



Universidad Politécnica
de Madrid

Escuela Técnica Superior de
Ingenieros Informáticos



Máster Universitario en Inteligencia Artificial

Trabajo Fin de Máster

Simulating Society: Leveraging Large Language Models as Citizen Agents to Study Urban Behavior

Author: José Miguel Nicolás García

Tutor: Pablo Calleja Ibáñez

MIT Supervisors: Luis Alberto Alonso Pastor & Adrián Mora Carrero

Madrid, July - 2025

Este Trabajo Fin de Máster se ha depositado en la ETSI Informáticos de la Universidad Politécnica de Madrid para su defensa.

Trabajo Fin de Máster
Máster Universitario en Inteligencia Artificial

Título: Simulating Society: Leveraging Large Language Models as Citizen Agents to Study Urban Behavior

July - 2025

Autor(a): José Miguel Nicolás García
Tutor(a): Pablo Calleja Ibáñez
Departamento de Inteligencia Artificial
ETSI Informáticos
Universidad Politécnica de Madrid

*A la memoria de mis dos abuelas,
fallecidas durante el transcurso de este máster.*

*Computing is not about computers anymore. It is about living.
Nicholas Negroponte, MIT Media Lab Founder*

Acknowledgements

First and foremost, I want to thank my parents for always supporting my decisions, values, and aspirations, whether they seemed attainable or not. And to my brother, thank you for always being there, with a reassuring word or a pat on the back when I needed it most.

My deepest gratitude to Kent for welcoming me into his group like family, even inviting us into his home, and to every member of the City Science Group for their guidance, collaboration, and companionship throughout these nine incredible months.

Thank you, Luis, for being a mentor from day one, not just as a researcher, but as a human being.

To the entire Massachusetts Institute of Technology (MIT) and Media Lab community: thank you for inspiring me, challenging me, and opening the door to conversations that will shape the future. And to the institution itself, thank you for embracing me as one of your own and giving me the chance to live out a childhood dream.

Thank you to Pablo and Adrián for closely following this thesis and offering thoughtful guidance week after week.

To all the visiting students from MIT and Harvard who shared this unforgettable journey with me, thank you for your friendship and for making this one of the most meaningful chapters of my life.

To the LOST Theater group: thank you for encouraging me to step on stage and perform a three-hour play in a foreign language at MIT. I will never forget it.

And finally, thank you, Lucía, for walking this path by my side. Boston would not have been the same without you.

Abstract

Large Language Models (LLMs) have revolutionized natural language generation, enabling machines to produce coherent, contextual, and expressive text. These advances open up new possibilities in domains such as agent-based modeling, synthetic data generation, and social simulation, fields that require not only statistical accuracy but also behavioral realism and psychological nuance. In the context of urban planning, simulating populations that reflect real human complexity has become a critical step toward designing more adaptive, inclusive, and human-centered cities.

This thesis aims to explore the potential of LLMs to generate synthetic human agents that are demographically realistic and psychologically expressive. The objective is to develop and evaluate a methodological pipeline capable of producing coherent profiles aligned with population statistics and controllable personality traits, providing a foundation for their integration into humanized Agent Based Models (h-ABMs).

To this end, we designed a two-phase system. The first phase focuses on generating synthetic profiles based on real demographic data from Cambridge (Massachusetts), using models such as LLaMA, Qwen, Dolphin, and Mistral. These profiles were evaluated in terms of narrative richness, lexical diversity, semantic coherence, and cultural bias. In the second phase, we applied fine-tuning techniques, specifically Low-rank Decomposition (LoRA) combined with Direct Policy Optimization (DPO) to condition model outputs on Big Five personality vectors.

Our results demonstrate that LLMs can generate synthetic populations with remarkable diversity, internal coherence, and emotional realism. However, we also identified persistent biases, both cultural and emotional, that limit the representation of negative traits and complex personalities. The fine-tuned models showed promising control over traits like high Openness and Extraversion, while others such as Neuroticism and low Agreeableness remained more resistant to modulation. This highlights the need for more nuanced datasets and ethical considerations when modeling human identity. Overall, this work offers a reproducible, extensible framework for the controlled generation of synthetic human agents, with applications in simulation, behavioral research, education, and beyond.

Resumen

Los *Large Language Models (LLMs)* han revolucionado la generación de lenguaje natural, permitiendo producir textos coherentes, contextuales y expresivos. Estos avances abren nuevas posibilidades en ámbitos como la modelización basada en agentes, la generación de datos sintéticos y la simulación social, campos que requieren no solo precisión estadística, sino también realismo conductual y matices psicológicos. En el contexto de la planificación urbana, simular poblaciones que reflejen la complejidad humana real se ha convertido en un paso fundamental para diseñar ciudades más adaptativas, inclusivas y centradas en el ser humano.

Este trabajo pretende explorar el potencial de los LLMs para generar agentes humanos sintéticos que sean demográficamente realistas y psicológicamente expresivos. El objetivo es desarrollar y evaluar un pipeline metodológico capaz de producir perfiles coherentes alineados con estadísticas poblacionales y rasgos de personalidad controlables, proporcionando una base para su integración en el framework *humanized Agent Based Models (h-ABMs)*.

Para ello, diseñamos un sistema en dos fases. La primera fase se centra en la generación de perfiles sintéticos basados en datos demográficos reales de Cambridge (Massachusetts), utilizando modelos como LLaMA, Qwen, Dolphin y Mistral. Estos perfiles se evaluaron en términos de riqueza narrativa, diversidad léxica, coherencia semántica y sesgo cultural. En la segunda fase, aplicamos técnicas de ajuste fino, concretamente *Low-rank Decomposition (LoRA)* combinada con *Direct Policy Optimization (DPO)* para condicionar las salidas del modelo a los vectores de personalidad de la teoría de Cinco Grandes Rasgos.

Nuestros resultados demuestran que los LLMs pueden generar poblaciones sintéticas con una notable diversidad, coherencia interna y realismo emocional. Sin embargo, también identificamos sesgos, tanto culturales como emocionales, que limitan la representación de rasgos negativos y personalidades complejas. Los modelos afinados mostraron un control prometedor sobre rasgos como alta Apertura y Extroversión, mientras que otros como el Neuroticismo y la baja Amabilidad permanecieron más resistentes a la modulación. Esto pone de manifiesto la necesidad de contar con conjuntos de datos más precisos y consideraciones éticas a la hora de modelar la identidad humana. En general, este trabajo ofrece un marco reproducible y extensible para la generación controlada de agentes humanos sintéticos, con aplicaciones en simulación, investigación del comportamiento, educación y otros campos.

Contents

Acronyms	xv
1 Introduction	1
1.1 Objectives	2
1.2 Document Structure	3
1.3 MIT Media Lab City Science Group	3
2 State of the Art	7
2.1 Introduction to Language Representations	7
2.2 Introduction to Large Language Models	8
2.2.1 Historical Evolution of Large Language Models	8
2.2.2 Training Large Language Models	11
2.2.3 Emergent Capabilities and Current Limitations of LLMs	12
2.3 Optimizing Large Language Models	13
2.3.1 Prompt Engineering	14
2.3.2 Retrieval-Augmented Generation (RAG)	15
2.3.3 Fine-tuning of Large Language Models	16
2.3.4 Parameter-Efficient Fine-tuning	17
2.3.5 Alignment Algorithms	20
2.4 LLM-Based Autonomous Agents	21
2.4.1 Core Components	22
2.4.2 Evaluation Strategies	23
2.4.3 Applications	23
2.4.4 Challenges and Open Problems	24
2.5 Humanized Agents Societies	24
2.5.1 Environment for Agent Society	24
2.5.2 Behavior and Personality	24
2.5.3 Society Simulation with LLM-based Agents	25
2.6 State of the Art in Urban Simulation at the MIT City Science Group	26
3 Design and Development	29
3.1 Problem Analysis: Optimization of LLMs for Human-Centric Agents	29
3.1.1 Scientific modeling of personality: Big Five Personality Traits	30
3.2 Phase I: Context Optimization	31
3.2.1 Synthetic Population Generation	31
3.2.2 LLM-Guided Enrichment	33
3.2.3 Complete Profile Architecture	34
3.3 Phase II: Fine-tuning	36

3.3.1	Personality Evaluation and Selection	37
3.3.2	Dataset Design for DPO Training	41
3.3.3	Training Process	45
4	Experiments and Results	49
4.1	Evaluation of Phase I: Profile Generation	49
4.1.1	Similarity	49
4.1.2	Demographic Variables Analysis	51
4.1.3	LLM-generated Variables	56
4.2	Evaluation of Phase II: Fine Tuning	60
5	Final Conclusions and Future Work	65
5.1	Conclusions	65
5.2	Future Work	66
References		81
Appendices		82
A	Transformer Architecture	85
B	Profiling Prompts	89
B.1	Task 1	89
B.2	Task 2	89
B.3	Task 3	90
B.4	Task 4	90
B.5	Task 5	90
B.6	Task 6	91
B.7	Task 7	91
B.8	Task 8	91
C	JSON Profile Example	93
C.1	Input Example	93
C.2	Output example	94
D	Dataset Generation	97
D.1	Topics	97
D.2	Answering	100
D.2.1	Answering Prompt	100
D.2.2	Answer example	102
D.2.3	Simplified Dataset Entry	103
D.3	Big Five Evaluation	103
D.3.1	Evaluation Prompt	103
D.3.2	Evaluation Output Example	105

List of Figures

1.1	MIT	4
1.2	MIT Media Lab.	4
1.3	City Science Group	5
1.4	City Science Group logo	5
2.1	History and development of Language Models	9
2.2	ChatGPT training steps	12
2.3	LLM optimization	14
2.4	Retrieval-Augmented Generation pipeline	15
2.5	Retrieval-Augmented Generation paradigms	16
2.6	Taxonomy of Parameter-Efficient Fine-tuning (PEFT) methods	17
2.7	Main PEFT methods	19
2.8	Proximal Policy Optimization (PPO) vs DPO	21
2.9	Agent development timeline	22
2.10	Core components of an Agent	23
2.11	Generative Agents: Interactive Simulacra of Human Behavior	25
2.12	Reinforcement Learning with Big Five Personality Traits	25
2.13	Social interaction between agents	26
2.14	Travel Agent	27
2.15	h-ABM framework	27
2.16	h-ABM simulation	28
3.1	Big 5 personality traits	30
3.2	LLM-based personality profiling pipeline.	33
3.3	Myers–Briggs Type Indicator (MBTI) 16 personalities	40
3.4	Dataset generation pipeline for fine-tuning via DPO	41
4.1	Population pyramids	52
4.2	Country of origin	53
4.3	Work by industry	53
4.4	Education	54
4.5	Emotions comparison	58
4.6	Sentiment comparison	58
4.7	Comparison of emotions in fine-tuned models	63
A.1	Transformer structure	86

