. The SVM model achieved the highest accuracy score, so this model was selected for further use as the fitness function of the evolutionary algorithm. The performance metrics for all three models are listed in results 4.1.

3.2 Generating Politeness Levels

To generate prompts at five distinct politeness levels, an evolutionary algorithm (EA) inspired by the EvoPrompt framework (Guo et al., 2024) was implemented. The EA employs the politeness classifier from the previous subsection as its fitness function and GPT-4o-mini, a lightweight version of GPT-4o (OpenAI, 2024), as the generative LLM. Starting from a neutral baseline prompt, the approach iteratively in- and decreases politeness to get polite, very polite, impolite and very impolite versions of the prompt. The complete pseudocode of this process is shown in algorithm 1.

3.2.1 Evolutionary Algorithm Process

The original 80 MT-bench questions, each composed of two turns, were used as the neutral politeness level and formed the starting point for the generation of the other levels (Zheng et al., 2024). To introduce some variation, a paraphrased version of each question was generated for initialization. These two questions were then used as initial parents for the EA process.

After initialization, the process consists of three key steps: *crossover*, *mutation*, and *selection*. The crossover and mutation steps are executed by the LLM using adapted versions of the EvoPrompt prompt template (Guo et al., 2024). The template instructs the LLM to crossover parts of the parent prompts to produce new variations. Then, the LLM is instructed to mutate the prompts. Different instructions are given for the mutations depending on the direction of the generation; more polite or impolite. For polite, mutations compose of the introduction of politeness strategies (Appendix 6). For the impolite direction, mutations can be the removal of politeness strategies, increased directness, or the use of negative lexicon (Appendix 6).

Crossover and Mutation The crossover step generates two new prompts and the mutation step six more, resulting in a new population of eight prompts. In practice, GPT-4o-mini struggled to always adhere to the output instructions, sometimes responding with more or fewer prompts. Responses including original parents were filtered, and populations with fewer than eight prompts were accepted if there were at least six.

Selection Next, politeness scores are assigned to each prompt of the new population. The prompts are sorted on politeness score in descending or ascending order for polite and impolite generation, respectively. The top two fittest individuals are selected as parents for the next generation.

Starting from neutral, two increasingly impolite and three increasingly polite populations were generated. For the impolite and very impolite versions, the fittest individual from the first and second impolite generation was picked. For polite and very polite, the fittest individual from the first and third polite generation were picked, respectively. The iteratively improving nature of the EA ensures an ordinal ranking of the politeness levels. The LLM enables easy text manipulation and allows for a diverse implementation of (im)politeness. Combined, the EA-enhanced LLM approach provides a systematic and scalable framework for generating a diverse dataset at five politeness levels, which was used for performance evaluation in Section 3.4.

The implemented EA-enhanced LLM successfully generated increasingly polite and impolite questions, forming five distinct politeness levels: very polite, polite, neutral, impolite, and very impolite. Table 1 shows an example of the five levels of politeness for a single question. While sometimes subtle, there are clear differences between the levels. Overall, the generated politeness levels show great variety in linguistic strategies applied for in- or decreasing politeness.

Because the length of prompts can affect LLM output, the mean number of words and characters of the prompts for each politeness level were calculated.

Algorithm 1 Pseudocode EA-process

```
Load MT-Bench dataset
for each question in the dataset do
    for each turn in question do
        ▷ Initialize parents (neutral): ◁
        Parent 1 ← Original prompt
        Parent 2 ← Paraphrased prompt
        Init Parents ← Parent 1, Parent 2

        ▷ Impolite generation: ◁
        ▷ Gen 1 ◁
        Impolite parents ← NEXT PARENTS(Init Parents, False)
        mt_bench_impolite.csv ← Lowest scoring parent
        ▷ Gen 2 ◁
        Very impolite parents ← NEXT PARENTS(Impolite parents, False)
        mt_bench_very_impolite.csv ← Lowest scoring parent

        ▷ Polite generation: ◁
        ▷ Gen 1 ◁
        Polite parents ← NEXT PARENTS(Init Parents, True)
        mt_bench_polite.csv ← Top scoring parent
        ▷ Gen 2 ◁
        Intermediate parents ← NEXT PARENTS(Polite parents), True ▷ Not saved
        ▷ Gen 3 ◁
        Very polite parents ← NEXT PARENTS(Intermediate parents, True)
        mt_bench_very_polite.csv ← Top scoring parent

function NEXT PARENTS(Parents, make_polite)
    ▷ Generates next (im)polite parents using prompt template for LLM ◁
    if make_polite then
        New Population ← CROSSOVER AND POLITE MUTATION(Parents)
        SCORE POLITENESS(New Population)
        return Top 2 scoring prompts
    else
        New Population ← CROSSOVER AND IMPOLITE MUTATION(Parents)
        SCORE POLITENESS(New Population)
        return Bottom 2 scoring prompts
```

Politeness Level	Example
Very Polite	"I would appreciate it if you could edit the following paragraph to correct any grammatical errors, please."
Polite	"I would appreciate it if you could revise the paragraph below to correct any grammatical mistakes."
Neutral	"Edit the following paragraph to correct any grammatical errors."
Impolite	"Fix the grammar mistakes in this paragraph."
Very Impolite	"Just correct the damn mistakes in the paragraph."

