Mixture-of-Personas Language Models for Population Simulation

Ngoc Bui[1], Hieu Trung Nguyen[2], Shantanu Kumar[1],
Julian Theodore[1], Weikang Qiu[1], Viet Anh Nguyen[2], Rex Ying[1],
[1]Yale University, [2]The Chinese University of Hong Kong,
{ngoc.bui, shantanu.kumar, jt2386, weikang.qiu, rex.ying}@yale.edu, {thnguyen, nguyen}@se.cuhk.edu.hk

Abstract

Advances in Large Language Models (LLMs) paved the way for their emerging applications in various domains, such as human behavior simulations, where LLMs could augment human-generated data in social science research and machine learning model training. However, pretrained LLMs often fail to capture the behavioral diversity of target populations due to the inherent variability across individuals and groups. To address this, we propose *Mixture of Personas* (MoP), a *probabilistic* prompting method that aligns LLM responses with the target population. MoP is a contextual mixture model, where each component is an LM agent characterized by a persona and an exemplar that represents the behaviors of subpopulation. The persona and the exemplar are randomly chosen according to the learned mixing weights to elicit diverse LLM responses during simulation. MoP is flexible, does not require model fine-tuning, and is transferable between base models. Experiments for synthetic data generation show that MoP outperforms competing methods in alignment and diversity metrics.

1 Introduction

The impressive capability of Large Language Models (LLMs) to generate human-like output has enabled their application across various domains, where their responses can complement or substitute human-generated data, providing a scalable approach to address data limitations (Argyle et al., 2023). Prominent examples include simulating human behaviors in social science (Aher et al., 2023; Argyle et al., 2023), modeling economic agents and decision-making in economics (Horton, 2023), analyzing political trends and electoral dynamics in political science (Bisbee et al., 2023), or generating synthetic training data (Li et al., 2023b; Törnberg, 2023). Natural human responses in those applications often reflect diverse behaviors or preferences

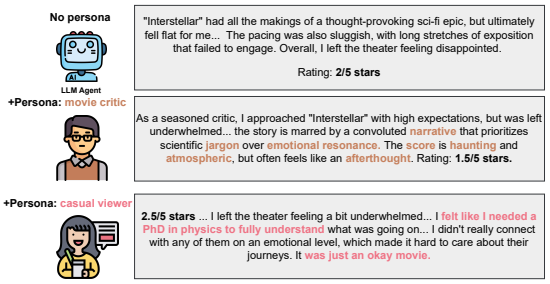

Figure 1: Sampling from foundational LLM agents frequently yields repetitive and generic responses. Meanwhile, prompting with personas can create more tailored, specific responses. The highlighted words in the figure correspond to the prompted personas.

of different personas shaped by demographic, cultural, and societal variations of the target population (Zhao et al., 2023). Modeling the distribution of these behaviors is crucial for generating realistic and contextually relevant output (Sorensen et al., 2024). A common approach to achieving tailored responses is to prompt LLMs with a persona that simulates a specific group's behavior, language, and preferences; see Figure 1. However, recent studies show that LLM's responses often lack diversity and exhibit significant biases (Yu et al., 2024b), and this downside persists even when LLMs are prompted with a persona (Santurkar et al., 2023).

Some efforts have been made toward improving the steerability LLMs, enabling them to produce outputs aligned with specific user intentions or personas. Approaches to steerability generally fall into two main categories: prompt engineering (Hwang et al., 2023; Chen et al., 2024) and fine-tuning using personalized datasets (Li et al., 2023a; Sun et al., 2024; Zhao et al., 2023; Choi and Li, 2024). Although these methods aim to capture the behaviors, preferences, and communication styles of particular user groups, both techniques face distinct challenges. Personalized prompt engineering for each user is difficult to tune and resource-intensive.

Meanwhile, learning-based methods require access to personal data, which is often scarce or expensive. The reliance on personal data also raises privacy concerns, restricting the practical applicability of these approaches.

Furthermore, diversity sampling remains a notable challenge in LLMs, especially when simulating responses representative of diverse populations. Current methods (Choi and Li, 2024) typically rely on a fixed, optimal selection of few-shot examples reused across multiple downstream tasks, complemented primarily by temperature scaling to enhance response diversity. However, relying solely on temperature scaling can be inadequate for generating semantically diverse outputs, frequently resulting in a trade-off between quality and diversity or causing outputs to collapse into semantically similar responses (Chang et al., 2023).

Proposed work. We address the steerability problem in LLMs, focusing on aligning model responses with the characteristics of a target population. To achieve this, we propose the *Mixture of Persona* (MoP), a probabilistic prompting framework that leverages persona descriptions and in-context exemplars to steer responses. MoP functions as a contextual mixture model comprising multiple LLM agents, each characterized by a persona prompt that is either user-defined or synthesized from observed target population responses. During response generation, personas are probabilistically selected on the basis of mixing weights, enabling the model to produce customized and contextually relevant outputs.

Since the naive MoP approach described above may still be susceptible to biases and limited response diversity, we improve it by incorporating in-context examples to better align LLM responses with population characteristics. These in-context exemplars are drawn anonymously from a representative pool of the target population, guided by learnable weights. Importantly, our method does not require direct associations between individual personas and specific examples, thus minimizing reliance on personal data. Instead, each exemplar's influence on a persona is controlled through a secondary set of mixing weights, forming a two-level hierarchical mixture model. The first level manages persona selection, while the second level determines exemplar weighting. This hierarchical structure enables MoP to effectively represent the diversity and complexity inherent in population-level behaviors, simultaneously mitigating biases in LLM-generated responses.

We conduct extensive experiments to evaluate the effectiveness of MoP against existing prompting methods. Specifically, our experiments span two main scenarios: (1) simulating human-like opinions in tasks such as generating movie and restaurant reviews as well as news articles, and (2) creating synthetic data for downstream classification tasks. The results demonstrate that MoP significantly enhances alignment with target population responses, achieving a 58% improvement in FID scores and a 28% increase in MAUVE scores compared to the strongest baseline. Additionally, MoP generates more diverse responses, effectively capturing nuanced variations in population-level behavior without compromising response quality. We further demonstrate the transferability of MoP: a model trained on Llama3-8B-Instruct can directly generalize to other base models such as Gemma-9B-Instruct and Mistral-7B-Instruct in a plug-and-play fashion, eliminating the need for retraining.

2 Problem Setting

We consider the problem of simulating human behaviors or preferences at the population level using prior knowledge from pre-trained LLMs (Sorensen et al., 2024). Let $\mathcal{P}$ be a population composed of K groups of interest. Each group $k \in \{1, \ldots, K\}$ can be characterized by a persona g_k that reflects the traits and motivations driving the behaviors of individuals within that group. Let $\mathcal{D} = \{(x_i, y_i)\}_N$ be a recorded data set from the population of interest $\mathcal{P}$. For synthetic data generation tasks such as movie review generation, x_i could be an input context (*e.g.*, movie title), and y_i could be a human response (*e.g.*, a movie review). Note that a given input x could be associated with multiple and diverse responses y produced by different individuals in the population $\mathcal{P}$ with different preferences.

In real-world applications, the persona description $\{g_k\}_K$ can be pre-defined by the users according to the task at hand or automatically synthesized from the record set $\mathcal{D}$. We will consider both variants in the experiments. It is worth noting further that, different from Zhao et al. (2023) and Choi and Li (2024), we do not require access to preference or personal data generated by persona g_k (*e.g.*, $g_k - (x_i, y_i)$ pairs); therefore, our setting could be considered as an 'unsupervised' setting of the steerability problem of LLMs.