List of Tables

2.1	Comparison of autoregressive and autoencoder models	11
2.2	Alignment of LLMs capabilities with agent requirements	21
3.1	Comparison among the main personality models	31
3.2	Mapping between numerical trait values and qualitative interpretations	39
3.3	Target Big Five personality vectors used for conditioned generation	39
3.4	Technologies and frameworks used in the training pipeline	46
4.1	Similarity metrics across LLM-generated profiles	50
4.2	Mean absolute error in categoric demographics	55
4.3	Top 20 most frequent words in general descriptions	56
4.4	Top 20 most frequent words in big five descriptions	57
4.5	Diversity and frequency analysis of LLM generated variables	59
4.6	Evaluation metrics for each model and configuration.	61
4.7	Big Five average values	62
4.8	Top 20 most frequent words across four personality types	64

Acronyms

ABM Agent Based Model

AI Artificial Intelligence

API Application Programming Interface

CoT Chain-of-Thought

CSV Comma-Separated Values

DPO Direct Policy Optimization

FFN Feed-Forward Network

h-ABM humanized Agent Based Model

JSON JavaScript Object Notation

LLM Large Language Model

LoRA Low-rank Decomposition

LSTM Long Short-Term Memory

MAE Mean Absolute Error

MBTI Myers–Briggs Type Indicator

MIT Massachusetts Institute of Technology

NLP Natural Language Processing

PEFT Parameter-Efficient Fine-tuning

PPO Proximal Policy Optimization

RAG Retrieval-Augmented Generation

RLHF Reinforcement Learning from Human Feedback

RNN Recurrent Neural Networks

SFT Supervised Fine-tuning

TRL Transformers Reinforcement Learning

VRAM Video Random Access Memory

Chapter 1

Introduction

Currently, the use of computational models to simulate populations has proven to be a necessary tool for the analysis of social dynamics, evaluation of public policies and study of complex scenarios [1, 2, 3, 4, 5, 6, 7, 8, 9]. In this field, Agent Based Models (ABMs) need to evolve [10] to incorporate not only simple rules and stereotyped behaviors, but also adaptive behaviors, social interaction and contextualized decision-making to capture the richness and heterogeneity of human behavior.

This has led to the emergence of the humanized h-ABM framework [11] that frames this thesis and seeks to endow agents with psychological attributes, internal motivations and plausible life trajectories. This is especially relevant in the context of urban planning, where understanding and predicting how people interact with their environment becomes essential to design fairer, more resilient, and sustainable cities.

In this context, the MIT Media Lab’s City Science research group ¹ proposes that the cities of the future should not only be built from the infrastructure, but also from a deep understanding of the people who inhabit them.

In recent years, advances in language modeling (see Section 2.2) have radically transformed the capabilities of automatic text generation, opening up new opportunities to simulate human agents with a high degree of coherence and expressiveness, really important features in social simulation, urban planning, the design of autonomous agents or the generation of synthetic data for analytical purposes. In these domains, the availability of artificial populations that not only reproduce demographic statistics, but also reflect the diversity, emotional complexity and behavioral variability of real human beings, represents a growing need. However, current solutions often rely on random or simplified profiles, and lack mechanisms to accurately model the personality, emotionality or life goals of the simulated individuals.

On the other hand, although preference-based fine-tuning techniques (such as DPO) have emerged, their application to the explicit control of psychological traits in LLMs is still in its infancy. This represents a unique opportunity to explore new ways of personalizing textual generation from personality vectors, aligning model output with theoretical representations such as the Big Five personality traits.

“Humanizing” simulation is not only an academic goal, but a necessary step to make urban design, mobility or education tools more sensitive to human diversity. Through the use of

¹<https://www.media.mit.edu/groups/city-science/overview/>

LLMs, the possibility of generating realistic, emotionally nuanced and culturally contextualized synthetic agents capable of participating in richer and more predictive simulations opens up.

This thesis is therefore a contribution to the intersection between generative Artificial Intelligence (AI) and social modeling. **It aims to demonstrate that LLMs can be used not only as text generators, but also as tools to build complex, assessable and controllable synthetic agents.**

In this context, this thesis is proposed as an opportunity to investigate to what extent it is possible to generate synthetic human profiles that combine narrative coherence, psychological diversity and demographic realism, using advanced LLM architectures, guided design techniques and fine-tuning methods. The ultimate goal is to move towards the generation of more useful and realistic synthetic agents, contributing to the development of richer, ethically informed and human-centered simulation and analysis tools.

1.1 Objectives

This project began with a series of training and technical objectives focused on acquiring an in-depth understanding of how language models work and their applicability in specific contexts. These included:

- Study in detail the theoretical and architectural foundations of current language models.
- Understanding and applying the process of optimizing a language model for a specific use case.
- Becoming familiar with the main tools in the ecosystem developed in recent years, including libraries such as Hugging Face transformers and datasets, PEFT Library, BitsAndBytes, Accelerate and the Ollama Application Programming Interface (API).
- Explore scientific frameworks for human personality representation, in order to humanize artificial agents from a solid psychological foundation.
- Apply all this knowledge to “Humanize” agent-based models in the context of the City Science Group, generating synthetic populations with realistic human profiles for integration into urban simulations.

After achieving the initial objectives, the project evolved organically to address more complex questions and bring greater value to the research work:

- Explore the potential of LLMs to generate coherent, diverse, and psychologically rich synthetic human profiles for social simulations.
- Compare the semantic and narrative behavior of different LLMs (LLaMA, Qwen, Mistral and Dolphin) evaluating their demographic fidelity, lexical diversity and discursive style.
- Detect and analyze structural and cultural biases present in the models, observing how they affect the representation of personality traits, social values and emotions.
- Evaluate the psychological realism of the generated profiles.
- Experiment with LLM fine-tuning techniques to create customized models and match their narrative behavior to specific personality vectors.

1.2 Document Structure

This document is organized into the following sections:

- **Introduction:** Provides an overview of the thesis, outlining its motivation, goals, and overall structure.
- **State of the Art:** Introduces essential concepts of LLMs to support understanding of the subsequent chapters and presents the current LLMs workflow, Agents and Society Simulation.
- **Design and Development:** Describes both the design considerations and the implementation process.
- **Experiments and Results:** Details the experiments conducted and interprets the findings of this thesis.
- **Final Conclusions and Future Work:** Recaps the objectives accomplished, reflects on key learnings, and suggests potential directions for future investigation.
- **Appendices:** Includes supplementary materials such as result tables, JavaScript Object Notations (JSONs) output examples, among others.

You can also consult complementary resources including datasets, scripts and complete results in the GitHub repository of this project².

1.3 MIT Media Lab City Science Group

This section describes the City Science Group, where I had the opportunity to conduct research and perform my Master's Thesis over the past year.

The MIT (see Figure 1.1) is internationally known for its research and focus on working across disciplines to address complex social problems. Within MIT, the Media Lab (see Figure 1.2) stands out as a hub of creativity with experts in diverse fields such as design, engineering, social sciences, and art to develop transformative projects. The Media Lab promotes "anti-disciplinary" research, encouraging projects that cross traditional academic lines.

In this context, the City Science Group, led by Kent Larson, is a multidisciplinary research team dedicated to reimagining urban environments (see Figures 1.3 and 1.4). The group explores innovative approaches to urban living beyond traditional architecture or planning, combining technology, social sciences, and innovation to help cities become more resilient and equitable.

The group aims to transform cities by improving their environmental performance, strengthening their social fabric, and stimulating local economies. Rather than focusing exclusively on top-down infrastructure upgrades or incremental policy changes, the group develops holistic models that address the complex, interdependent systems shaping urban life. One of their research highlights is the Unified Urban Performance Model, a systems-level framework to evaluate and simulate the effects of various urban interventions.

The City Science Group therefore advocates for deep, integrated interventions, including:

- Synchronizing housing with local job opportunities.
- Providing essential amenities (healthcare, education, recreation) within walking distance.

²<https://github.com/CityScope/personality-driven-synthetic-populations>

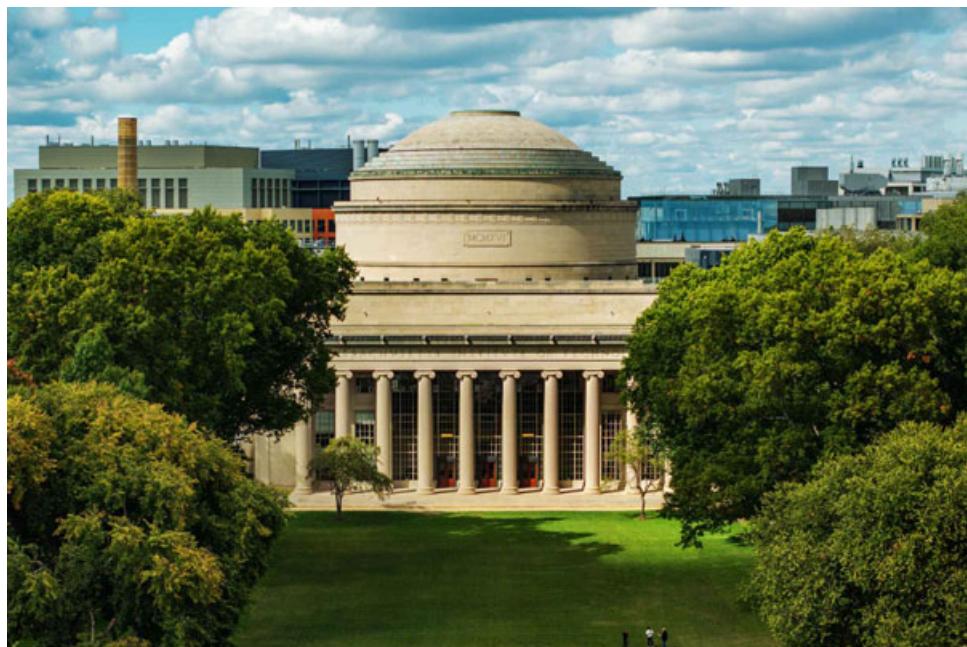


Figure 1.1: MIT. Source: MIT Website.



Figure 1.2: MIT Media Lab. Source: MIT Media Lab website.

Introduction



Figure 1.3: City Science Group. Source: MIT Media Lab website.



Figure 1.4: City Science Group logo. Source: MIT Media Lab website.

- Developing community-scale autonomous vehicles to enhance shared mobility.
- Designing transformable living spaces that dynamically adapt to residents' needs over the course of the day.
- Fostering local food production through plant-based diets, synthetic meat, and hyper-local supply chains.
- Exploring high-density, modular energy systems, and preparing urban areas for future fusion-based energy.

A key part of their work is building **simulation and support tools** so that policymakers and community leaders can make decisions based on data. These tools help identify the most promising interventions and prioritize them according to both local needs and global challenges.

Their research embodies this holistic, interdisciplinary philosophy, seeking to empower communities to lead their own transition toward a sustainable future.

Chapter 2

State of the Art

The second chapter of this thesis presents the state of the art of humanized LLMs agents. To do so, we analyze the current techniques for working with LLMs, their core technology, the concept of agents, and the work of the MIT City Science group, where this research was carried out.