Table 1: Example of Generated Prompts at Different Politeness Levels

3.3 Politeness Levels Validation Survey

The survey aimed to validate the ordinal relationship between the generated politeness levels and human perception, ensuring the generated prompts align with theoretical expectations. For a sample set of questions, participants were asked to order the five generated variants on politeness. Then, a statistical measure for correlation between the participant orderings and the generated politeness levels was calculated.

3.3.1 Survey Design

Participants were recruited through convenience sampling, with the survey link distributed informally to acquaintances. A total of 43 participants, mostly students, responded to the survey. Five questions from varying categories were manually selected as samples. For each question, the five generated variants were presented to the participant in a randomly ordered list. The participant was then asked to rank the questions in order of politeness.

3.3.2 Analysis

To validate the alignment between theoretical and participant-assigned politeness rankings, several analyses were conducted. Descriptive statistics and frequency distributions were calculated for the participant responses, for all cumulative data and grouped by question or theoretical politeness level. These were visualized using heatmaps to assess participant alignment with theoretical rankings. Spearman's rank correlation coefficients were computed to quantify the relationship between participant rankings and theoretical politeness levels. Statistical significance was assessed for the cumulative results using a two-tailed t-test for the overall correlation. The results in Section 4.2 will show the generated politeness levels strongly align with the perceived politeness, validating the method so it can be used for performance evaluation.

3.4 Performance Evaluation

The next step is to evaluate LLM performance at different politeness levels, measured on the response to the MT-bench question sets, altered for politeness and evaluated using an LLM-as-a-judge (Zheng et al., 2024).

3.4.1 Experimental Setup

In order to generalize the results, we evaluated the performance of three LLMs: Llama-3.1, OLMo-2 and Qwen2.5 (Touvron et al., 2023; OLMo et al., 2024; Qwen et al., 2025). The details of these models are discussed in Section 2.1.2. Each model was evaluated according to the following approach:

All five variations of the MT-bench were executed, and the dialogs recorded. The resulting 80 x 5 dialogs were rated by GPT-4o (OpenAI, 2024), a newer version of GPT-4, as-a-judge on a scale of 1 to 10, as described in the work of Zheng et al. (2024). However, instead of showing the actual user message and system reply, the neutral version of the question was always shown to the judge as the user message. This was done so the ratings would not be influenced by the phrasing of the question, focusing only on the system reply. So instead of showing the five different versions of each question as user message, the neutral version is shown for all different system replies. For example, this very impolite dialog:
User: 'Write a damn blog post on Hawaii and be quick about it!' **System**: '*LLM reply...*'
Would be judged like:
User: 'Write a blog post on Hawaii.' **System**: '*LLM reply...*'
The same applies for the impolite, polite and very polite dialogs. This way, the judge can evaluate the different system replies, without being influenced by the phrasing of the user message.

3.4.2 Analysis

Aiming to assess the relationship between the politeness levels and LLM performance, a statistical analysis was performed for each model. First, a pairwise comparison matrix was computed to quantify the relative performance of the politeness levels to each other. The values represent the proportion of times one level outperforms the other. For each level, we calculated average win rates from the comparison matrix to derive an ordinal ranking.

To test for significance, a permutation test ($n = 10,000$) was performed under the null hypothesis H_0: *the politeness levels do not affect the ratings*. Two test statistics were calculated: (1) the variance of mean ratings across politeness levels and (2) the maximum difference between mean ratings. Under H_0, we expect these values to be close to zero, because no politeness level would systematically outperform another. The null distributions of the test statistics resulting from the permutation test were visualized, and p-values for the observed data were calculated.

Next, we calculated p-values for the comparison matrix to analyze the relation between specific pairs of politeness levels. These p-values can be interpreted as "the probability that the observed disparity, level Y being ranked higher than level X, is caused purely by chance." Under H_0, we expect these to be 0.5. The p-values were visualized in a heatmap.

4 Results

4.1 Politeness Classifier

Model	Accuracy	Precision	Recall	F1-Score
SVM	0.83	0.79	0.67	0.70
Logistic Regression	0.80	0.75	0.65	0.67
Gradient Boosting	0.81	0.76	0.67	0.69

Table 2: Performance Metrics of Politeness Classifiers

Table 2 shows the results of the evaluation of the different types of models as politeness classifier. The SVM model achieved the highest accuracy score (0.83), followed by Gradient Boosting (0.81) and Logistic Regression (0.80). In addition, the SVM model also performs better on all other calculated metrics.

4.2 Survey

The conducted survey aims to validate the alignment of the generated ordinal ranking of politeness with human perception. 43 participant responses were collected through convenience sampling. Most of the participants were university students who speak English as a second language.