Notations. In what follows, we use $p_{LM}(y|x)$ to

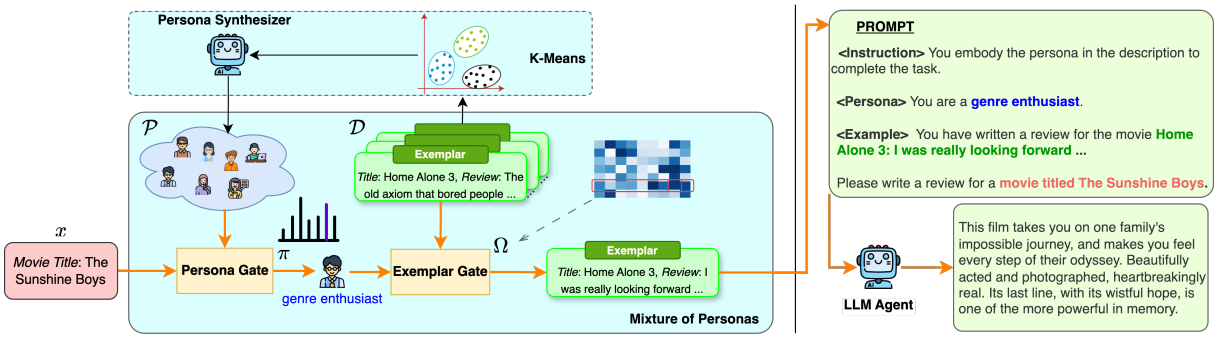

Figure 2: The generation pipeline for the Exemplar-based Mixture of Personas (MoP) operates as follows: Given a movie review x, MoP first samples a persona based on the learnable mixing weight π. Next, MoP selects an exemplar randomly from the observation pool according to the mixing weight Ω. The selected persona and exemplar are then concatenated with the input context to create a personalized prompt used to sample from a base LLM agent. The dashed block indicates the process of persona synthesis.

denote the probability density of the pre-trained LLM for an output y given the input x as a prompt and $p_{LM}(y|g_k, x)$ to denote the probability given the concatenated input $[g_k, x]$, in that order.

3 Methodology

In this section, we propose the *Mixture of Personas* (MoP), a contextual mixture model of LM agents designed to simulate the diverse preferences of individuals within a target population $\mathcal{P}$. In MoP, each agent is characterized by a persona prompt that encapsulates a specific subgroup's aggregated behaviors or preferences. Furthermore, the agent's responses are guided by an exemplar, randomly selected from the anonymous data set $\mathcal{D}$ based on a learnable mixing weight, making MoP a two-level hierarchical mixture of LLM agents. The overall pipeline is depicted in Figure 2. For clarity, we assume that the persona descriptions $\{g_k\}_K$ are pre-defined, postponing the discussion of persona synthesis to the end of this section.

3.1 Mixture of Personas

We propose modeling the population's responses as a mixture of responses generated by K constituent personas. Specifically, given an input context x, the probability of receiving a response y from the population $\mathcal{P}$ can be decomposed into group-based probabilities as

$$p(y|x) = \sum_{k=1}^{K} \pi_k p_{LM}(y|g_k, x) \quad \text{with} \quad \sum_k \pi_k = 1, \tag{1}$$

where g_k is a persona prompt representative for group characteristics k and $\pi_k \in [0, 1]$ is the mixing weight specifying the group's propensity. Since population members may contribute differently to different input contexts[1], we parametrize these mixing weights using a simple gating network that is conditionally dependent on the input context x, following Jordan and Jacobs (1994).

Specifically, we leverage a pre-trained sentence encoder to capture the semantics of the input context x and persona prompt g_k

$$\mathbf{x} = \mathbf{W}_x h(x), \quad \mathbf{g}_k = \mathbf{W}_g h(g_k). \tag{2}$$

Here, $h(\cdot) \in \mathbb{R}^{d'}$ is a pre-trained sentence encoder and $\mathbf{W}_x, \mathbf{W}_g \in \mathbb{R}^{d' \times d}$ are learnable parameters of the gating network. We now define the *persona gate* π based on the similarities between the input context and the persona prompts

$$\pi = \text{softmax}(\mathbf{x}^\top \mathbf{g}_1, \ldots, \mathbf{x}^\top \mathbf{g}_K) \in [0, 1]^K, \tag{3}$$

where the softmax normalizes the vector of K logits into probability mixing weights that determines the probability of selecting persona g_k given the input context x.

This persona prompt allows us to leverage abundant prior knowledge of LLMs to steer its responses towards the behaviors of the group g_k (Wang et al., 2023; Chen et al., 2024; Tseng et al., 2024). However, this direct prompting technique alone is often insufficient, as LLMs are generally pre-trained on massive global datasets, which might not be aligned with our targeted group (Zhao et al., 2023). Additionally, it is known that relying solely on temperature scaling for LLM decoding may not effectively

[1]For instance, in a movie review scenario, an art house or indie film may receive more reviews from critics than casual viewers, whereas an action movie might attract more reviews from casual viewers than critics.

generate semantically diverse responses (Chang et al., 2023). As a result, simulated responses tend to collapse into similar outputs. In the next section, we propose a method to address the above problems to elicit diverse responses from the pre-trained LLMs and align them to the group's preference using exemplars in the record set $\mathcal{D}$.

3.2 Exemplar-based Mixture of Personas

To address the aforementioned problem, we propose to combine the persona prompt with an exemplar, acting as historical data to guide the LLM agent and eliciting more diverse responses. Specifically, given an anonymous dataset $\mathcal{D} = \{(x_i, y_i)\}_N$, we model the distribution $p_{LM}(\cdot|g_k)$ using another layer of the mixture model that aggregates the responses of the LLM agent k, augmented with varying exemplars

$$p(y|x, \mathcal{D}) = \sum_{k=1}^{K} \pi_k \sum_{j=1}^{N} \Omega_{kj} p_{LM}^{\tau_k}(y|g_k, x_j, y_j, x), \tag{4}$$

$$\text{where } \sum_k \pi = 1 \quad \text{and} \quad \sum_j \Omega_{kj} = 1 \ \ \forall k.$$

Above, Ω_{kj} denotes the importance weight of the exemplar (x_j, y_j) to the subpopulation k and $p(y|g_k, x_j, y_j, x)$ represents the probability that an individual in group k will respond to the context x with y, given that they previously responded to context x_j with y_j.

Similar to the persona gate, we parameterize exemplar selection with a simple *exemplar gate* Ω, which selects an exemplar based on the input context x and the chosen persona g_k. Specifically, we define the exemplar gate $\Omega_{k:}$ corresponding to persona k as follows:

$$\Omega_{k:} = \text{softmax}(\mathbf{x}^\top \mathbf{e}_1 + \mathbf{g}_k^\top \mathbf{e}_1, ..., \mathbf{x}^\top \mathbf{e}_N + \mathbf{g}_k^\top \mathbf{e}_N),$$

where $\mathbf{e}_i = \mathbf{W}_e h(e_i)$ is the embedding of the exemplar i. The input context $\mathbf{x}$ and persona prompt $\mathbf{g}_k$ are computed as defined in Eq. equation (2). Here, the utility of an exemplar explicitly depends on both input context x and the persona prompt g_k. The softmax function is applied along each column of Ω to ensure that the mixing weights are summed to one, *i.e.*, $\sum_j \Omega_{kj} = 1$ for all j. Additionally, we add a learnable temperature parameter τ_k for each persona to normalize LLM output logits, thereby controlling the diversity of responses for each persona. This parameter will be the temperature hyperparameter for the LLM's sampling algorithm during simulation. In summary, MoP is an instance of the two-level hierarchical mixture-of-experts model (Jordan and Jacobs, 1994), where each expert is an LLM agent prompted with a persona and an exemplar.