2.1 Introduction to Language Representations

The first fundamental concept to understand a Language Model is to comprehend how the language is computationally represented. Natural language understanding in neural models begins with transforming raw text into numerical representations (embeddings) [12].

To transform raw text into embeddings, the input must first be tokenized. Tokenization traditionally splits text into words based on whitespace and punctuation. However, this word-level approach often fails with rare or unknown words that are not included in the model's vocabulary.

To address this, **Subword Tokenization** splits the text into smaller units before the embedding. Techniques like Byte-Pair Encoding or WordPiece help mitigate the out-of-vocabulary problem and provide consistency with morphologically related words:

$$\text{unhappiness} \rightarrow [\text{un}, \text{happiness}] \rightarrow [\text{un}, \text{happy}, \text{ness}]$$

Tokenization permits shared representations with related forms, improving generalization.

The evolution of embedding representations goes from encoding methods to sophisticated, context-aware, and even multimodal embeddings that have enabled the development of LLMs:

- **Basic Representations: One-hot Encoding**

Each word is represented as a sparse binary vector of length $|V|$ (the vocabulary size). This vector contains a single 1 at the index corresponding to the word, and 0s in all other positions. It is simple, but lacks semantic information, does not scale well, and cannot generalize.

- **Static Word Embeddings**

These methods assign each word a fixed dense vector based on co-occurrence statistics or shallow models:

- **Word2Vec** [13]: Uses shallow neural networks with Continuous Bag of Words and Skip-Gram architectures to capture local syntactic relationships.
- **GloVe** [14]: Builds embeddings from global word co-occurrence matrices, capturing broader semantic relationships and enabling analogical reasoning.
- **FastText** [15]: Extends Word2Vec with subword information (character n-grams), allowing for meaningful embeddings of rare or out-of-vocabulary words.
- **Contextual Word Embeddings**
Unlike static embeddings, these models assign dynamic vectors that reflect the word’s context within a sentence:
 - **ELMo** [16]: Generates embeddings using bidirectional Long Short-Term Memorys (LSTMs); each word’s representation depends on the entire input sequence.
 - **BERT** [17], **RoBERTa** [18]: Use Transformer [19] architectures with bidirectional self-attention to produce fully contextualized embeddings foundational to modern LLMs.
- **Sentence and Document Embeddings**
Many applications require fixed-size representations of larger text units:
 - **[CLS] Token in BERT**: The final hidden state of the [CLS] token is commonly used to represent the whole sentence.
 - **Sentence-BERT (SBERT)** [20]: Fine-tunes BERT using a Siamese network for semantic similarity, producing more suitable sentence-level representations.
- **Structural Enhancements: Positional Encoding**
In Transformers, positional encoding addresses the lack of recurrence by injecting token-order information into the embeddings. This mechanism is essential for modeling sequences and is elaborated in Section 2.3.
- **Multimodal and Unified Representations**
Recent models such as CLIP [21] embed text and images into a shared vector space, enabling cross-modal tasks like image captioning or text-based retrieval. These advances push language representations beyond text, toward unified semantic spaces.

2.2 Introduction to Large Language Models

LLMs are extremely large neural networks designed for Natural Language Processing (NLP), distinguished by their massive scale and Transformer-based architecture. Modern LLMs (such as OpenAI’s GPT [22] series and Meta’s LLaMA [23] models) consist of many stacked Transformer blocks that use multi-head self-attention and feed-forward layers to capture long-range dependencies in text.

2.2.1 Historical Evolution of Large Language Models

The development of LLMs has followed a fast transformative trajectory, moving from sequential models to massively parallel architectures (for a summary, see Figure 2.1). This evolution can be summarized in the following key stages:

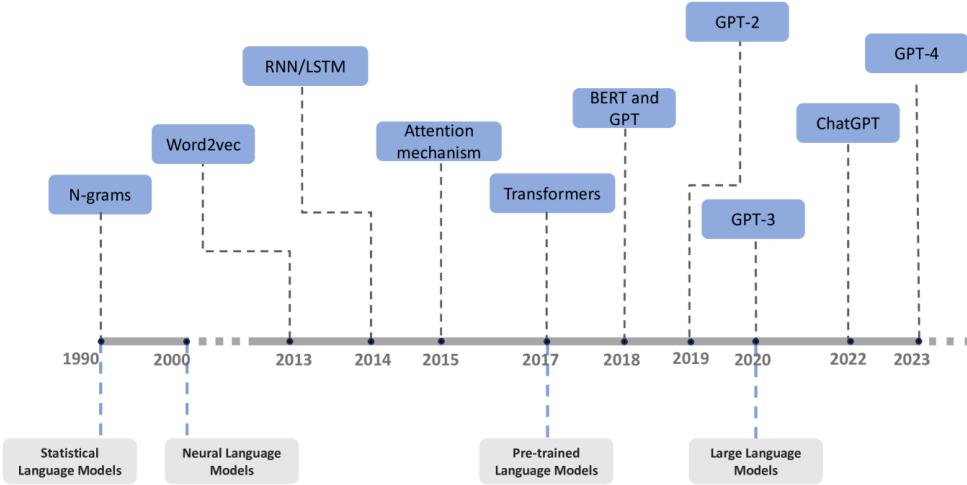


Figure 2.1: History and development of Language Models [24]

Pre-Transformer Era: RNNs and LSTMs

Before 2017, sequence modeling in NLP was dominated by recurrent architectures like Recurrent Neural Networks (RNNs) [25] and LSTM networks [26]. While effective at handling sequential data, these models suffered from limited long-range context modeling, vanishing gradients, and slow training because of their sequential nature.

The Transformer Breakthrough (2017)

The introduction of the Transformer architecture [19] revolutionized NLP by enabling fully parallelizable training and superior handling of long-term dependencies through self-attention (see Appendix A for more information).

The Transformer architecture brought two dominant language modeling paradigms:

1. **Autoregressive Models:** Predict the next token in a sequence based on previous tokens, employing a causal attention mechanism that restricts each token to attend only to its predecessors.

Autoregressive models estimate the joint probability of a sequence as:

$$P(x_1, x_2, \dots, x_n) = \prod_{t=1}^n P(x_t | x_1, \dots, x_{t-1}). \quad (2.1)$$

They use **causal self-attention**, masking future tokens to ensure unidirectional context. This makes them well-suited for *generative tasks* (e.g., text completion, summarization, code generation), but less effective for tasks requiring bidirectional understanding.

OpenAI's GPT models exemplify the autoregressive paradigm and demonstrate the power of scale:

- **GPT (2018)** [27]: Introduced unidirectional Transformer decoders for text generation.

- **GPT-2 (2019)** [28]: Demonstrated that performance improves significantly with scale (1.5B parameters).
 - **GPT-3 (2020)** [22]: With 175B parameters, enabled few-shot and zero-shot learning.
 - **GPT-4 (2023)** [29]: Introduced multimodal capabilities and alignment techniques, such as instruction tuning and Reinforcement Learning from Human Feedback (RLHF).
2. **Autoencoder Models:** Use bidirectional contexts to reconstruct or predict masked portions of the input, employing masked language modeling objectives. This approach captures richer contextual relationships, which is beneficial for tasks that require understanding of entire sequences.

Autoencoders, like BERT, use **masked language modeling**, predicting randomly masked tokens (\tilde{x}) using full context:

$$P_{\theta}(x \mid \tilde{x}) = \prod_{i \in M} P_{\theta}(x_i \mid \tilde{x}). \quad (2.2)$$

This bidirectional attention captures rich syntactic and semantic dependencies, excelling at *understanding tasks* (e.g., classification, Named Entity Recognition, Question Answering), but lacks natural generative capabilities.

BERT [17] introduced masked language modeling to capture bidirectional dependencies. Its success inspired a group of optimized variants:

- **RoBERTa** [18]: Trained with more data and dynamic masking.
- **ALBERT** [30]: Introduced parameter sharing and factorization for memory efficiency.
- **DistilBERT** [31]: A lightweight, distilled version maintaining most of BERT's performance.

A comparative summary of these two main paradigms can be found in Table 2.1 below.

3. **Hybrid Models (Encoder-Decoder Transformers):** Some models combine both paradigms through encoder-decoder architectures:

- **BART** [32]: Integrates bidirectional encoding (like BERT) with autoregressive decoding, suitable for both understanding and generation.
- **T5** [33]: Also uses an encoder-decoder design, with all tasks framed as text-to-text generation.

New Alternative Architectures and Objectives

Beyond GPT and BERT, several architectures explored novel objectives and training setups:

- **XLNet** [34]: Combined permutation-based language modeling with autoregressive benefits.
- **ELECTRA** [35]: Introduced replaced-token detection for more sample-efficient pretraining.

Scaling Transformers

Aspect	Autoregressive	Autoencoder
Attention	Unidirectional (causal)	Bidirectional
Objective	Next-token prediction	Masked-token reconstruction
Strengths	Generation	Understanding
Limitations	No future context	Not generative
Examples	GPT-2/3/4	BERT, RoBERTa

Table 2.1: Comparison of autoregressive and autoencoder models

As model sizes grew, architectural and infrastructure innovations emerged to maintain tractability:

- **Megatron-LM** [36]: Enabled model parallelism for training extremely large models.
- **GShard and Switch Transformer** [37]: Introduced Mixture-of-Experts models with sparse activation, where only subsets of parameters are active per input.

2.2.2 Training Large Language Models

Training a state-of-the-art LLM involves a carefully orchestrated combination of objectives, optimization techniques, infrastructure considerations, and alignment strategies. This section outlines the key components and design choices in this process.

Pretraining Objectives

Modern LLMs are pretrained on massive text corpora using self-supervised objectives that enable learning without manual labels. The two most common objectives are:

- **Causal Language Modeling:** Used in autoregressive models, where the model is trained to predict the next token given all previous tokens using a causal attention mask:

$$\mathcal{L}_{\text{CLM}} = - \sum_{t=1}^n \log P(x_t | x_1, \dots, x_{t-1}). \quad (2.3)$$

- **Masked Language Modeling:** Used in autoencoder models, which predict randomly masked tokens based on the full bidirectional context:

$$\mathcal{L}_{\text{MLM}} = - \sum_{t \in \mathcal{M}} \log P_\theta(x_t | x_{\setminus \mathcal{M}}), \quad (2.4)$$

where \mathcal{M} denotes the set of masked token positions.

Pretraining is conducted on large-scale datasets (e.g., Common Crawl, Wikipedia, books, code), and scales along three primary axes: model size (parameters), dataset size (tokens), and compute budget (FLOPs). Empirical scaling laws [38] demonstrate that performance improves predictably with scale until saturation or overfitting occurs.

Optimization and Regularization

LLMs are optimized using variants of stochastic gradient descent, with recommended hyperparameters such as:

- **Adam Optimizer** [39] with default $\beta_1 = 0.9$, $\beta_2 = 0.98$, $\epsilon = 10^{-9}$.
- **Weight Decay:** Typically 0.01 to regularize the model.