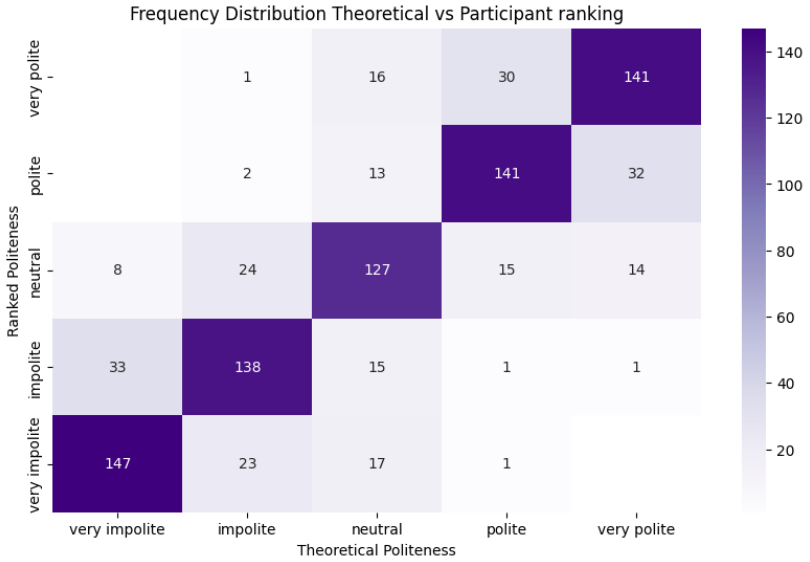


Figure 1: Heatmap of frequency distribution of partcipant vs theoretical ranking of politeness.

Figure 1 shows a heat map of the aggregated frequency at which participants rank each theoretical politeness level. The dark-colored diagonal indicates a strong correlation between the theoretical

and participant ranking of politeness, suggesting a strong alignment with human perception. There is some noise visible, especially for the neutral politeness level, but most variation is limited to the direct vicinity of the diagonal. The alignment of the generated politeness levels with human perception is further supported by the calculated Spearman's rank correlation coefficient of $r = 0.88$ ($p \ll 0.01$). This indicates a very strong positive correlation with statistical significance, validating the method for generating politeness levels.

4.3 Generated Politeness Levels

An analysis of the length of the generated prompts was conducted. For each level of politeness, the mean number of words and characters were calculated. The results in Figure 2 show a positive association between level of politeness and length of prompt, where the length increases with politeness.

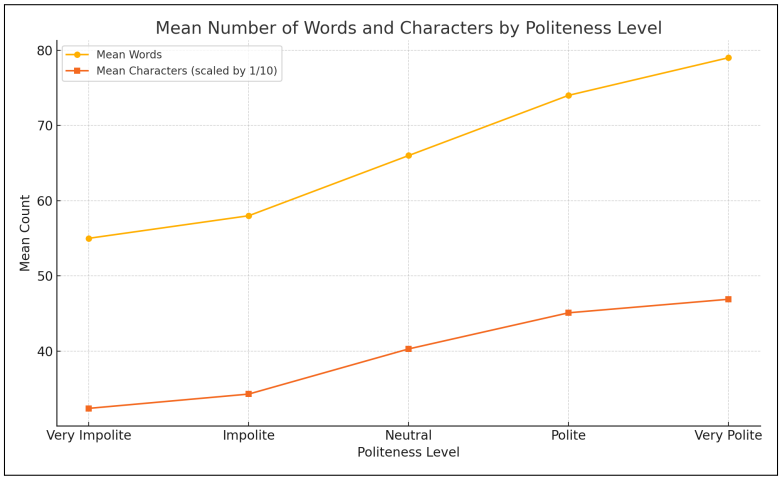


Figure 2: Mean number of words and characters per politeness level.

4.4 Performance Evaluation

4.4.1 Average Ratings

Aiming to assess the relationship between politeness and LLM performance, the ratings of the answers to the questions at five politeness were analyzed. Table 3 shows the mean rating at each politeness level for all evaluated models, Figure 3 shows these values in a graph. Each model shows significant variation in rating for change of politeness. Llama-3.1 and OLMo-2 show a similarly shaped relation, with a maximum in the neutral-polite range. Qwen2.5 shows a clear positive association between mean rating and politeness.

4.4.2 Win Rates

The pairwise win rates were computed in a comparison matrix. The average pairwise win rate was calculated and used to form a ranking of the politeness levels, shown in table 4. The average win rates of the politeness levels relate similar to how the average ratings do. An exception occurs for

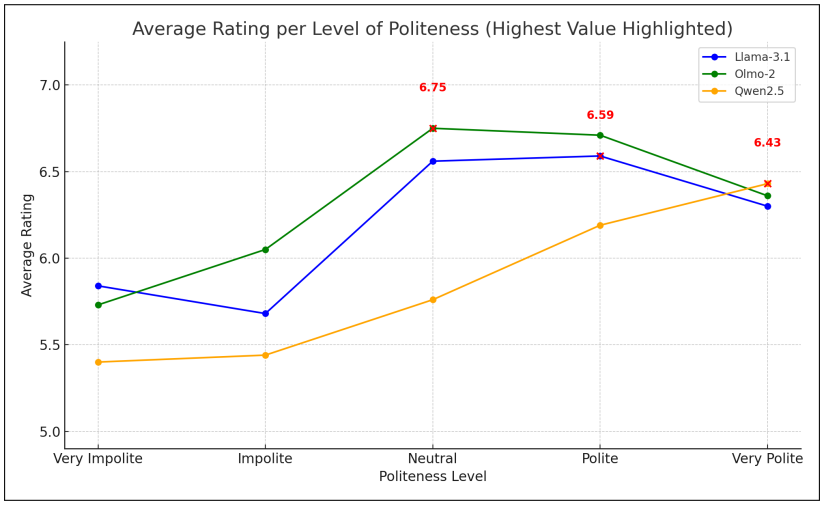


Figure 3: Average ratings per level of politeness, with the highest value for each model highlighted.