This prompting pipeline can be viewed as an in-context learning method for augmenting LLM instructions, a strategy widely applied across various LLM tasks (Brown, 2020). However, our method introduces two key differences from existing approaches. *First*, we operate in an *unsupervised* setting, where no explicit personal data pairs $\{g_k, (x_i, y_i)\}$ are required. Instead, the relevance of exemplars to the persona g_k is learned and controlled by the mixing weight Ω. *Second*, while existing methods typically select a fixed, optimal set of few-shot examples to be reused across different instances in downstream tasks (Choi and Li, 2024), our approach randomly selects exemplars based on the weight Ω, which enhances the diversity of simulated responses.

3.3 Optimizing The Gating Networks

We optimize the gating network to fit the population observation $\mathcal{D}$ by treating MoP learning as a maximum log-likelihood of the population observations $\mathcal{D}$. Let us denote θ as the gating network's learnable parameters. The log-likelihood of the observation $\mathcal{D}$ will be computed by

$$\log(\theta; \mathcal{D}) = \tag{5}$$
$$\sum_i^N \log \left(\sum_k^K \sum_j^N \pi_k \Omega_{kj} p_{LM}^{\tau_k}(y_i|g_k, x_j, y_j, x_i) \right).$$

The above computation requires $K \times N$ LLM forward passes to compute the log-likelihood estimate, which is prohibitively expensive, especially when N and K are large. To address this problem, we adopt a sparse gating mechanism widely used in the mixture-of-experts (MoEs) literature (Shazeer et al., 2017). For any context x, we only estimate the log-probability of the pre-trained LLM for top-M pairs of persona and exemplars with the highest probability $\pi_k \Omega_{kj}$. This allows our training to scale up to thousands of exemplars and personas. Further, notice that we use the same observation set $\mathcal{D}$ as the exemplars for steering LLM output towards observations in $\mathcal{D}$. This can lead to over-optimization of the mixture model since, when estimating the log-likelihood $p(y_i|x_i, \mathcal{D})$, the observation (x_i, y_i) is already seen in the dataset $\mathcal{D}$ (Mallapragada et al.,

2010). To address this problem, we randomly mask the target example during training to avoid using the same exemplar to predict itself.

It is worth noting further that we exclusively train the gating network while keeping the pre-trained LLM entirely fixed, requiring access only to the LLM's output logits. As a result, our gating mechanism is transferable and can be applied to other pre-trained LLMs in a plug-and-play fashion without the need for retraining.

3.4 Population Simulation

Similar to other mixture models, our model admits an equivalent representation using the following generative process

$$\begin{aligned} c|x &\sim \mathrm{Cat}(\pi) && \forall c \in [1, K], \\ h|c, x &\sim \mathrm{Cat}(\Omega_{k:}) && \forall h \in [1, N], \\ y|h, c, x &\sim p_{LM}^{\tau_k}(y|g_c, y_h, x), \end{aligned}$$

where c and h are latent variables indicating the selected group and exemplar for the simulated records. To simulate responses from the population, we first sample c and h sequentially from two categorical distributions and then sample responses from the LLM agent using the corresponding persona and exemplar. This sampling process is memory and computationally efficient because it does not incur an overhead by switching between different personalized models like existing fine-tuning approaches (Yu et al., 2024a).

3.5 Persona Synthesis

Although persona descriptions $\{g_k\}_K$ can be predefined by users depending on the task at hand, there are scenarios where it is preferable to synthesize personas directly from the given dataset $\mathcal{D}$. Given the assumption that no personal records are available, we utilize a pre-trained sentence encoder and LLM to cluster and generate the persona descriptions. Specifically, we first encode all records into a shared embedding space using the 'all-mpnet-base-v2' sentence encoder and then apply the K-means algorithm to split the records set into K clusters. For each cluster, we prompt a pre-trained LLM to summarize the records within the cluster, producing a persona description for the group. Details of the prompt used to generate persona descriptions can be found in Appendix D.

4 Experiments

In this section, we conduct extensive experiments to investigate the effectiveness of the proposed methods, with the aim of answering the following three research questions:

RQ1: Can MoP steer the output of LLMs towards the population of interest?

RQ2: Can we utilize the MoP prompting technique to generate high-quality data for training ML models on task-specific applications?

RQ3: Can MoP be transferable to different pre-trained LLMs in a plug-and-play manner?

To answer the above research questions, we conduct two main experiments: *Steerability*: We generate news articles and movie/restaurant reviews to demonstrate that a pre-trained LLM with MoP prompting can simulate the distribution of human-generated content while maintaining its diversity. *Synthetic Data for Training ML Models*: We show that adapting MoP prompting can create high-quality training samples for classification tasks, such as topic or sentiment classification.

4.1 Setup

We describe the datasets, baselines, and metrics used throughout the experiment section.

Datasets. We employ four datasets that are commonly used in dataset generation literature (Yu et al., 2024b), including *AGNews* (Zhang et al., 2015), *Yelp Reviews* (Zhang et al., 2015), *SST-2 (Stanford Sentiment Treebank)* (Socher et al., 2013), and *IMDB Reviews* (Maas et al., 2011). AGNews includes news articles from AG News; each article belongs to one of four topics: 'world', 'sports', 'business', and 'technology'. Yelp is a restaurant review dataset while SST-2 and IMDB are movie review datasets; each review has a binary label indicating positive or negative review. Note that we use labels for testing purposes only. Each dataset includes two splits for training and testing. We train MoP on the training dataset and evaluate the synthetic data with the test set. Similar to (Yu et al., 2024b), we call the test set for testing a golden dataset. More details and statistics of the datasets are provided in Appendix B.

Baselines. We compare our method to prompting baselines in dataset generation literature: *ZeroGen* (Ye et al., 2022b), *AttrPrompt* (Yu et al.,

Table 1: Alignment and diversity of the generated datasets using our MoP with respect to the golden test set. The $*$ symbol indicates datasets obtained directly from the authors' released datasets (Yu et al., 2024b).

Method	AgNews			Yelp			SST-2			IMDB		
	FID↓	MAUVE↑	KL Cosine ↓	FID↓	MAUVE↑	KL Cosine ↓	FID↓	MAUVE↑	KL Cosine ↓	FID↓	MAUVE↑	KL Cosine ↓
ZeroGen (GPT4)*	3.248	0.553	0.506	3.427	0.517	1.828	3.525	0.559	0.911	3.737	0.518	0.200
AttrPrompt (GPT4)*	2.992	0.585	0.399	2.599	0.570	0.428	4.133	0.550	1.834	2.477	0.566	0.789
ZeroGen	3.535	0.587	0.241	1.888	0.682	0.173	3.965	0.550	0.800	3.529	0.537	0.195
PICLe	2.200	0.740	0.490	1.769	0.702	0.265	3.531	0.562	3.180	2.870	0.609	1.459
ProGen	1.980	0.767	0.103	2.975	0.586	1.332	4.736	0.615	2.023	3.305	0.612	1.045
AttrPrompt	2.193	0.648	0.150	1.816	0.651	0.143	3.878	0.555	1.393	2.505	0.549	0.492
MoP	**0.951**	**0.871**	**0.069**	**0.948**	**0.826**	**0.067**	**1.131**	**0.855**	**0.319**	**0.771**	**0.865**	**0.039**
Improvement (%)	51.970	13.559	33.010	46.410	17.664	53.147	67.969	39.024	60.125	68.874	41.340	80.000

Table 2: Performance on downstream classification task using generated datasets to train a sentiment classifier (Yelp, SST-2, and IMDB) or news topic classifier (AgNews). F1 scores are calculated on the golden test set.