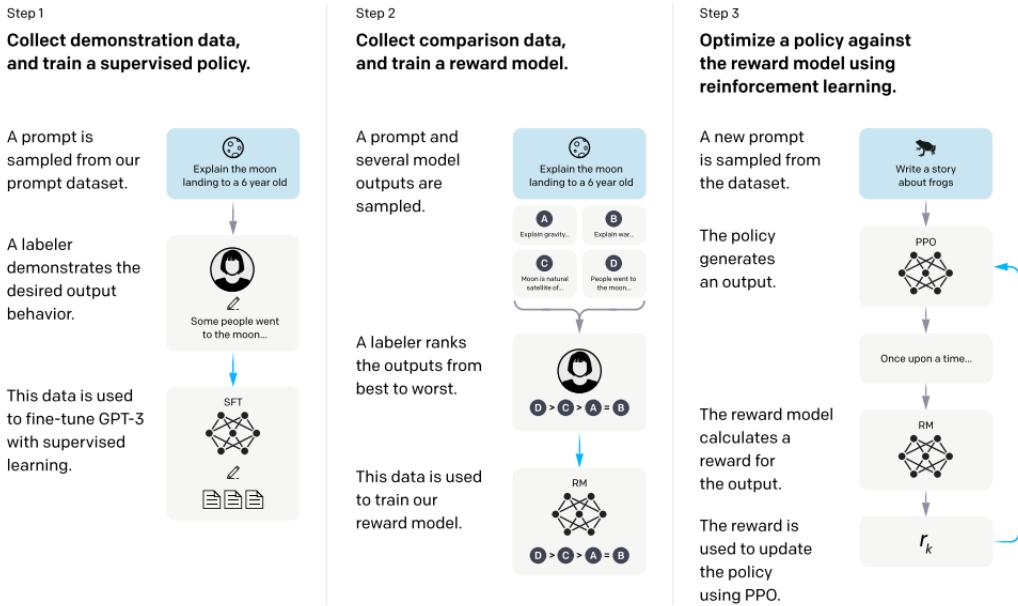


Figure 2.2: ChatGPT training steps. Source: OpenAI ChatGPT blog.

- **Learning Rate Scheduling:** Transformers often use warm-up and inverse square root decay:

$$\eta(\text{step}) = d_{\text{model}}^{-0.5} \cdot \min(\text{step}^{-0.5}, \text{step} \cdot \text{warmup}^{-1.5}) \quad (2.5)$$

- **Dropout:** Typically set around 0.1 in attention and feedforward layers.
- **Label Smoothing:** Reduces overconfidence by adjusting the one-hot target distribution.
- **Batch Size:** Often in the range of thousands of tokens per device. Large batch training requires gradient accumulation or pipeline parallelism.
- **Mixed Precision Training:** Float16 (FP16) or Brain Float16 (BF16) is used to reduce memory footprint and accelerate training.

Alignment: Instruction Tuning and RLHF

After pretraining, alignment steps are introduced to make the model more useful and safe. Instruction tuning adapts pretrained LLMs to follow natural language instructions by fine-tuning on datasets composed of prompt-response pairs (e.g., FLAN, Self-Instruct). This is followed by:

- **Supervised Fine-tuning (SFT):** Trains on human-labeled instruction-following datasets.
- **RLHF [40, 41]:** Aligns model outputs with human preferences using a reward model. The policy is updated using PPO.

In Figure 2.2 we can observe the main steps in ChatGPT training.

2.2.3 Emergent Capabilities and Current Limitations of LLMs

Before using Language Models it is important to consider their pros and cons.

Emergent Capabilities

LLMs exhibit remarkable emergent capabilities that scale with model size, data, and compute. These include not only improved accuracy on NLP benchmarks but also qualitatively new behaviors and generalization abilities.

- **Few-shot and Zero-shot Learning:** Generalize to new tasks from a handful or no examples, leveraging in-context learning.
- **Reasoning and Creativity:** Can perform symbolic reasoning, problem-solving, and even creative tasks such as story writing, poetry, or code generation.
- **Tool Use and Agent Behavior:** Are increasingly equipped to call APIs, use tools (e.g., Toolformer [42], plug-ins), and behave as autonomous agents capable of planning and decision-making (e.g., ReAct [43], AutoGPT [44]).
- **Multimodal Capabilities:** Some models integrate vision and language, supporting tasks like image captioning, visual Question Answering, and grounding.

Current Limitations and Challenges

Despite their capabilities, LLMs still present significant open challenges:

- **Hallucinations:** Often generate fluent but factually incorrect outputs due to reliance on statistical patterns rather than verifiable truth.
- **Bias and Fairness:** May inherit or amplify societal biases present in training data, affecting fairness across demographics and sensitive contexts.
- **Interpretability and Controllability:** It remains difficult to explain, steer, or precisely control outputs beyond prompt engineering and heuristic constraints.
- **Data Contamination and Leakage:** Benchmarks may be compromised due to overlap with training data, inflating reported performance.
- **Cost and Sustainability:** Training and deploying LLMs involves immense computational cost, raising concerns over energy consumption and environmental impact.

2.3 Optimizing Large Language Models

LLM optimization is not a linear pipeline but a dynamic, iterative process that involves choosing the right technique based on the type of error encountered. This process¹ can be visualized along two orthogonal axes:

- **Context Optimization:** Enhances factual accuracy by providing the model with relevant external knowledge, typically through *Retrieval-Augmented Generation (RAG)*.
- **LLM Optimization:** Improves the model's consistency, formatting, tone, and reasoning, often via *prompt engineering* or *fine-tuning*.

This is the state of the art workflow (see Figure 2.3 for a schematic approach) in LLMs optimization:

1. Start with **prompt engineering**: Write clear instructions, add context with structured inputs, or few-shot examples and evaluate performance using a predefined test set.

¹<https://platform.openai.com/docs/guides/optimizing-lm-accuracy>

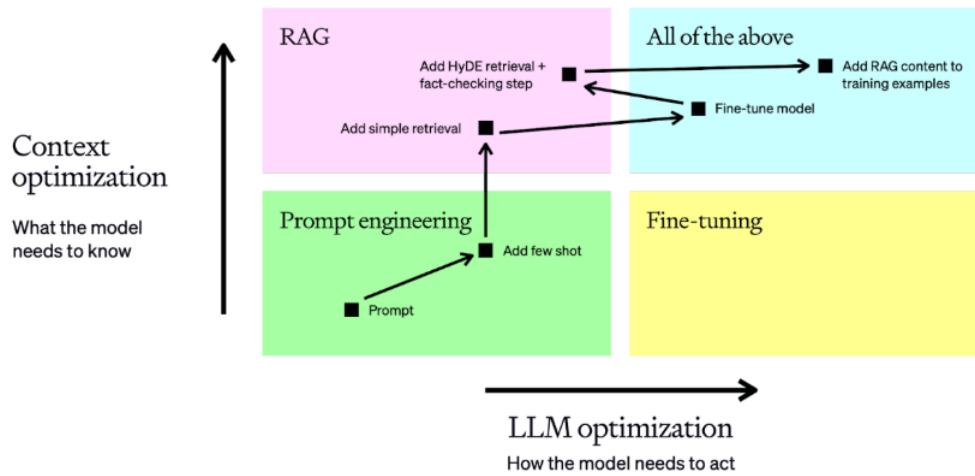


Figure 2.3: LLM optimization. Source: OpenAI.

2. If performance issues stem from knowledge gaps: Use **RAG** to dynamically retrieve and inject relevant information into the prompt.
3. If the model exhibits inconsistent behavior: Apply **fine-tuning** using domain-specific data to enforce consistent outputs.
4. Combine methods when necessary: **RAG + fine-tuning** is often required in production systems for optimal results.

2.3.1 Prompt Engineering

Prompt engineering consists of systematically designing prompts to guide a Language Model towards desired responses. It serves as the foundational optimization strategy for LLMs, significantly affecting precision, consistency, and task-specific performance. Classical approaches include:

- **Zero and Few-shot** [45]: Provide clear instructions (zero-shot) or some examples (few-shot) to guide the model without additional training.
- **Chain-of-Thought (CoT)** [46]: Asking models to explicitly reason step-by-step improves performance significantly on reasoning and arithmetic tasks.
- **Reasoning and Acting (ReAct)** [43]: Propose a framework combining explicit reasoning steps with external action calls (e.g., API calls or tool-use) allowing models to better navigate tasks requiring both reasoning and external information retrieval or interaction.

Informed by OpenAI guidelines, best practices include:

1. **Clear and Explicit Instructions:** Define explicitly what constitutes an acceptable response, minimizing ambiguity.
2. **Structured Formatting:** Use consistent, structured formatting (e.g., numbered lists, structured JSON responses) to improve model consistency.

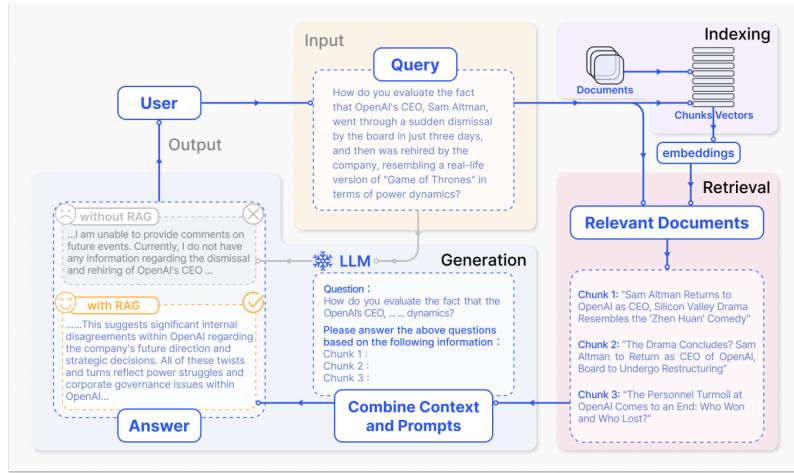


Figure 2.4: Retrieval-Augmented Generation pipeline [48]

3. **Contextual Examples (Few-shot):** Provide representative examples.
4. **Iterative Optimization:** Evaluate prompt variations systematically, adjusting instructions, examples, and structures based on objective evaluation metrics and test sets.
5. **Prompt Decomposition:** Decompose complex tasks into simpler sub-tasks.
6. **External Tools and References:** Incorporate external reference texts or tool calls (ReAct prompting) to overcome knowledge limitations inherent in the base model.

However, prompt engineering faces challenges: context positioning (e.g., “lost in the middle” [47]), prompt clarity and complexity, and scalability constraints as the context length increases.

2.3.2 Retrieval-Augmented Generation (RAG)

RAG enhances LLMs by dynamically retrieving (see Figure 2.4) relevant external knowledge to overcome hallucinations, outdated training, and opaque reasoning. By semantically retrieving chunks from vector databases and conditioning the LLM on them, RAG improves factuality and relevance.

Current RAG paradigms [48] include (see Figure 2.5) naive RAG (basic retrieval and generation), advanced RAG (query rewriting, reranking, context compression), and modular RAG (combining retrieval with tools[49], memory modules, or specialized adapters).