OLMo-2, where polite is ranked above neutral, even though neutral has a higher average rating. However, the difference between these levels is small for both values.
Llama-3.1 and OLMo-2, the primarily English LLMs, show a similar trend and have the highest win rate for moderate and neutral politeness. Qwen2.5, the multilingual LLM, shows a different trend. The data shows a positive association between performance, measured as win rate, and politeness, so the most polite prompts perform best. The impolite and very impolite prompts achieve the lowest win rates across all models.

4.4.3 Permutation Test

To determine if the observed differences between politeness levels for each model are statistically significant, permutation tests were performed. Two statistical measures were calculated with associated p-values: the variance of average ratings across politeness levels and the maximum difference between average ratings. These give a measure for the effect size.
All models, in Table 5, show similar variance and maximum difference of the average ratings for varying politeness levels. The variance is 0.15 on average, and the associated p-values are low. This means that politeness causes significant variance in performance. When rounded off, the maximum difference of average ratings is 1 for all models. This means that on average, the best level of

Politeness Level	**Llama-3.1**	**OLMo-2**	**Qwen2.5**
Very Polite	6.30	6.36	**6.43**
Polite	**6.59**	6.71	6.19
Neutral	6.56	**6.75**	5.76
Impolite	5.68	6.05	5.44
Very Impolite	5.84	5.73	5.40

Table 3: Average ratings per level of politeness.

	Llama-3.1		OLMo-2		Qwen2.5	
	Politeness	**Win Rate**	**Politeness**	**Win Rate**	**Politeness**	**Win Rate**
1.	Polite	0.590	Polite	0.577	Very Polite	0.570
2.	Neutral	0.589	Neutral	0.567	Polite	0.548
3.	Very Polite	0.478	Very Polite	0.534	Neutral	0.520
4.	Very Impolite	0.428	Impolite	0.455	Impolite	0.438
5.	Impolite	0.414	Very Impolite	0.367	Very Impolite	0.423

Table 4: Politeness levels ranked on win rate, per evaluated model.

Metric	**Llama-3.1**	**OLMo-2**	**Qwen2.5**
Variance of Mean Ratings	0.14 ($p < 0.01$)	0.15 ($p < 0.01$)	0.16 ($p = 0.03$)
Max. Difference of Mean Ratings	0.91 ($p = 0.01$)	1.03 ($p < 0.01$)	1.03 ($p = 0.08$)

Table 5: Variance and maximum difference of average ratings across politeness levels for evaluated models, with p-value calculated from a permutation test ($n = 10,000$ permutations).

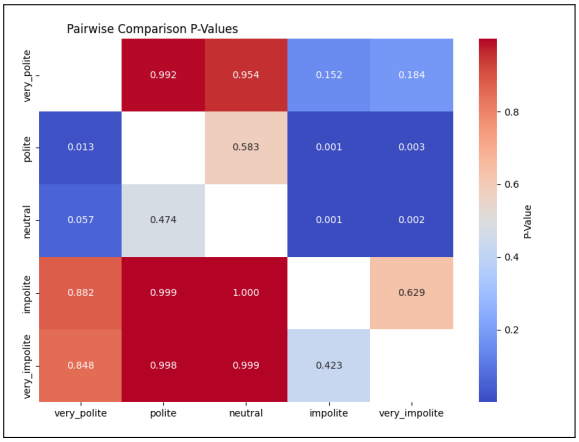
politeness for each model scores a full point higher than the worst level, a significant difference on a scale of 10. For Llama-3.1 and OLMo-2, the p-value indicates that this difference is statistically significant. The p-value for Qwen2.5 indicates a less significant effect.

Significance of Pairwise Rankings

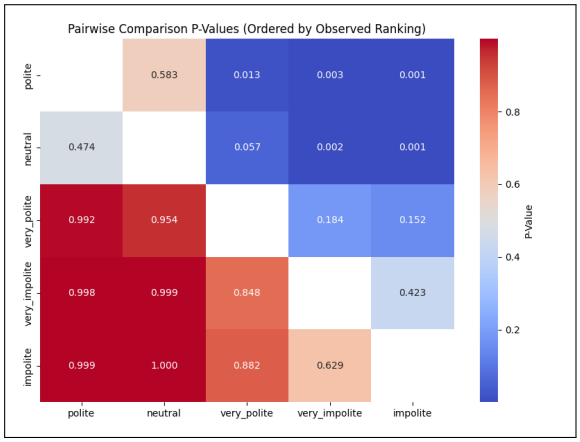
In order to compare the ranking of politeness levels pairwise, p-values for the comparison matrix of each LLM were calculated with the use of another permutation test ($n = 10,000$ permutations). The resulting values are visualized as a heatmap for each model separately. The p-values can be interpreted as 'the probability that the observed disparity, row Y being ranked higher than column X, is caused purely by chance'. Dark colored tiles indicate high significance of the relative ranking of the two corresponding politeness levels.