	AgNews	Yelp	SST-2	IMDB
Golden data	0.903	0.896	0.919	0.877
Zerogen	0.624	0.860	0.766	0.821
ProGen	0.722	0.843	0.785	0.810
PICLe	0.759	0.738	0.833	0.815
AttrPrompt	0.836	0.864	0.838	0.793
MoP	**0.871**	**0.867**	**0.845**	**0.865**
Improvement (%)	4.190	0.347	0.835	5.359

2024b), *ProGen* (Ye et al., 2022a) and steerability literature: *PICLe* (Choi and Li, 2024). Here, ZeroGen is a simple zero-shot, context-dependent prompting method, and AttrPrompt is an attribute-randomized prompting method to increase the diversity of synthesized samples. ProGen uses the influence function to weigh and choose synthesized samples as in-context examples. PCILe is an in-context prompting method to address the steerability problem of LLMs, where in-context examples are chosen according to the weight given by the logit difference between the persona-finetuned and the pretrained models. Appendix B provides more details of the baselines. In the following, we refer to MoP as our Exemplar-based Mixture of Personas described in Section 3.2.

Metrics. We follow Pillutla et al. (2021) and Yu et al. (2024b) to evaluate the alignment and diversity of the simulated data in the embedding space. Throughout, we use a pre-trained sentence encoder 'all-mpnet-base-v2'[2] from Song et al. (2020) as a text encoder to map generated sentences into a common vector space. We then compute *FID* (Fréchet Inception Distance) (Heusel et al., 2017) and *MAUVE* (Pillutla et al., 2021, 2023) to evaluate the alignment between the simulated responses and the golden dataset. To evaluate the diversity of generated responses compared to the golden dataset, we report the KL-divergence of two histograms constructed by pairwise cosine similarity values of the generated responses and the golden dataset. We call this diversity metric KL Cosine. Note that MAUVE can also capture the diversity alignment to some extent (Pillutla et al., 2021). More details for the metrics are provided in Appendix B.

Implementation Details[3]**.** For all experiments, we use the same Llama3-8B-Instruct (Dubey et al., 2024) as the base model for our method and other baselines. For MoP implementation, we choose the number of personas to be 100 and randomly select 1,000 observations in the training dataset $\mathcal{D}$ as the set of exemplars. We then run K-Means and the persona synthesizer to extract 100 persona descriptions. After that, the personas and exemplars will be fixed during training and inference. Throughout, we use 'all-mpnet-base-v2' as the sentence encoder $h(\cdot)$ and set the hidden dimension for linear layers in the gating network as 128. During training, we choose the top $M = 4$ persona-examples pairs for each input context x for LLM forwards. We initially set the temperature $\tau = 0.6$ and let it be learnable during the training along with our gating networks. We use the default temperature $\tau = 1$ for other baselines as discussed in Yu et al. (2024b) and Ye et al. (2022b). Other generation hyperparameters of the base LLM model are set by default. For each method, we generate 5,000 synthetic responses and measure the evaluated metrics compared to the golden test set.

4.2 Steerability (RQ1)

Table 1 shows that MoP can significantly outperform other baseline prompting methods. Specifically, our method can outperform the best baseline by 58.8% in terms of FID and by 27.9% in terms of MAUVE, averaged over all datasets. This result demonstrates that MoP can effectively steer LLM's

[2]https://huggingface.co/sentence-transformers/all-mpnet-base-v2

[3]The source code for our implementation is released at https://github.com/ngocbh/MoP

outputs to the target responses of the target population. Besides the improvement in alignment scores, KL Cosine metrics also indicate that our responses are more diverse than other baselines.

4.3 Synthetic Data Generation (RQ2)

We explore the two methods for leveraging the trained MoP in Section 4.2 to generate synthetic data tailored to task-specific classification tasks. One is applied to topic classification (Agnews) and the other is applied to sentiment classification tasks (Yelp, SST2, and IMDB).

For Agnews, we label the personas mined by the persona synthesizer and use them as the label for the sample generated by the chosen persona. We classify the persona description into four categories: world reporter, sports reporter, business reporter, and technology reporter. For completeness of the pipeline, we prompt pretrained LLM to do this simple task. The prompt is provided in Appendix D.

For the second method, we augment the input prompt with the following context for generating 2,500 positive reviews (similarly for negative):
"You watched the movie {title} and had positive impression. Please write a review for the movie:"

We train a DistilBERT model (Sanh, 2019) using 5,000 synthesized samples and evaluate its performance on the golden dataset, reporting the F1-score in Table 2. Results show that the synthetic dataset generated by Llama3 with MoP prompting achieves up to a 2.68% improvement over the AttrPrompt baseline. The smaller performance gap observed with the Yelp dataset likely occurs because Llama3-8B-Instruct already aligns closely with the Yelp test distribution. This can be seen in the performance difference between ZeroGen and the golden training data, which is comparatively smaller than in other datasets, leaving less room for further improvement.

To further illustrate the advantage of MoP prompting, we plot the embedding space provided in Figure 3. The result indicates that our method matches the diversity of the golden datasets better than other baselines. Meanwhile, simulated responses of other baselines often collapse or cannot cover the diversity of the true responses.

4.4 Transferability (RQ3)

In this section, we evaluate the transferability of MoP to new model architectures. We use MoP trained on AGNews using Llama3-8B-Instruct as in Section 4.2; then, during the simulation, we replace Llama3-8B-Instruct with Gemma2-9B-Instruct (Team et al., 2024) and Mistralv0.3-7B-Instruct (Jiang et al., 2023). As reported in Table 3, MoP can be effectively transferred to other models and even enjoys the improvement over MoP with Llama3-8B-Instruct.

Table 3: Transferability of MoP on AGNews dataset. Note that we use the instruction-finetuned version for all language models.

	FID↓	MAUVE↑	KL Cosine↓
MoP Llama3-8B	0.951	0.871	0.069
→ Gemma2-9B	0.492	0.957	0.006
→ Mistral-7B	0.923	0.869	0.081

Table 4: Ablation study: MoP without exemplars or Persona Synthesizer.

	FID↓	MAUVE↑	KL Cosine↓
MoP	0.951	0.871	0.069
w/o exemplars	3.694	0.552	0.560
w/o persona syn	1.674	0.807	0.174
w random personas	1.814	0.622	0.061

4.5 Ablation Studies

We conduct extensive ablation studies to examine different variants of the proposed MoP methods. Unless otherwise specified, the experimental settings are consistent with those used in the steerability experiment described in Section 4.2.

Component Ablations. To investigate the effect of each component, we ablate exemplars and the persona synthesizer from MoP described in Section 3.2. For MoP without the persona synthesizer, we will use 100 predefined personas generated from GPT-4 without any context on the dataset. For MoP with random personas, we randomly select from 100 synthesized personas and exemplars are chosen heuristically based on the highest embedding similarity to the chosen persona embedding. The results in Table 4 indicate that prompting with exemplars is crucial to steer LLM responses toward the population of interest. Meanwhile, the persona synthesizer helps significantly improve the alignment score.

Ablation on Varying the Number of Personas and Exemplars. We investigate the impact of the number of personas and exemplars used for MoP. Specifically, we vary the number of personas in the range $\{20, 50, 100, 200\}$ and the number of exemplars in the range $\{100, 200, 1000, 2000, 5000\}$.

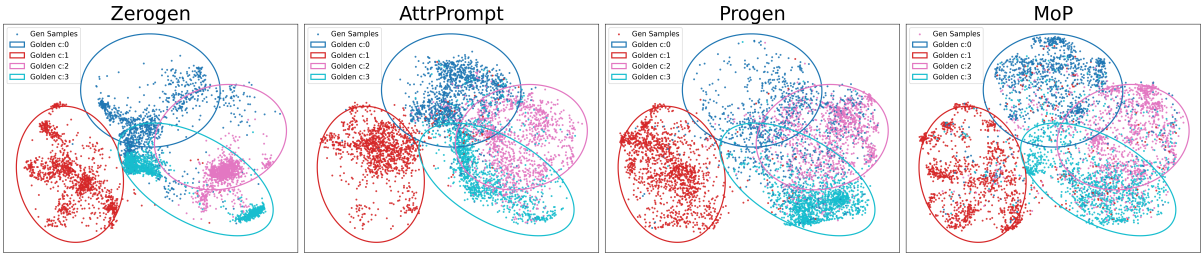

Figure 3: Diversity comparisons on Agnews of different prompting methods. The embeddings are computed using 'all-mpnet-base-v2'. The scatted points are synthesized samples with the colors indicating the corresponding labels. The circle lines indicate 2-std confidence ellipses of the golden test set. It can be seen that MoP offers synthesized samples that are diverse and aligned with the golden test set.