Evaluation frameworks for RAG systems measure retrieval and generation using metrics like BLEU, ROUGE, MRR, or EM, covering dimensions such as context relevance, answer faithfulness, and noise robustness [50].

Fine-tuning adapts internal model knowledge, RAG offers on-demand, up-to-date external information, and combining both approaches can obtain robust results. Future work targets hybrid RAG-fine-tuning methods [51], multimodal retrieval, and greater robustness to noisy or contradictory data [48].

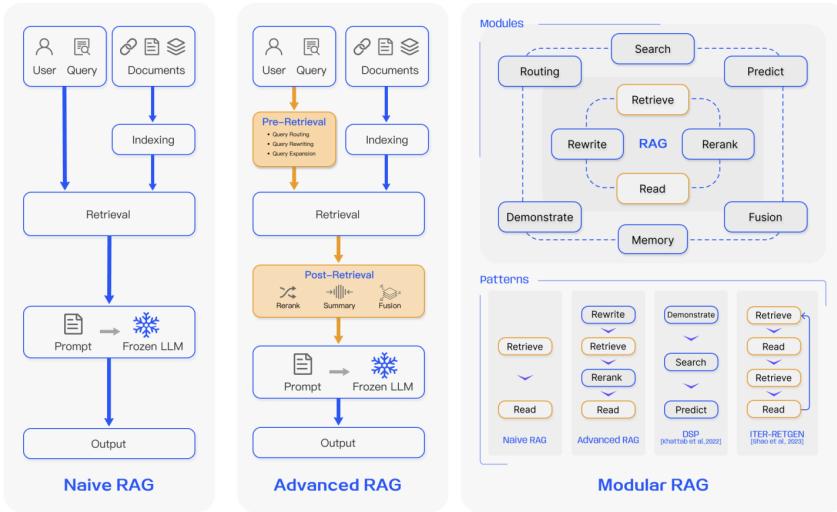


Figure 2.5: Retrieval-Augmented Generation paradigms [48]

2.3.3 Fine-tuning of Large Language Models

Fine-tuning is the process of adapting a pre-trained LLM to a task by continuing training on task-specific data. This adaptation updates the model's internal parameters, aligning the model's behavior more closely with the desired objectives. Fine-tuning is essential when general-purpose LLMs fall short on domain-specific or task-specific performance, and remains one of the most effective approaches for improving model accuracy in supervised learning settings.

Stages of fine-tuning:

- Dataset Curation and Preprocessing:** Collect, clean, annotate, and tokenize Task-specific data. High-quality datasets are crucial to avoid introducing biases or overfitting.
- Initialization:** Starting from a checkpoint of a pretrained model, retaining the learned representations.
- Training Loop:** The model is updated through backpropagation using stochastic gradient descent, Adam, or its variants. The loss function is selected based on the task (e.g., cross-entropy for classification, sequence loss for generation).
- Regularization and Optimization:** Techniques such as dropout, learning rate scheduling, gradient clipping, early stopping, and weight decay are used to prevent overfitting and stabilize training. Mixed precision training may be employed to accelerate convergence.
- Evaluation and Validation:** Model performance is evaluated using appropriate metrics: accuracy, F1, BLEU, ROUGE, perplexity, etc. Evaluation protocols may include zero-shot, few-shot, or full supervision, depending on the use case.

Types of fine-tuning Paradigms:

- Full fine-tuning:** All model parameters are updated. It has high task adaptation capacity but it is computationally expensive and prone to catastrophic forgetting.
- Domain Adaptation:** Fine-tuning on domain-specific unlabeled corpora.

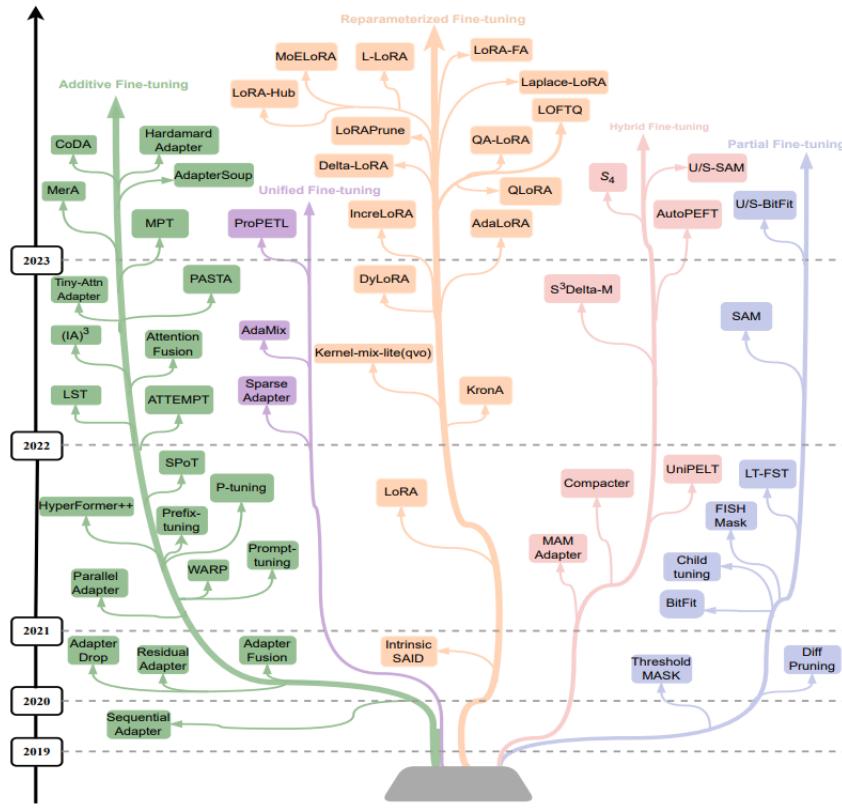


Figure 2.6: Taxonomy of Parameter-Efficient Fine-Tuning methods [53]

- **Multi-task Fine-tuning:** Simultaneous training on multiple tasks to encourage generalization and prevent overfitting. Often used in setups like T5 or multitask BERT.
- **Continual Learning / Lifelong fine-tuning:** Models are incrementally fine-tuned as new data becomes available over time. Methods such as rehearsal, regularization-based constraints, and adapter modules help avoid forgetting prior tasks.

Fine-tuned models may face challenges such as overfitting on small datasets, catastrophic forgetting of generic capabilities, or high computational costs. Combining fine-tuning with RAG, as well as instruction tuning and RLHF strategies, is emerging as a practical solution to balance task specialization with generalization, scalability, and safety.

2.3.4 Parameter-Efficient Fine-tuning

As large-scale pretrained LLMs grow in size and complexity, full fine-tuning becomes increasingly impractical because of its high computational and memory requirements (e.g., models such as Falcon-180B [52] demanding upwards of 5120GB of memory for full fine-tuning).

PEFT [53] enable task-specific adaptation of LLMs by modifying only a small subset of parameters while preserving most of the pretrained weights. PEFT methods aim to maintain or surpass the performance of full fine-tuning, with a fraction of the trainable parameters.

PEFT methods seek to study the structure of the Transformer to achieve this approach, emerging a large number of techniques (see Figure 2.6). For this reason, to correctly understand this section, we recommend visiting Appendix A and learning about the Transformer architecture.

This section presents an overview of the most significant ones (see Figure 2.7). PEFT techniques can be categorized into the following five main types:

1. **Additive Fine-tuning:** Introduces extra trainable parameters without modifying pre-trained weights. The most representative methods are:

- **Adapter-based Fine-tuning:** Adapters are small neural modules inserted within transformer blocks. The sequential adapter is defined as [54]:

$$X = \text{ReLU}(XW_{\text{down}})W_{\text{up}} + X, \quad (2.6)$$

where $W_{\text{down}} \in \mathbb{R}^{d \times k}$ and $W_{\text{up}} \in \mathbb{R}^{k \times d}$.

Variants include:

- Residual Adapter [55], Parallel Adapter [55], and AdapterDrop [56].
- CoDA [57]: selects important tokens via a routing function.
- Tiny-Attn Adapter [58]: introduces a multi-head attention-like module with expert averaging.
- AdapterFusion [59], MerA [60], Hyperformer++ [61], and AdapterSoup [62]: support adapter composition and transfer.
- **Soft Prompt-based Fine-tuning:** Soft prompts are learnable embeddings inserted into the input or intermediate representations [63]:

$$\hat{X} = [P, X] \in \mathbb{R}^{(l+n) \times d}, \quad (2.7)$$

where P is a matrix of l prompt tokens.

Prefix-tuning modifies attention layers:

$$\text{head} = \text{Attn}(QW_q, [\hat{P}_k, KW_k], [\hat{P}_v, VW_v]), \quad (2.8)$$

where $\hat{P}_k = \text{FFN}(P_k)$, $\hat{P}_v = \text{FFN}(P_v)$.

Notable works include WARP [64], Prompt-tuning [63], Prefix-tuning [65], P-tuning [66], ATTEMPT [67], and Multitask Prompt Tuning [68].

2. **Partial Fine-tuning:** Selectively updates a subset of existing parameters (e.g., biases or masked weights).
- **Bias Update (BitFit)** [72]: Updates only the bias terms in attention and Feed-Forward Network (FFN) layers.
 - **Pretrained Weight Masking** [73]: Uses a mask matrix M to selectively retain pre-trained weights:

$$\hat{W} = M \odot W, \quad (2.9)$$

with \odot as the Hadamard product. Examples: Threshold-Mask [74], FISH Mask [75].

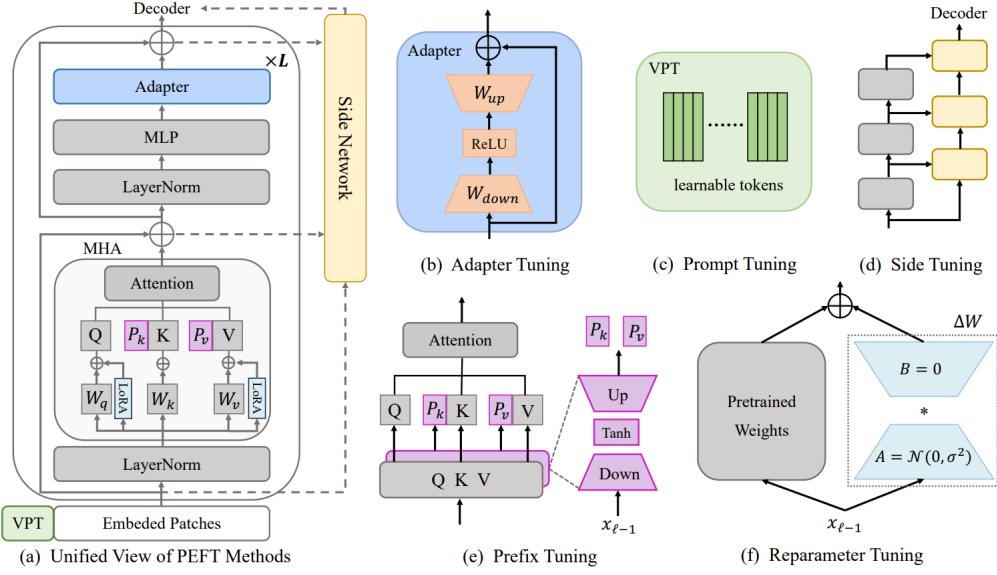


Figure 2.7: Main PEFT methods [84]

- **Delta Weight Masking:** Updates a sparse delta of the original weight. For example, LT-SFT [76] computes:

$$\Delta W = M \odot \nabla_W L(W). \quad (2.10)$$

- Other methods include Child-Tuning [77], Diff Pruning [78], and SAM [79].