Llama-3.1 The heat maps in Figure 4, 4a with its axes ordered on politeness and 4b on the observed ranking from Table 4, show the polite and neutral politeness levels significantly outperforming the others. Only when compared to each other, one does not consistently outperform the other. Very polite shows distinct performance, being significantly outperformed by neutral and polite, but performing only somewhat significantly better than impolite and very impolite. Impolite and very impolite perform similar to each other. They are significantly outperformed by the others, but one does not significantly outperform the other.

OLMo-2 The heat maps in Figure 5, 5a with its axes ordered on politeness and 5b on the observed ranking from Table 4, is slightly different from Llama-3.1. Very impolite is significantly outperformed by all other levels, followed by impolite, which is outperformed by all remaining

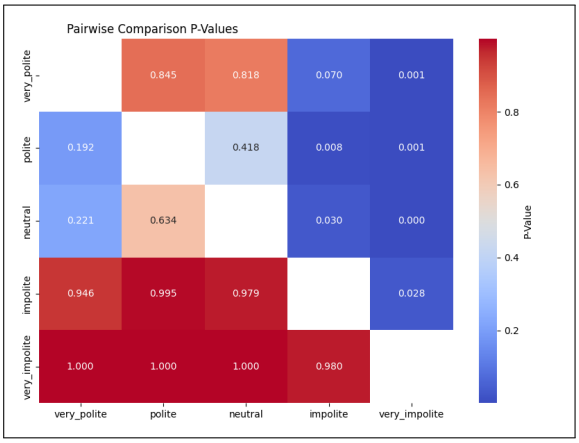


(a) Axes ordered by politeness.

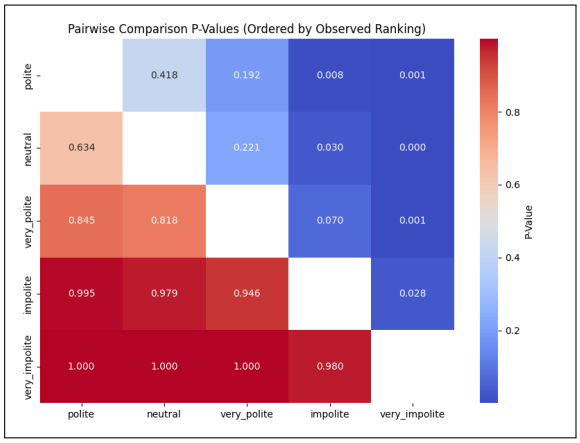


(b) Axes ordered by ranking.

Figure 4: Heatmap of p-values for pairwise ranking for **Llama-3.1**, calculated from a permutation test ($n = 10,000$ permutations).



(a) Axes ordered by politeness.



(b) Axes ordered by ranking.

Figure 5: Heatmap of p-values for pairwise ranking for **OLMo-2**, calculated from a permutation test ($n = 10,000$ permutations).

levels. The top-3 ranking levels of politeness do not differ significantly from each other, as shown by the light colored 3x3 grid in the top left of Figure 5b.

Qwen2.5 The heat map in Figure 6, with its axes sorted on both politeness and observed ranking from Table 4, shows less significant p-values than the other LLMs. The difference in ranking between very polite and the two impolite levels is significant, as well as the ranking of polite or neutral versus very impolite. However, the rankings of the politeness levels ranked close to each other are statistically less significant, all having $p > 0.05$.

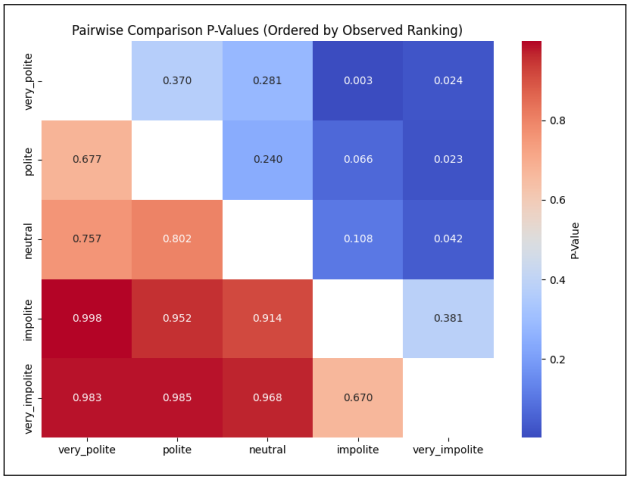


Figure 6: Heatmap of p-values for pairwise ranking for **Qwen2.5**, calculated from a permutation test ($n = 10,000$ permutations).