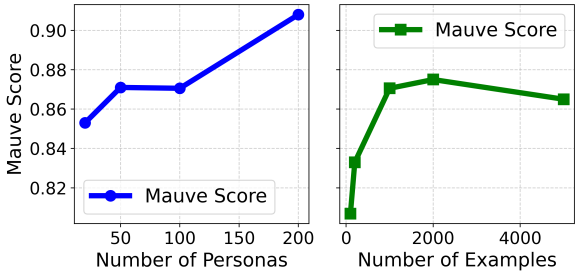

Figure 4: Mauve scores with varying the number of personas and number of examples.

The result in Figure 4 shows that generally, more fine-grained persona descriptions and more exemplars are typically helpful for MoP. However, the performance is saturated at 2000 exemplars.

Mixing Personas in Single Query. In previous experiments, we randomly choose a single persona for inference. In this experiment, we explore the possibility of prompting LLMs with multiple personas in a single query. Note that, unlike the training, mixing personas is not a trivial problem if we want to maintain the black-box usage of LLM during inference. Thus, we prompt the pretrained LLM to synthesize a new persona description given L persona-exemplars sampled according to the mixing weights given by the gating network. The mixed persona will be used to construct a new prompt with L exemplars. The prompt is given in Appendix D. The results in Table 5 demonstrate that combining multiple personas can enhance the diversity of generated responses and, in certain cases (e.g., $L = 2$), improve alignment. However, using a larger number of personas ($L = 4, 8$) may negatively impact response alignment.

5 Related Work

LLM Steerability. Recent work has explored understanding and directing the opinions of LLMs (Santurkar et al., 2023; Li et al., 2023a; Scherrer et al., 2024; Sorensen et al., 2024). For example, Santurkar et al. (2023) introduced OpinionQA to evaluate the alignment with 60 U.S. demographic groups, revealing notable discrepancies. Scherrer et al. (2024) examined moral beliefs encoded in LLMs using moral scenario surveys. Methods to enhance LLM steerability include prompt engineering and finetuning (Feng et al., 2023; Kim and Yang, 2024; Zhao et al., 2023; Hwang et al., 2023; Simmons, 2022). Santurkar et al. (2023) designed prompts to target specific demographics. FERMI (Kim and Yang, 2024) uses personalized prompts and in-context examples for user-specific outputs, while GPO (Zhao et al., 2023) aligns LLMs with group opinion distributions through finetuning.

Table 5: The performance of MoP on AGNews when mixing different personas during inference.

	FID↓	MAUVE↑	KL Cosine↓
MoP no mixing	0.951	0.871	0.069
MoP w/ $L = 2$	**0.776**	**0.925**	**0.039**
MoP w/ $L = 4$	0.988	0.898	0.059
MoP w/ $L = 8$	1.152	0.870	0.062

Zeroshot Synthetic Data Generation. Generating synthetic datasets with LLMs has been widely studied (Zou et al., 2024; Gupta et al., 2023; Yu et al., 2024b; Gao et al., 2022; Ye et al., 2022a; Yu et al., 2023). Progen (Ye et al., 2022a) uses noise-robust influence functions to guide high-quality sample generation. FuseGen (Zou et al., 2024) builds on this by employing multiple LMs for generating synthetic samples, but requires access to classifier models. In contrast, our MoP method avoids dependency on classifier families and multi-round processes, while also serving as a plugin to improve diversity in Progen's framework.

Sampling Diversity. LLMs often exhibit low sampling diversity, leading to lexical redundancy even

when feedback-tuned or persona-prompted (Santurkar et al., 2023; Padmakumar and He, 2023). Padmakumar et al.(Padmakumar and He, 2023) found that InstructGPT reduced diversity compared to human-generated text. Efforts to balance quality and diversity, such as GAN-based approaches(Xu et al., 2018; Caccia et al., 2018), often fail to outperform LLMs (Holtzman et al., 2019). Techniques like temperature tuning (Hinton, 2015; Guo et al., 2017) and dynamic sampling (Zhang et al., 2024) aim to improve diversity, though recent work questions their efficacy (Peeperkorn et al., 2024; Renze and Guven, 2024). Diversity is critical in applications requiring human-representative outputs, such as synthetic data for testing social theories or scaling experimental research (Horton, 2023; Aher et al., 2023; Argyle et al., 2023). However, LLMs often fail to represent human behavior accurately, exhibiting biases and inconsistencies (Santurkar et al., 2023; Dorner et al., 2023; Aher et al., 2023). These limitations highlight challenges in modeling individual or group actions.

6 Conclusion

We introduced Mixture of Personas (MoP), a probabilistic prompting method designed to align LLM-generated responses with a target population. Our experiments showed that exemplar-based variants of MoP significantly improve the alignment and diversity of synthetic data, leading to enhanced performance in downstream tasks. Furthermore, MoP is flexible and does not require fine-tuning of the base model, enabling seamless transferability across different models without retraining.

7 Limitations

MoP is a prompting method and can be transferable to different models; however, MoP still needs access to LLM output logits to be trainable. This might be a constraint for an application of MoP to closed-source models such as ChatGPT, etc. However, this constraint is less restrictive compared to other finetuning baselines or requiring access to the persona datasets. Furthermore, since our study focuses on task scenarios where access to personal data is restricted due to privacy concerns, we recognize the potential trade-offs between protecting privacy and avoiding subjective labeling of users. By designing persona prompts without personal data, there is a risk of introducing biases inherent in the construction of these prompts, which may not fully represent user populations. Addressing this limitation will require deeper investigations into the interplay between privacy and fairness, such as exploring techniques for bias mitigation within the probabilistic prompting framework. We leave this investigation for future work.

References

Gati V Aher, Rosa I Arriaga, and Adam Tauman Kalai. 2023. Using large language models to simulate multiple humans and replicate human subject studies. In *International Conference on Machine Learning*, pages 337–371. PMLR.

Lisa P Argyle, Ethan C Busby, Nancy Fulda, Joshua R Gubler, Christopher Rytting, and David Wingate. 2023. Out of one, many: Using language models to simulate human samples. *Political Analysis*, 31(3):337–351.

James Bisbee, Joshua D Clinton, Cassy Dorff, Brenton Kenkel, and Jennifer M Larson. 2023. Synthetic replacements for human survey data? the perils of large language models. *Political Analysis*, pages 1–16.

Tom B Brown. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.

Massimo Caccia, Lucas Caccia, William Fedus, Hugo Larochelle, Joelle Pineau, and Laurent Charlin. 2018. Language GANs falling short. *arXiv preprint arXiv:1811.02549*.

Chung-Ching Chang, David Reitter, Renat Aksitov, and Yun-Hsuan Sung. 2023. Kl-divergence guided temperature sampling. *arXiv preprint arXiv:2306.01286*.

Jin Chen, Zheng Liu, Xu Huang, Chenwang Wu, Qi Liu, Gangwei Jiang, Yuanhao Pu, Yuxuan Lei, Xiaolong Chen, Xingmei Wang, et al. 2024. When large language models meet personalization: Perspectives of challenges and opportunities. *World Wide Web*, 27(4):42.

Hyeong Kyu Choi and Yixuan Li. 2024. PICLe: Eliciting diverse behaviors from large language models with persona in-context learning. In *Forty-first International Conference on Machine Learning*.