3. **Reparameterized Fine-tuning:** Uses low-rank or alternative representations to reduce update dimensions.

The most representative method is LoRA, which inserts low-rank matrices alongside pre-trained weights [80]:

$$\Delta W = W_{\text{down}} W_{\text{up}}, \quad W_{\text{down}} \in \mathbb{R}^{d \times r}, \quad W_{\text{up}} \in \mathbb{R}^{r \times k}. \quad (2.11)$$

Derivatives include: DyLoRA [81], AdaLoRA [82] and QLoRA [83].

4. **Hybrid Fine-tuning:** Combines multiple PEFT strategies.

Manual combinations (Compacter [85], UniPELT [86]) and automatic combinations (AutoPEFT [87]) combine multiple PEFT strategies across layers or tasks, leveraging optimization techniques.

5. **Unified Fine-tuning:** Provides a general framework encompassing multiple PEFT forms.

Unified approaches (AdaMix [88], SparseAdapter [89], ProPETL [90]) generalize multiple PEFT paradigms into a single reusable framework across layers and tasks.

Empirical results show that PEFT methods significantly reduce the number of trainable parameters (often under 1%) while achieving comparable or superior performance to full fine-tuning on benchmarks like GLUE [91], WMT16 [92], and MMLU [93]. For example, **ProPETL-Adapter** [90] outperformed full fine-tuning of RoBERTa with only 1.5% parameters and **QLoRA** [83] enabled 4-bit quantization fine-tuning with 75% less memory on LLaMA-13B [23].

2.3.5 Alignment Algorithms

Preference learning [94, 95] for LLMs aims to train models whose outputs align more closely with human preferences [96]. Given a distribution $P(x, y)$ of human preferences over prompt-response pairs (x, y) , the goal is to produce a new model such that

$$P(x, y_{\theta'}(x)) > P(x, y_{\theta}(x)). \quad (2.12)$$

This process is guided by feedback that reflects the degree of human preference, and it can be **direct** (labels, rankings, reward scores) or **model-based** (evaluations from reward models or LLM-as-a-judge systems).

Preference learning comprises four elements:

- **Model** (π_{θ}): the base or fine-tuned LLM being optimized.
- **Data** (D): can be collected online (on-policy) or offline (off-policy).
- **Feedback** (r): scalar reward or preference label.
- **Algorithm** (A): updates model parameters using the above.

Both reinforcement learning based and SFT-based methods for ralignment can be formulated under a unified gradient framework [96]:

$$\nabla_{\theta} = \mathbb{E}_{(q, o) \sim D} \left[\frac{1}{|o|} \sum_{t=1}^{|o|} \delta_A(r, q, o, t) \nabla_{\theta} \log \pi_{\theta}(o_t | q, o_{<t}) \right], \quad (2.13)$$

where D is the data source of questions q and outputs o , and δ_A is the gradient coefficient determined by algorithm A and feedback r .

These algorithms can be divided into four main groups:

1. **Point-wise Methods:** optimize the probability of each output independently with a reward function.
 - Rejection Sampling Fine-Tuning:

$$L_{RS} = - \sum_t \log \pi_{\theta}(y^+ | t | x, y^+ < t) \quad (2.14)$$

- PPO [97]:

$$J_r(\theta) = \sum_x \mathbb{E}_{y \sim \pi_{\theta}(\cdot | x)} \left[r(x, y) - \beta \log \frac{\pi_{\theta}(y | x)}{\pi_{\text{ref}}(y | x)} \right] \quad (2.15)$$

2. **Pair-wise Contrast:** Learn preferences by comparing pairs (see Figure 2.8) of responses preferred vs. rejected (e.g., DPO [98], IPO [99], SimPO [100]). The most significant method is DPO:

$$L_{\text{DPO}}(\pi; \pi_{\text{ref}}) = - \log \sigma \left(\beta \log \frac{\pi(y^+ | x)}{\pi_{\text{ref}}(y^+ | x)} - \beta \log \frac{\pi(y^- | x)}{\pi_{\text{ref}}(y^- | x)} \right) \quad (2.16)$$

3. **List-wise Contrast:** generalizes from 2 to n ranked responses, constructing loss terms over all ordered pairs (e.g., RRHF [101], PRO [102], GRPO [103]).

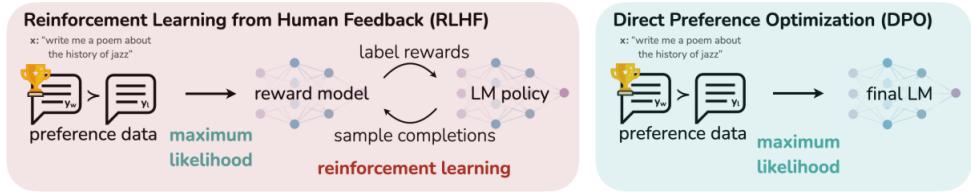


Figure 2.8: PPO vs DPO[98]

4. **Training-Free Alignment:** Modify input prompts (e.g., BPO [104], ICDPO) or output distributions (e.g., FUDGE [105], RAIN [106]).

Evaluation techniques combine traditional **rule-based metrics** (accuracy, F1, ROUGE, EM), which are limited for open-ended tasks, with **LLMs-based** assessments performing pairwise comparisons or answer grading, though these may suffer from position bias and verbosity preferences.

2.4 LLM-Based Autonomous Agents

The LLMs revolution has catalyzed the development of autonomous agents (see Figure 2.9) capable of perceiving environments, making decisions, and executing actions. This section explains the current research [107, 108] on LLM-based agents by integrating foundational concepts with recent technical advances.

The concept of agents appears for the first time in philosophical traditions tracing back to Aristotle and Hume, describing entities that have desires, beliefs, intentions, and action capacities. In AI, an agent refers to an artificial entity that senses its environment using sensors, makes decisions, and takes actions using actuators. They are computational entities that exhibit key properties:

- **Autonomy:** Operates without direct human intervention.
- **Reactivity:** Responds to environmental changes.
- **Pro-activeness:** Displays goal-directed behavior.
- **Social ability:** Interacts with other agents/humans.

As you can see in Table 2.4, recent advances in LLM position them as promising substrates for implementing key properties.

Agent Requirement	LLM Capability
Natural Language Interaction & Social ability	High-quality generation and deep understanding
Knowledge Management	Linguistic, commonsense, and domain knowledge storage
Memory Mechanisms	Summarization, compression, and retrieval
Reasoning, Planning, Generalization and Reactivity	Chain-of-Thought & task decomposition

Table 2.2: Alignment of LLMs capabilities with agent requirements

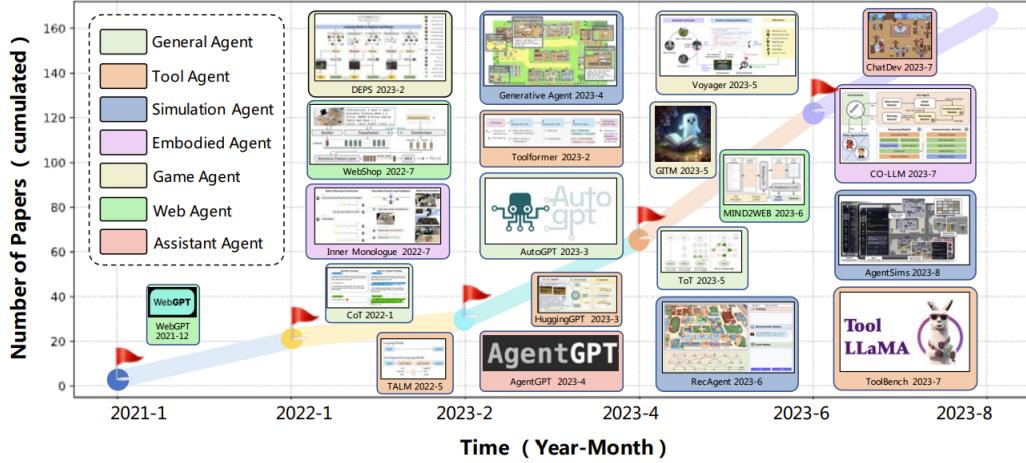


Figure 2.9: Agent development timeline [107]

In this context, agents acquire capabilities either **through fine-tuning** on supervised data (e.g., CoH [109], EduChat [110]), synthetic datasets (e.g., ToolBench [111]), or real-world data (e.g., WebShop [112]), or alternatively without fine-tuning by leveraging **prompt engineering** (e.g., CoT [46], SocialAGI [113]) and **mechanism engineering** approaches such as trial-and-error (e.g., DEPS [114], RoCo [115]), crowdsourcing, experience accumulation (e.g., Voyager [116]), or self-driven evolution (e.g., LMA3 [117]).

2.4.1 Core Components

The architecture of an LLM-based agent (see Figure 2.10) typically integrates:

- **Perception:** Handles multimodal inputs via textual, visual (e.g., ViTs [118]), or auditory modalities.
- **Profile:** Encodes the agent's identity, goals, preferences, and constraints. Profiles can be handcrafted, LLM-generated, or aligned with datasets.
- **Brain (LLM Core):** Acts as a reasoning engine. Provides natural language interaction, planning, generalization, and common sense knowledge.
 - **Memory Modules:** Manages short and long-term storage, including read, write, and reflection operations. Supports knowledge accumulation and behavior consistency.
 - * Reading: Weighted scoring functions (e.g., relevance, similarity) using methods like FAISS [119] or HNSW [120].
 - * Writing: Deals with memory duplication (e.g., condensation, aggregation) and overflow (e.g., FIFO buffers [121]).
 - * Reflection: Summarizes past events into high-level insights (e.g., Generative Agents [122], ExpeL [123]).
 - **Planning Modules:** Decomposes tasks, generates strategies, and updates plans based on feedback. Includes single/multi-path and reactive planning.