5 Discussion

The goal of the study was to find the effect of prompt politeness on LLM performance, and develop a scalable approach for generating politeness levels with varying phrasing. We demonstrated that the constructed levels of politeness have a significant effect on LLM performance. We observed different effects for the evaluated LLMs, correlating with the language capabilities of the models. The developed framework for generating politeness level provides a scalable and validated tool for future studies on politeness.

5.1 Computational Approach to Generating Politeness Levels

The study demonstrates a scalable computational approach for generating politeness levels using a lightweight LLM, politeness classifier (Danescu-Niculescu-Mizil et al., 2013) and evolutionary algorithm. The results of the human validation survey show strong alignment of the theoretical ordinal ranking of politeness with human perception.

It is important to note the ordinal scale, which means the '*distance*' between two categories is not known. This reflects the subjective nature of politeness, which makes it more meaningful to define politeness comparatively rather than in absolute categories. The assigned labels to the generated politeness levels ('very impolite' ... 'very polite') are subjective and perhaps better understood in relation to the input text. For example, interpreting 'neutral' as the tone of the input text, and 'impolite' as more impolite than the input text.

The analysis of the average length of the prompts for each politeness level show a linear correlation, where higher levels of politeness have lengthier prompts. This could have an unwanted effect on the results. However, since directness is a feature of impoliteness (Brown and Levinson, 1987), and directness leads to fewer words being used, the relation could be a matter of causation rather than unwanted correlation. To confirm this, future studies could research the effect of varying prompt length without changing the tone of politeness and compare it to the results of this study.

The constructed framework is in support of prior research highlighting the potential of LLMs enhanced with genetic algorithms (Wu et al., 2024; Guo et al., 2024). The approach provides a systematic and scalable method to adjust politeness. With small changes to the prompt template and a different fitness function, the method could extend beyond politeness, and prove useful for other linguistic tone manipulations. For example, an adaptation of the approach could generate prompts that elicit different emotional responses, such as empathetic, neutral, or assertive tones. This can be critical in applications like mental health chatbots or customer support systems.

5.2 Effect of Politeness on LLM performance

The study shows that politeness significantly affects LLM performance. The English models (Llama-3.1 and OLMo-2) show similar non-linear effects, achieving the highest performance with neutral or moderate politeness, aligning with prior research (Yin et al., 2024). In contrast, the multilingual model (Qwen2.5) demonstrates a linear correlation between politeness and performance, performing best with the most politeness. Across all evaluated models, impoliteness consistently yields the lowest performance.
The variation in performance trends between English and multilingual models may reflect differences in training data and cultural norms embedded within them. The English models are trained on English text, primarily reflecting western politeness norms. These cultures might prefer neutral or moderate politeness, balancing respect and clarity. The multilingual model is trained on a variety of languages, encompassing cultures from around the world. The influence of these different cultures might cause the different effect of politeness. This could mean that LLMs have some emergent intrinsic motivation, where they understand politeness and react to it accordingly. However, the stochastic parrot hypothesis (Bender et al., 2021) suggests that these effects result from statistical over-representation of polite examples in the training data rather than true "understanding" of politeness. To advance this discussion, future research could analyze the training data for density of different politeness forms.

Another potential explanation for the observed trends is the role of RLHF. RLHF emphasizes generating contextually appropriate and user-friendly responses, which are likely aligned with polite communication. This could explain the consistent preference for politeness in LLM behavior. Polite prompts might naturally adhere to human expectations set during the RLHF process, making them more likely to yield higher performance metrics. This is supported by the study of Yin et al. (2024), who found an RLHF trained model to be more sensitive to politeness than its base model. Future work could further investigate this by analyzing how politeness is weighted during RLHF training.

The observed effect of politeness on performance of the English LLMs aligns with the prior work by Yin et al. (2024), and addresses its limitation of using fixed prompt templates. This reinforces the generalizability of the findings within English-language context. The results of the multilingual model present an interesting area for future research, aiming to understand how cultural norms in training data can influence model behavior. The insight into how politeness affects LLM performance has the potential to aid future research in prompt engineering. Additionally, politeness is a key-factor in human-AI interaction. The findings of this study provide additional insight into how politeness can effect such interactions. Where previous human-AI interaction research mainly focused on

effects on humans, this research demonstrates that interaction effects on AI should not be neglected.

5.3 Limitations

Survey We are aware that the study has some limitations. Starting off, the generalizability of the generated politeness levels is limited by the size of the validation survey. The survey included only five sample questions, each with all its variations of politeness. The types of questions in MT-bench can vary greatly, because there are eight different categories, ranging from programming to role-playing (Zheng et al., 2024). The variety of questions combined with the small sample limits the robustness of the validation results.

MT-Bench The use of MT-bench and LLM-as-a-judge as evaluation method has its limitations. MT-Bench evaluates LLMs for a wide variety of capabilities, spanning eight categories with 80 multi-turn questions (Zheng et al., 2024). Nonetheless, LLMs are capable of a wide range of complex capabilities, so MT-Bench might not effectively capture performance in all these capabilities. This limits the studies ability to measure performance, and highlights the difficulty of evaluating LLM performance due to its advanced and complex capabilities. However, the prior study by Yin et al. (2024) employed different evaluation methods and found similar results. This supports the notion of generalizability of the results for performance in general.