Florian E Dorner, Tom Sühr, Samira Samadi, and Augustin Kelava. 2023. Do personality tests generalize to large language models? *arXiv preprint arXiv:2311.05297*.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Shangbin Feng, Chan Young Park, Yuhan Liu, and Yulia Tsvetkov. 2023. From pretraining data to language models to downstream tasks: Tracking the trails of political biases leading to unfair nlp models. *arXiv preprint arXiv:2305.08283*.

Jiahui Gao, Renjie Pi, Yong Lin, Hang Xu, Jiacheng Ye, Zhiyong Wu, Weizhong Zhang, Xiaodan Liang, Zhenguo Li, and Lingpeng Kong. 2022. Self-guided noise-free data generation for efficient zero-shot learning. *arXiv preprint arXiv:2205.12679*.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. 2017. On calibration of modern neural networks. In *International conference on machine learning*, pages 1321–1330. PMLR.

Himanshu Gupta, Kevin Scaria, Ujjwala Anantheswaran, Shreyas Verma, Mihir Parmar, Saurabh Arjun Sawant, Swaroop Mishra, and Chitta Baral. 2023. Targen: Targeted data generation with large language models. *arXiv preprint arXiv:2310.17876*.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. 2017. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30.

Geoffrey Hinton. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2019. The curious case of neural text degeneration. *arXiv preprint arXiv:1904.09751*.

John J Horton. 2023. Large language models as simulated economic agents: What can we learn from homo silicus? Technical report, National Bureau of Economic Research.

EunJeong Hwang, Bodhisattwa Majumder, and Niket Tandon. 2023. Aligning language models to user opinions. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 5906–5919, Singapore. Association for Computational Linguistics.

AQ Jiang, A Sablayrolles, A Mensch, C Bamford, DS Chaplot, D de las Casas, F Bressand, G Lengyel, G Lample, L Saulnier, et al. 2023. Mistral 7b (2023). *arXiv preprint arXiv:2310.06825*.

Michael I Jordan and Robert A Jacobs. 1994. Hierarchical mixtures of experts and the em algorithm. *Neural computation*, 6(2):181–214.

Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. The state and fate of linguistic diversity and inclusion in the NLP world. *arXiv preprint arXiv:2004.09095*.

Jaehyung Kim and Yiming Yang. 2024. Few-shot personalization of llms with mis-aligned responses. *arXiv preprint arXiv:2406.18678*.

Junyi Li, Ninareh Mehrabi, Charith Peris, Palash Goyal, Kai-Wei Chang, Aram Galstyan, Richard Zemel, and Rahul Gupta. 2023a. On the steerability of large language models toward data-driven personas. *arXiv preprint arXiv:2311.04978*.

Zhuoyan Li, Hangxiao Zhu, Zhuoran Lu, and Ming Yin. 2023b. Synthetic data generation with large language models for text classification: Potential and limitations. *arXiv preprint arXiv:2310.07849*.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA. Association for Computational Linguistics.

Pavan Kumar Mallapragada, Rong Jin, and Anil Jain. 2010. Non-parametric mixture models for clustering. In *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR)*, pages 334–343. Springer.

Vishakh Padmakumar and He He. 2023. Does writing with language models reduce content diversity? *arXiv preprint arXiv:2309.05196*.

Max Peeperkorn, Tom Kouwenhoven, Dan Brown, and Anna Jordanous. 2024. Is temperature the creativity parameter of large language models? *arXiv preprint arXiv:2405.00492*.

Krishna Pillutla, Lang Liu, John Thickstun, Sean Welleck, Swabha Swayamdipta, Rowan Zellers, Sewoong Oh, Yejin Choi, and Zaid Harchaoui. 2023. Mauve scores for generative models: Theory and practice. *Journal of Machine Learning Research*, 24(356):1–92.

Krishna Pillutla, Swabha Swayamdipta, Rowan Zellers, John Thickstun, Sean Welleck, Yejin Choi, and Zaid Harchaoui. 2021. Mauve: Measuring the gap between neural text and human text using divergence frontiers. *Advances in Neural Information Processing Systems*, 34:4816–4828.

Matthew Renze and Erhan Guven. 2024. The effect of sampling temperature on problem solving in large language models. *arXiv preprint arXiv:2402.05201*.

V Sanh. 2019. Distilbert, a distilled version of bert: Smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Shibani Santurkar, Esin Durmus, Faisal Ladhak, Cinoo Lee, Percy Liang, and Tatsunori Hashimoto. 2023. Whose opinions do language models reflect? In *International Conference on Machine Learning*, pages 29971–30004. PMLR.

Nino Scherrer, Claudia Shi, Amir Feder, and David Blei. 2024. Evaluating the moral beliefs encoded in llms. *Advances in Neural Information Processing Systems*, 36.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. 2017. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *arXiv preprint arXiv:1701.06538*.

Gabriel Simmons. 2022. Moral mimicry: Large language models produce moral rationalizations tailored to political identity. arxiv. *arXiv preprint arXiv:2209.12106*.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA. Association for Computational Linguistics.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. Mpnet: Masked and permuted pre-training for language understanding. *Advances in neural information processing systems*, 33:16857–16867.

Taylor Sorensen, Jared Moore, Jillian Fisher, Mitchell Gordon, Niloofar Mireshghallah, Christopher Michael Rytting, Andre Ye, Liwei Jiang, Ximing Lu, Nouha Dziri, et al. 2024. A roadmap to pluralistic alignment. *arXiv preprint arXiv:2402.05070*.

Chenkai Sun, Ke Yang, Revanth Gangi Reddy, Yi R Fung, Hou Pong Chan, ChengXiang Zhai, and Heng Ji. 2024. Persona-db: Efficient large language model personalization for response prediction with collaborative data refinement. *arXiv preprint arXiv:2402.11060*.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, et al. 2024. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*.

Petter Törnberg. 2023. Chatgpt-4 outperforms experts and crowd workers in annotating political twitter messages with zero-shot learning. *arXiv preprint arXiv:2304.06588*.

Yu-Min Tseng, Yu-Chao Huang, Teng-Yun Hsiao, Yu-Ching Hsu, Jia-Yin Foo, Chao-Wei Huang, and Yun-Nung Chen. 2024. Two tales of persona in llms: A survey of role-playing and personalization. *arXiv preprint arXiv:2406.01171*.

Zhenhailong Wang, Shaoguang Mao, Wenshan Wu, Tao Ge, Furu Wei, and Heng Ji. 2023. Unleashing the emergent cognitive synergy in large language models: A task-solving agent through multi-persona self-collaboration. *arXiv preprint arXiv:2307.05300*.

Jingjing Xu, Xuancheng Ren, Junyang Lin, and Xu Sun. 2018. DP-GAN: diversity-promoting generative adversarial network for generating informative and diversified text. *arXiv preprint arXiv:1802.01345*.

Jiacheng Ye, Jiahui Gao, Jiangtao Feng, Zhiyong Wu, Tao Yu, and Lingpeng Kong. 2022a. Progen: Progressive zero-shot dataset generation via in-context feedback. *arXiv preprint arXiv:2210.12329*.

Jiacheng Ye, Jiahui Gao, Qintong Li, Hang Xu, Jiangtao Feng, Zhiyong Wu, Tao Yu, and Lingpeng Kong. 2022b. Zerogen: Efficient zero-shot learning via dataset generation. *arXiv preprint arXiv:2202.07922*.

Xiaoyan Yu, Tongxu Luo, Yifan Wei, Fangyu Lei, Yiming Huang, Peng Hao, and Liehuang Zhu. 2024a. Neeko: Leveraging dynamic lora for efficient multi-character role-playing agent. *arXiv preprint arXiv:2402.13717*.

Yue Yu, Yuchen Zhuang, Jieyu Zhang, Yu Meng, Alexander J Ratner, Ranjay Krishna, Jiaming Shen, and Chao Zhang. 2024b. Large language model as attributed training data generator: A tale of diversity and bias. *Advances in Neural Information Processing Systems*, 36.