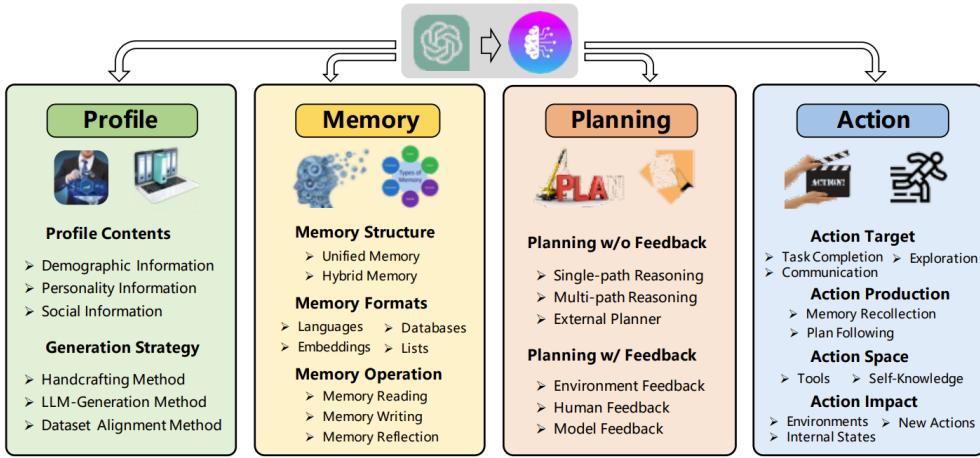


Figure 2.10: Core components of an Agent [107]

- * Without Feedback: Includes **single-path** (e.g., CoTs [46], ReWOO [124]) and **multi-path** (e.g., ToT [125], CoT-SC [126]) reasoning.
- * With Feedback: Uses environmental or human input to dynamically replan (e.g., ReAct [43], Voyager [116], LLM-Planner [127]).
- * External Planning: Converts tasks to Planning Domain Definition Language (PDDL) and solves them via symbolic planners (e.g., LLM+P [128]).
- **Action:** Execute decisions via natural language, tool invocation (APIs), or embodied interfaces in real or virtual environments.
 - **APIs** (e.g., Toolformers [42], Gorilla [49], API-Bank [129], TaskMatrix.AI [130]).
 - **Databases/Knowledge Graphs** (e.g., ChatDBs [131], MRKL [132]).
 - **External Models** (e.g., ChemCrow [133], MM-ReAct [43], ViperGPT [134]).

2.4.2 Evaluation Strategies

The evaluation of LLM-based agents is a crucial and active research area. Some of the most used metrics are:

Subjective Evaluation:

- Human annotation (e.g., Generative Agents [122]).
- Turing tests (e.g., EmotionBench [135], Out of One, Many [136]).

Objective Evaluation:

- Metrics: task success, similarity to humans, efficiency.
- Protocols: simulation-based, social evaluation, multi-task, software testing.
- Benchmarks: AgentBench [137], ToolBench [111], WebArena [138], EmotionBench [135].

2.4.3 Applications

LLM-based agents are applied across diverse contexts:

- **Single-Agent Systems:** Task automation, lifelong learning, scientific research.
- **Multi-Agent Systems:** Collaboration (e.g., ChatDev [139], MetaGPT [140]), debate, or simulation (e.g., AgentSims [141], SocialAI School [142]).
- **Human-Agent Collaboration:** Instructor-executor and peer-based interaction.
- **Agent Societies and Simulation:**
 - Social dynamics: cooperation, conflict, group behavior.
 - Simulation environments: textual, virtual, or physical.
 - Ethical considerations: adversarial robustness, alignment, misuse.

2.4.4 Challenges and Open Problems

Despite remarkable progress, LLM-based agents face significant challenges:

- **Role-playing Limitations:** Poor modeling of uncommon roles or cognitive traits.
- **Generalized Human Alignment:** Need to simulate diverse values for different applications.
- **Prompt Robustness:** Prompt-based pipelines are fragile and context-sensitive.
- **Hallucination:** False but confident outputs can be harmful.
- **Knowledge Boundary:** Over-informed agents break realism in simulations.
- **Efficiency:** High inference cost due to autoregressive LLM structure.

LLM-based agents bridge symbolic, reactive, and learning-based paradigms. Future work must address memory abstraction, planning feedback, agent alignment, and scalable evaluation to move toward robust and responsible agentic AI.

2.5 Humanized Agents Societies

Traditionally, social experiments relied on living participants, but lacked flexibility and scalability. Researchers now envision simulated societies where trustworthy agents, including LLM-based ones [143], can replicate human behavior [122, 144, 145]. This section explores how agents progress from individuality to sociality [108].

2.5.1 Environment for Agent Society

Environments used to simulate these societies typically fall into three categories:

- **Text-based:** natural or structured text environments (e.g., Textworld [146], CAMEL [147]).
- **Virtual sandbox:** visually immersive simulations (see Figure 2.11).
- **Physical:** real-world settings (e.g., PaLM-E [148], RoboAgent [149]).

2.5.2 Behavior and Personality

- **Individual behavior:** Agents perceive input (PaLM-E [148]), process it internally (planning, reasoning, reflection [46, 43, 128]), and produce output (actions, opinions [140]).

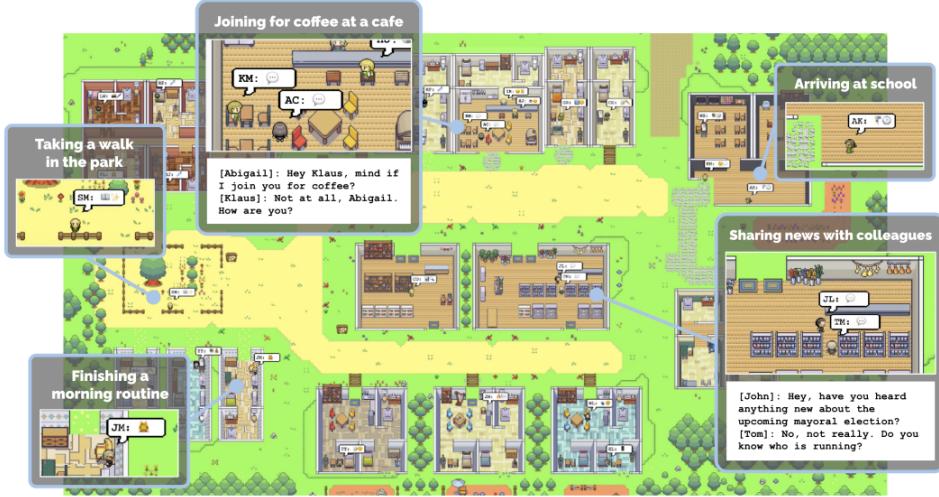


Figure 2.11: Generative Agents: Interactive Simulacra of Human Behavior [122]

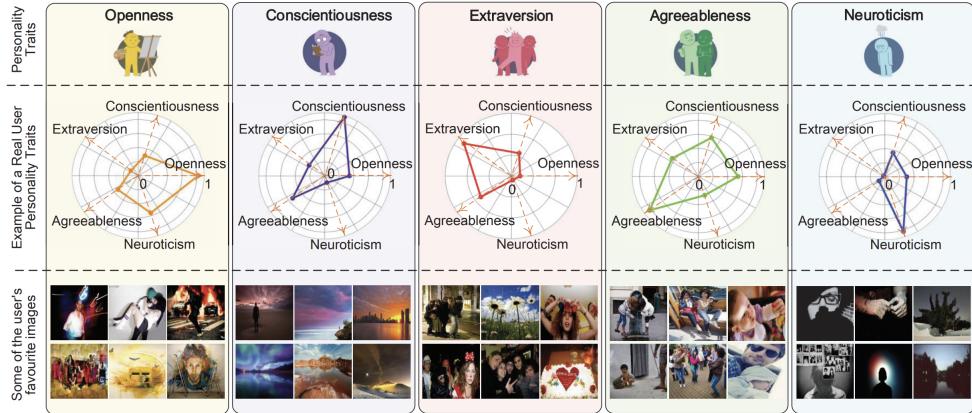


Figure 2.12: Reinforcement Learning with Big Five Personality Traits [158]

- **Group behavior:** Spans cooperation (e.g., ChatDev [139], AutoGen [150], AgentVerse [151]) and altruism, as well as neutrality [152] and even negative destructive behaviors [151].
- **Personality:** Including cognition [153], emotion [154], empathy [155]), and character (Big Five [156] and MBTI [157]).

As will be seen in later chapters, **the achievement of personality within the Big Five framework** of agents will be a central theme in this thesis. Although there are previous studies on this approach (see Figure 2.12), they typically show little granularity and focus on a single variable [159].

2.5.3 Society Simulation with LLM-based Agents

Simulated societies provide a powerful tool for investigating complex social dynamics. They enable the study of collective intelligence (e.g., ChatDev [139], MetaGPT [140]), the propagation of information (see Figure 2.13) and attitudes within social networks (as in S³ [160]), and sophisticated ethical decision-making processes, like trust, deception, and incomplete information (e.g., the Werewolf game [161]). Moreover, these agent-based simulations could improve

2.6. State of the Art in Urban Simulation at the MIT City Science Group

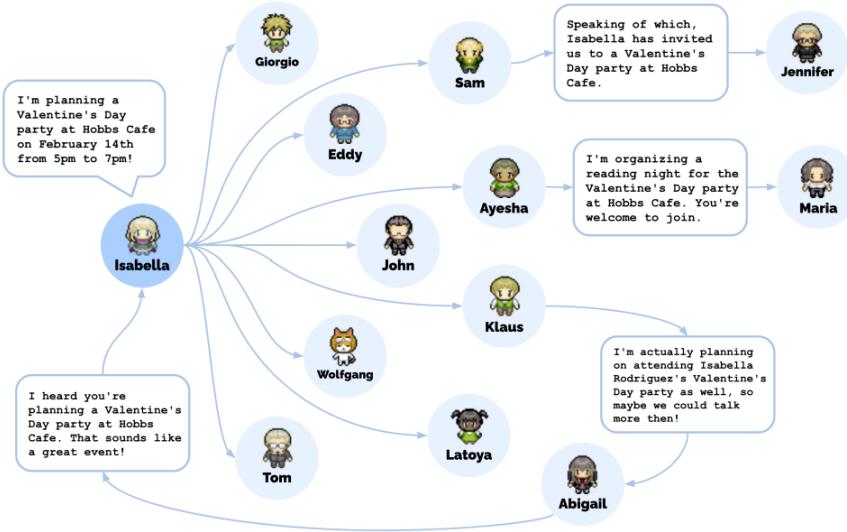


Figure 2.13: Social interaction between agents [122]

policy research, allowing exploration of different economic and political configurations and their social impacts [162].

Simulated societies are open (permitting flexible addition or removal of agents and resources), persistent (maintaining stable organizational structures over time), and follow consistent rules and protocols that ensure coherent interactions [143].

However, these systems also present critical risks. They may generate unintended harmful social phenomena, discrimination or exclusion to systemic bias [163]. The persistence of stereotypes and the reproduction of societal prejudices embedded in training data remain challenging issues [164]. Privacy and data security are also a concern when agents retain sensitive user information over extended periods [165].

2.6 State of the Art in Urban Simulation at the MIT City Science Group

The City Science Group has an extensive research trajectory in urban simulation. Its flagship project is **CityScope** [1], an initiative that comprises a suite of tangible and digital platforms designed to address challenges in spatial design and urban planning. These tools include simulations to assess the impact of interventions in urban environments, as well as systems that promote community engagement.