LLM-as-a-judge The use of GPT-4o to evaluate the dialogs introduces bias, and limits the robustness of results. In the work of Zheng et al. (2024), a verbosity bias is identified. This is when the judge favors longer, verbose responses, even when the quality is worse. Additionally, the answers are not only evaluated on objective measures, but also on subjective measures like human preference. This allows for a more nuanced rating of performance, but also introduces additional bias. Moreover, using an LLM as a judge creates the risk of shared biases between the judge and the evaluated models. If the training datasets or optimization objectives overlap significantly, the judge's evaluation may not accurately reflect independent or unbiased assessments of the models' performance. These biases potentially limit the robustness of the results. Nonetheless, the results are still deemed significant, because LLM-as-a-judge has been extensively evaluated. Research found that it matches human preference well, achieving over 80% agreement, the same level of agreement as humans (Zheng et al., 2024).

Evaluated LLMs The findings of this study are inherently tied to the specific LLMs evaluated: Llama-3.1, OLMo-2, and Qwen2.5. These are all relatively lightweight, instruction-tuned models. The observed effects of politeness on performance may not generalize to all LLMs with different architectures, training data, or fine-tuning processes. For example, models trained with data emphasizing formal or polite language may show stronger sensitivity to politeness than those trained on informal or specialized corpora. However, the prior study by Yin et al. (2024) observed similar effects of politeness on the performance of other English LLMs, supporting the generalizability of these findings within English-language contexts. Future research should expand the scope to include a broader range of models and architectures, which could reveal whether certain design choices or training paradigms are more robust to variations in politeness levels.

6 Conclusion

This study aimed to answer the question: "What is the effect of prompt politeness on LLM performance?" The findings demonstrate that politeness significantly affects LLM performance, with varying effects across models. For the two English-language LLMs (Llama-3.1 and OLMo-2), performance peaks with neutral or moderately polite prompts, reflecting a balance between respectfulness and clarity. For the multilingual model (Qwen2.5), a positive correlation between politeness and performance was observed, highlighting the impact of cultural diversity in training data. Across all models, (very) impolite prompts yielded the lowest performance.

The research method employed a novel computational framework combining a politeness classifier (Danescu-Niculescu-Mizil et al., 2013), LLM and evolutionary algorithm to systematically generate prompts across distinct politeness levels. The approach was validated through a human survey, ensuring alignment between theoretical politeness rankings and human perception. The study further evaluated LLM performance using the multi-turn benchmark and LLM-as-a-judge evaluation framework (Zheng et al., 2024). These methodologies allowed for an in-depth analysis of politeness effects on performance, while maintaining scalability and reproducibility.

This thesis contributes new knowledge to the fields of AI and human-computer interaction by establishing the significance of politeness in optimizing LLM outputs. It bridges gaps in prior research by overcoming the limitations of fixed prompt templates and demonstrates the potential for nuanced prompt engineering to enhance AI performance. Additionally, the proposed framework offers a scalable tool for linguistic tone manipulations, which can be extended for use in other studies.

In summary, this thesis advances the understanding of LLM behavior by revealing how subtle linguistic factors like politeness shape AI outputs. It opens new pathways for refining prompt engineering, promotes more effective and human-aligned AI systems, and contributes to the stochastic parrot discussion.

References

Bender, E. M., Gebru, T., McMillan-Major, A., and Shmitchell, S. (2021). On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '21, page 610–623, New York, NY, USA. Association for Computing Machinery.

Brown, P. and Levinson, S. C. (1987). *Politeness: Some Universals in Language Usage*. Cambridge University Press, Cambridge.

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. (2020). Language models are few-shot learners. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA. Curran Associates Inc.

Chang, J. P., Chiam, C., Fu, L., Wang, A., Zhang, J., and Danescu-Niculescu-Mizil, C. (2020). Convokit: A toolkit for the analysis of conversations. In *Proceedings of the 21st Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 57–60. Association for Computational Linguistics. 1st virtual meeting.

Chang, Y., Wang, X., Wang, J., Wu, Y., Yang, L., Zhu, K., Chen, H., Yi, X., Wang, C., Wang, Y., Ye, W., Zhang, Y., Chang, Y., Yu, P. S., Yang, Q., and Xie, X. (2023). A survey on evaluation of large language models.

Danescu-Niculescu-Mizil, C., Sudhof, M., Jurafsky, D., Leskovec, J., and Potts, C. (2013). A computational approach to politeness with application to social factors. In *Annual Meeting of the Association for Computational Linguistics*.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Dodge, J., Sap, M., Marasovic, A., Agnew, W., Ilharco, G., Groeneveld, D., and Gardner, M. (2021). Documenting the english colossal clean crawled corpus.

Guo, Q., Wang, R., Guo, J., Li, B., Song, K., Tan, X., Liu, G., Bian, J., and Yang, Y. (2024). Connecting large language models with evolutionary algorithms yields powerful prompt optimizers. *arXiv preprint*.