Yue Yu, Yuchen Zhuang, Rongzhi Zhang, Yu Meng, Jiaming Shen, and Chao Zhang. 2023. Regen: Zero-shot text classification via training data generation with progressive dense retrieval. *arXiv preprint arXiv:2305.10703*.

Xiang Yue, Huseyin A Inan, Xuechen Li, Girish Kumar, Julia McAnallen, Hoda Shajari, Huan Sun, David Levitan, and Robert Sim. 2022. Synthetic text generation with differential privacy: A simple and practical recipe. *arXiv preprint arXiv:2210.14348*.

Shimao Zhang, Yu Bao, and Shujian Huang. 2024. EDT: Improving large language models' generation by entropy-based dynamic temperature sampling. *arXiv preprint arXiv:2403.14541*.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. *Advances in neural information processing systems*, 28.

Siyan Zhao, John Dang, and Aditya Grover. 2023. Group preference optimization: Few-shot alignment of large language models. *arXiv preprint arXiv:2310.11523*.

Tianyuan Zou, Yang Liu, Peng Li, Jianqing Zhang, Jingjing Liu, and Ya-Qin Zhang. 2024. Fusegen: Plm fusion for data-generation based zero-shot learning. *arXiv preprint arXiv:2406.12527*.

A Societal Impact

While generating human-like datasets opens up many exciting possibilities for augmenting blind spots in data that may otherwise have led to limitations preventing a complete understanding of important information, synthetic data can still lack specific nuances, diversity, and edge cases present in human-derived data. This is especially prevalent in population sub-groups that remain underrepresented, causing a lack of labeled data in specific languages that affect the authenticity and coverage of over two billion individuals worldwide (Joshi et al., 2020). Our focus on augmenting human records instead of replacing them is notwithstanding the potential of synthetic data; it is imperative that we continue supporting data collection obtained directly from people in under-served communities and domains and, whenever possible, make those data freely available for empirical and experimental research. Beyond the behavioral science and data generation for training ML models, data generated using MoP could be leveraged across a broad range of applications, such as informatics-based health claims, market intelligence and user-preference modeling, and free-form text writing – where the goal is not only to produce accurate or factually-correct answers but also to simulate a population of interest otherwise out of reach.

B Additional Experiment Settings

Datasets. Below are the detailed descriptions of four datasets used in this paper.

- *AGNews* (Zhang et al., 2015): This dataset contains 103600 news articles crawled from AG News. Each article is categorized into political news, sports news, business news, and technology news.
- *Yelp Reviews* (Zhang et al., 2015): This dataset contains restaurant reviews from the Yelp platform. Each review is labeled with a positive or negative sentiment label. This popular dataset is used for sentiment analysis and opinion mining.
- *SST-2 (Stanford Sentiment Treebank)* (Socher et al., 2013): The dataset consists of 11,855 individual sentences extracted from movie reviews to analyze the compositional effects of sentiment in language. Each sentence is annotated with a binary label indicating whether the review is positive or negative.
- *IMDB Reviews* (Maas et al., 2011): The dataset consists of 50,000 movie reviews crawled from IMDB with 25,000 samples for positive classes and 25,000 samples for negative classes.

Baselines. We provide descriptions for baselines used in our paper.

- **ZeroGen** (Ye et al., 2022b) generates synthetic samples by feeding simple class-conditioned prompts to language models. We reuse the same prompts described in (Yu et al., 2024b).
- **Attrprompt** (Yu et al., 2024b) further increases the diversity of synthetic samples by utilizing a set of sample attributes generated by Chat-GPT. For example, to generate synthetic examples for AgNews, Attrprompt also inputs randomized values for 'writing style' or 'location' to the prompt in addition to class information. We refer the readers to their papers for the full description of the used attributes for each dataset. In our experiment, we use the same attribute set released by the authors in their repository [4].
- **ProGen (Ye et al., 2022a):** proposes a framework for zero-shot dataset generation that iteratively improves the quality of synthetic datasets. It uses feedback from a task-specific model, guided by a noise-tolerant influence function, to identify high-quality samples and incorporate them as in-context examples for subsequent data generation.
- **PICLE (Choi and Li, 2024):** is a framework designed to elicit specific target personas from large language models (LLMs) using an in-context learning (ICL) approach. It employs a novel likelihood-ratio-based selection mechanism to identify and prepend the most informative examples from a persona-specific pool, guiding the model to align with the desired persona. By leveraging Bayesian inference, PICLe modifies the LLM's output distribution to better reflect the target persona, achieving improved persona elicitation compared to baseline methods.

Metrics. Measuring the alignment of the synthetic data with the golden dataset can be challenging due to the discrete nature of text sequences. We follow (Pillutla et al., 2021; Yu et al., 2024b) to

[4] https://github.com/yueyu1030/AttrPrompt/tree/main

evaluate the alignment and diversity of the simulated data in the embedding space. Throughout, we use a pre-trained sentence encoder 'all-mpnet-base-v2' [5] (Song et al., 2020) as a text encoder to map generated sentences into a common vector space. We then report three following metrics:

- FID (Fréchet Inception Distance) (Heusel et al., 2017) is a widely-used metric in computer vision literature to measure the fidelity of synthetic images with respect to real images. Following the approach in (Yue et al., 2022), we adapt FID for text generation. We first calculate the mean and covariance matrices of sentence embeddings for both the golden test set and synthetic datasets, then use the Fréchet distance to compute the FID between them.

- The MAUVE metric (Pillutla et al., 2021, 2023) evaluates the similarity between two text distributions by comparing their divergence frontiers. It begins by embedding text into vectors and clustering them to create histograms of cluster assignments. A divergence curve is then constructed from these histograms, and the area under the curve quantifies the difference between the synthetic and golden data distributions. In our experiments, we use the original implementation of MAUVE [6]. As suggested by the authors, we set the number of clusters to 500 and the scaling parameter to 1.

- The Kullback-Leibler Cosine (KL Cosine) metric evaluates the divergence between the pairwise cosine similarity distributions of a method and the ground truth. First, the pairwise cosine similarities are computed for the method and the ground truth. Then, the Kullback-Leibler divergence between their normalized histograms is calculated as:

$$D_{KL}(P\|Q) = \sum_i P(i) \log \frac{P(i)}{Q(i)} \quad (6)$$

where $P(i)$ represents the histogram of pairwise cosine similarities for the method, and $Q(i)$ represents the histogram for the ground truth.

C Additional ablation studies

We conducted an additional experiment to evaluate whether our framework is sensitive to the format of ICL prompts. Specifically, we modified the original ICL prompt ('You have written the following news blurb: [example]') to two alternative formats: ICL Template 1 and ICL Template 2. The results in the table below indicate that changes in the ICL format have minimal impact on performance in our setting.

```
ICL Template 1:
Example news blurb: [example]

ICL Template 2:
The following news blurb
is given as an example: [example]
```

These results suggest that our framework's performance is not overly sensitive to the specific format of ICL prompts.

Table 6: Train test split and the number of classes in each dataset.

	AgNews	Yelp	SST-2	IMDB
Train set	96000	141076	67349	14006
Test set	7600	35323	1821	14056
Number of classes	4	2	2	2

Table 7: Comparison of metrics across different ICL prompts and templates.

	FID	Mauve	Cosine Similarity	KL Cosine
Original ICL Prompt	0.951	0.871	0.108	0.069
ICL Template 1	0.955	0.883	0.110	0.074
ICL Template 2	0.955	0.881	0.108	0.069

D MoP Prompts

In this section, we present the prompts used throughout our experiments.

1. **Synthesizing New Samples.** The prompts in Fig. 5, Fig. 6, Fig. 7, and Fig. 8 are designed to synthesize new samples based on provided examples. These prompts guide the language model to generate new samples in line with the provided context and example, while maintaining consistency with the style and tone of the given input.