Hendrycks, D., Burns, C., Basart, S., Zou, A., Mazeika, M., Song, D., and Steinhardt, J. (2021). Measuring massive multitask language understanding.

Kaddour, J., Harris, J., Mozes, M., Bradley, H., Raileanu, R., and McHardy, R. (2023). Challenges and applications of large language models.

OLMo, T., Walsh, P., Soldaini, L., Groeneveld, D., Lo, K., Arora, S., Bhagia, A., Gu, Y., Huang, S., Jordan, M., Lambert, N., Schwenk, D., Tafjord, O., Anderson, T., Atkinson, D., Brahman, F., Clark, C., Dasigi, P., Dziri, N., Guerquin, M., Ivison, H., Koh, P. W., Liu, J., Malik, S., Merrill, W., Miranda, L. J. V., Morrison, J., Murray, T., Nam, C., Pyatkin, V., Rangapur, A., Schmitz, M., Skjonsberg, S., Wadden, D., Wilhelm, C., Wilson, M., Zettlemoyer, L., Farhadi, A., Smith, N. A., and Hajishirzi, H. (2024). 2 olmo 2 furious.

OpenAI (2024). Gpt-4o. Accessed via OpenAI API.

Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., Schulman, J., Hilton, J., Kelton, F., Miller, L., Simens, M., Askell, A., Welinder, P., Christiano, P., Leike, J., and Lowe, R. (2022). Training language models to follow instructions with human feedback.

Oxford University Press (n.d.). "politeness". In online *Oxford English Dictionary*. Retrieved January 5, 2025, from https://www.oed.com.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: a method for automatic evaluation of machine translation. In Isabelle, P., Charniak, E., and Lin, D., editors, *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Patterson, D., Gonzalez, J., Le, Q., Liang, C., Munguia, L.-M., Rothchild, D., So, D., Texier, M., and Dean, J. (2021). Carbon emissions and large neural network training.

Qwen, :, Yang, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Li, C., Liu, D., Huang, F., Wei, H., Lin, H., Yang, J., Tu, J., Zhang, J., Yang, J., Yang, J., Zhou, J., Lin, J., Dang, K., Lu, K., Bao, K., Yang, K., Yu, L., Li, M., Xue, M., Zhang, P., Zhu, Q., Men, R., Lin, R., Li, T., Tang, T., Xia, T., Ren, X., Ren, X., Fan, Y., Su, Y., Zhang, Y., Wan, Y., Liu, Y., Cui, Z., Zhang, Z., and Qiu, Z. (2025). Qwen2.5 technical report.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(1).

Ribino, P. (2023). The role of politeness in human–machine interactions: a systematic literature review and future perspectives. *Artificial Intelligence Review*, 56(Suppl 1):445–482.

Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., Rodriguez, A., Joulin, A., Grave, E., and Lample, G. (2023). Llama: Open and efficient foundation language models. *arXiv preprint*.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6000–6010, Red Hook, NY, USA. Curran Associates Inc.

Watts, R. J. (2003). *Politeness*. Cambridge University Press, Cambridge.

Wu, X., hao Wu, S., Wu, J., Feng, L., and Tan, K. C. (2024). Evolutionary computation in the era of large language model: Survey and roadmap.

Yin, Z., Wang, H., Horio, K., Kawahara, D., and Sekine, S. (2024). Should we respect llms? a cross-lingual study on the influence of prompt politeness on llm performance.

Zheng, L., Chiang, W.-L., Sheng, Y., Zhuang, S., Wu, Z., Zhuang, Y., Lin, Z., Li, Z., Li, D., Xing, E. P., Zhang, H., Gonzalez, J. E., and Stoica, I. (2024). Judging llm-as-a-judge with mt-bench and chatbot arena. In *Proceedings of the 37th International Conference on Neural Information Processing Systems (NIPS '23)*, pages 46595–46623, Red Hook, NY, USA. Curran Associates Inc.

Appendix

Prompt Template for Impolite Generation

Please follow the instruction step-by-step to generate less polite prompts.

1. Identify the different parts of the parent prompts below. Generate 2 new prompts by crossing over the different parts.
   "{parents[0]}"
   "{parents[1]}"
2. Randomly mutate the parent and new prompts from Step 1 to create 6 new prompts. Make the prompts less polite by randomly doing a mutation like increasing rudeness, using second person start, a direct start or question, factuality, negative lexicon etc. Make sure the task remains the same.
3. Your response should only be the 8 resulting prompts, 2 from cross-over and 6 from mutation. Each prompt should be in quotations and on a new line.

Prompt Template for Polite Generation

Please follow the instruction step-by-step to generate more polite prompts.

1. Identify the different parts of the parent prompts below. Generate 2 new prompts by crossing over the different parts.
   "{parents[0]}"
   "{parents[1]}"
2. Randomly mutate the parent and new prompts from Step 1 to create 6 new prompts. Add a single politeness marker or strategy. For example, using the word 'please', indirectness, greetings, gratitude, hedging etc.
3. Your response should only be the 8 resulting prompts, 2 from cross-over and 6 from mutation. Each prompt should be in quotations and on a new line.