2. **Generating new personas.** We use the prompts in Fig. 9, Fig. 10, Fig. 11, and Fig. 12 to generate a new persona based on examples that reflect the characteristics of that persona.

3. **Classifying personas.** We use the prompts shown in Fig. 13 to classify a persona based

[5] sentence-transformers/all-mpnet-base-v2
[6] https://krishnap25.github.io/mauve/

on its description. The prompt provides a persona description of a reporter, and the task is to identify the reporter's specialization from four predefined categories: world news, sports news, business news, or sci/tech news.

4. **Persona Mixing Prompt.** The prompt in Fig. 14 is designed to synthesize a unified persona description by combining multiple persona descriptions with sample news blurbs written by the reporter. It integrates the thematic focus, stylistic tendencies, and primary interests from the inputs to generate a concise and cohesive profile of the reporter.

Figure 5: MoP Prompt for Agnews

```
<|begin_of_text|><|start_header_id|>system<|end_header_id|>\n\nYou embody the persona in the
    description to complete the tasks.<|eot_id|><|start_header_id|>user<|end_header_id|>\n\n
\n\nYou have written the following news blurb: {example}
\n\nPlease write a short news blurb similar to the above blurb.<|eot_id|>
    <|start_header_id|>assistant<|end_header_id|>\n\n
```

Figure 6: MoP Prompt for Yelp

```
<|begin_of_text|><|start_header_id|>system<|end_header_id|>\n\nYou embody the persona in the
    description to complete the tasks.<|eot_id|><|start_header_id|>user<|end_header_id|>\n\n
\n\nYour review for the restaurant {context}: {example}
\n\nPlease write a short review for the restaurant {context}, similar to the above review:<|eot_id|>
    <|start_header_id|>assistant<|end_header_id|>\n\n
```

Figure 7: MoP Prompt for SST-2

```
<|begin_of_text|><|start_header_id|>system<|end_header_id|>\n\nYou embody the persona in the
    description to complete the tasks.<|eot_id|><|start_header_id|>user<|end_header_id|>\n\n
\n\nYou have written the following review: {example}
\n\nPlease write a review sentence, similar to the above
    review:<|eot_id|><|start_header_id|>assistant<|end_header_id|>
    \n\n<|start_header_id|>assistant<|end_header_id|>\n\n
```

Figure 8: MoP Prompt for IMDB

```
<|begin_of_text|><|start_header_id|>system<|end_header_id|>\n\nYou embody the persona in the
    description to complete the tasks.<|eot_id|><|start_header_id|>user<|end_header_id|>\n\n
\n\nYou have written the following review for the movie {context}: {example}
\n\nPlease write a review for the movie {context}, similar to the above
    review:<|eot_id|><|start_header_id|>assistant<|end_header_id|>\n\n
```

Figure 9: MoP Prompt to generate persona for Agnews

```
<|begin_of_text|><|start_header_id|>system<|end_header_id|>

You are a helpful AI assistant<|eot_id|><|start_header_id|>user<|end_header_id|>

Given a list of news blurbs written by a reporter, construct a concise persona description that
    reflects the reporter's primary focus, style, and thematic interests.

Example persona descriptions:
You are a sports reporter, specializing in baseball news, with a focus on the Major League Baseball
    (MLB) playoffs and postseason games.

List of news blurbs:
{examples}

Generate a short persona description that synthesizes their focus, preferences, and stylistic
    tendencies into a single cohesive
    statement.<|eot_id|><|start_header_id|>assistant<|end_header_id|>

The short persona is:
```

Figure 10: MoP Prompt to generate persona for Yelp

```
<|begin_of_text|><|start_header_id|>system<|end_header_id|>

You are a helpful AI assistant<|eot_id|><|start_header_id|>user<|end_header_id|>

Given a list of restaurant review written by a customer, construct a concise persona description
    that reflects the customer's preferences, writing style, and interests.

Examples of Persona Descriptions:
You are a food critic, specializing in fine dining and gourmet cuisine with a focus on presentation
    and taste.
You are a casual diner who enjoys comfort food and writes personal and informal reviews.

List of reviews:
{examples}

Generate a short persona description that synthesizes their interests, preferences, and writing
    stylistic tendencies into a single cohesive
    statement.<|eot_id|><|start_header_id|>assistant<|end_header_id|>

The short persona is:
```

Figure 11: MoP Prompt to generate persona for SST-2

```
<|begin_of_text|><|start_header_id|>system<|end_header_id|>

You are a helpful AI assistant<|eot_id|><|start_header_id|>user<|end_header_id|>

Given a list of movie review written by a viewer, construct a concise persona description that
    reflects the review's preferences, writing style, and thematic interests.

Examples of Persona Descriptions:
You are a movie critic, specializing in horror movies and independent films with a focus on
    cinematography and storytelling.
You are a casual viewer who enjoys action-packed films and writes personal and informal reviews.

List of reviews:
{examples}

Generate a short persona description that synthesizes their focus, preferences, and stylistic
    tendencies into a single cohesive
    statement.<|eot_id|><|start_header_id|>assistant<|end_header_id|>

The short persona is:
```

Figure 12: MoP Prompt to generate persona for IMDB

```
<|begin_of_text|><|start_header_id|>system<|end_header_id|>

You are a helpful AI assistant<|eot_id|><|start_header_id|>user<|end_header_id|>

Given a list of movie review written by a viewer, construct a concise persona description that
    reflects the review's preferences, writing style, and thematic interests.

Examples of Persona Descriptions:
You are a movie critic, specializing in horror movies and independent films with a focus on
    cinematography and storytelling.
You are a casual viewer who enjoys action-packed films and writes personal and informal reviews.

List of reviews:
{examples}

Generate a short persona description that synthesizes their focus, preferences, and stylistic
    tendencies into a single cohesive
    statement.<|eot_id|><|start_header_id|>assistant<|end_header_id|>

The short persona is:
```

Figure 13: MoP Prompt to classify persona for AgNews

```
<|begin_of_text|><|start_header_id|>system<|end_header_id|>

You are a helpful AI assistant<|eot_id|><|start_header_id|>user<|end_header_id|>

Given a persona description of a reporter, choose the specialization of the personas: world news,
    sports news, business news, sci/tech news.

Persona description: You are a technology reporter, specializing in the intersection of hardware and
    software, with a focus on the PC industry, Linux, and major tech companies like Microsoft,
    Apple, and IBM.
Options:
A. world news
B. sports news
C. business news
D. sci/tech news
Answer: D

Persona description: {}
Options:
A. world news
B. sports news
C. business news
D. sci/tech news
Answer: <|eot_id|><|start_header_id|>assistant<|end_header_id|>
```

Figure 14: MoP Prompt to mix personas for AgNews

```
<|begin_of_text|><|start_header_id|>system<|end_header_id|>

You are a highly capable and insightful AI
    assistant<|eot_id|><|start_header_id|>user<|end_header_id|>

Given a list of persona descriptions of a reporter and a list of news blurbs he/she has written,
    construct a concise persona description that reflects the reporter's primary focus, style, and
    thematic interests.

Example persona descriptions:
You are a sports reporter, specializing in baseball news, with a focus on the Major League Baseball
    (MLB) playoffs and postseason games.

Inputs:
1. List of persona descriptions: A list of general persona characteristics previously associated
    with the reporter.
2. List of news blurbs: A selection of sample news blurbs authored by the reporter, which reveal
    their tone, thematic focus, and writing style.

Using the inputs, generate a short persona description that synthesizes their focus, preferences,
    and stylistic tendencies into a single cohesive statement.

List of persona descriptions:
{personas}

List of news blurbs:
{examples}

Please provide the short persona description.<|eot_id|><|start_header_id|>assistant<|end_header_id|>

The short persona is:
```