Within this initiative, ABMs stand out as a core methodological framework. In fact, there is a strong collaboration between the City Science Group and the **GAMA** [166] ABM library team, recognized as state of the art in urban agent-based simulation.

Particularly notable are studies on mobility simulation and swarm-based modeling [3, 4], autonomous vehicle mobility simulation [5, 6, 7, 2, 8], and even applications of agent-based models for safety assessments in informal settlements [9].

In recent years, these models have been progressively integrated with new AI techniques, such as Deep Learning approaches in CityScope visualization [168]. Furthermore, there is active re-

State of the Art

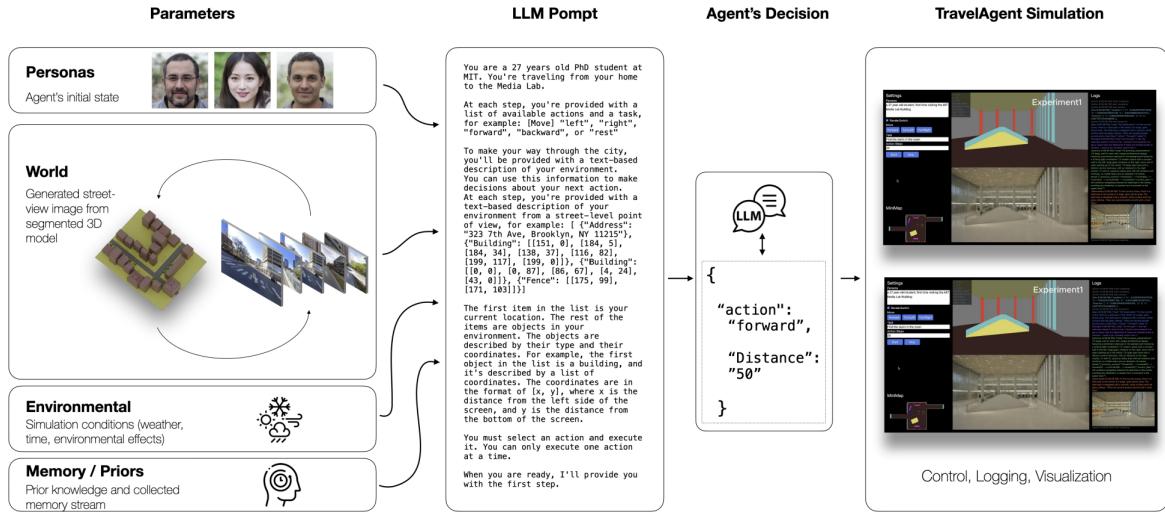


Figure 2.14: Travel Agent [167]

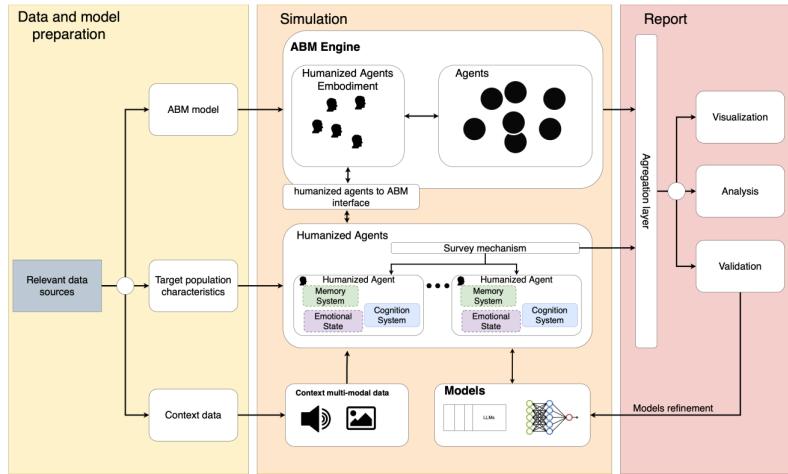


Figure 2.15: h-ABM framework [11]

search on the feasibility of novel LLM-driven agent-based societies [10, 169]. In Figure 2.14, we can observe how LLMs are combined with agent-based strategies to simulate human mobility in the city [167].

Within this evolving landscape, **humanized Agent-Based Models (h-ABMs)** propose combining traditional ABM paradigms with the reasoning and generative capabilities of LLMs, enhancing the realism of human agents in simulations. This new framework (which is precisely the foundation for this thesis) introduces a modular architecture, composed of multiple subsystems including memory systems (long-term, short-term, working memory), cognition systems, and emotional state modules. This architecture allows agents to exhibit human-like features such as remembering past experiences, adapting to new situations, expressing opinions, and simulating emotional responses.

The h-ABM framework (see Figure 2.15) incorporates conversational and survey capabilities, allowing agents to answer questions and share subjective perspectives about their simulated

2.6. State of the Art in Urban Simulation at the MIT City Science Group

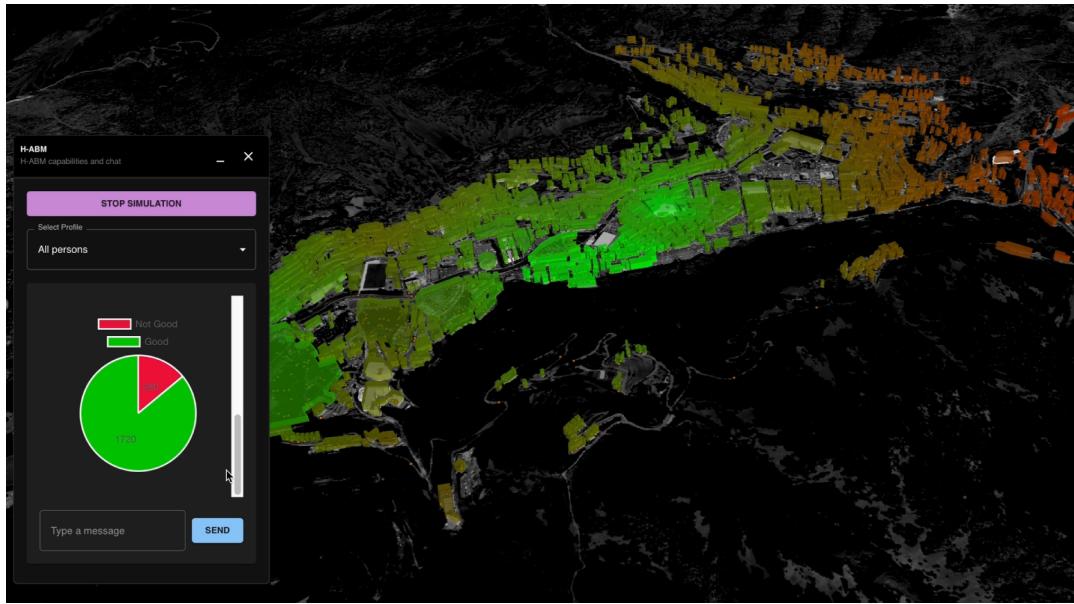


Figure 2.16: h-ABM simulation. Asking agents: How would you rate the access to shopping amenities from your residence considering walking as your only option? Source: MIT City Science Group.

environment, providing qualitative data in addition to quantitative simulation metrics. For example, agents can report on their perceived access to amenities, describe their satisfaction levels, or express sentiments about safety and well-being, enriching the policy insights derived from the model.

In Figure 2.16, we can see how a population of humanized agents responds to the question "How would you rate the access to shopping amenities from your residence considering walking as your only option?", illustrating the framework's ability to capture opinions, sentiments, and contextual reflections of simulated agents.

Chapter 3

Design and Development

This chapter details the design decisions and development strategies adopted throughout this work. It outlines the technical foundations, the proposed architecture, and the tools, technologies, and methodologies used in the implementation. Furthermore, it discusses the limitations encountered and the justifications behind each design choice, aiming to ensure the coherence and robustness of the proposed system.

3.1 Problem Analysis: Optimization of LLMs for Human-Centric Agents

While the group already possesses a rich simulation infrastructure (see Sections 1.3 and 2.6), this work focuses on a key challenge: *to humanize the agents* by incorporating psychologically grounded behavior, particularly through **personality modeling**.

Other researchers in the group are working on agent traits such as emotional variability within the population. Therefore, this thesis will focus particularly on a central aspect of humanizing an agent: *the study of agent personality*.

This problem is a rather open goal, and although an LLM has limitations in achieving it (as we have studied in previous chapters), it is still the closest technology to achieve this goal. So taking into account these limitations (see Subsections 2.2.3 and 2.5.3) we aim to identify the most effective approach to address this challenge.

To this end, two main objectives were defined: to provide a context to the model with **a profile broad enough** to understand the human traits, and to **adapt the model** to adhere to a personality.

Following the framework for working with LLMs introduced in the state of the art (Section 2.3), two different phases will be developed in this project:

- **Context Optimization:** The aim of this phase is to create a synthetic population generation pipeline, where the traits of each individual are stored in a JSON format with a hierarchical structure. This structure facilitates the application of RAG to retrieve and incorporate only the desired characteristics into the simulation of that agent.
- **LLM Optimization:** In this phase, language models will be fine-tuned using PEFT and preference alignment algorithms (see Subsections 2.3.4 and 2.3.5) to induce personality-consistent responses.

3.1. Problem Analysis: Optimization of LLMs for Human-Centric Agents



Figure 3.1: Big 5 personality traits. Source: AI generated

3.1.1 Scientific modeling of personality: Big Five Personality Traits

As we have just explained, agents' personality is a central theme in this study, but in order to define what personality is, we need to ask several questions: How do we model a personality? How do we distinguish between different types of personalities? Is there any model approved by the scientific community to do so?

The answer is the Big Five Personality Traits [170, 171, 172], also known as the OCEAN model (see Figure 3.1), constitute a widely accepted dimensional framework in psychology to characterize individual differences. Developed through empirical lexical analysis and refined via factor analysis, the model captures five major dimensions of human personality:

- **Openness to Experience:** Imagination, aesthetic sensitivity, curiosity, and preference for novelty.
- **Conscientiousness:** Orderliness, responsibility, goal-directed behavior, and self-discipline.
- **Extraversion:** Sociability, assertiveness, and energetic expression.
- **Agreeableness:** Compassion, altruism, and cooperativeness versus antagonism.
- **Neuroticism:** Emotional instability, anxiety, and vulnerability to stress.

The Big Five model has been validated across cultures, languages, and demographic groups. Unlike typological approaches, it reflects the natural distribution of psychological traits in the population, providing a realistic basis for stochastic simulation. It is extensively used in:

- Psychometrics and psychological assessment.
- Organizational behavior and job performance prediction.
- Social behavior and interpersonal modeling.

Each trait is measured on a continuous scale (0 to 1). Thus, this study models the personality of an agent as a **vector of five variables ranging from 0 to 1**. This vector representation allows for nuanced and high-resolution behavioral modeling:

Design and Development

- Fine-grained variation between agents.
- Easy integration with regression and probabilistic sampling.
- Direct interpretability in LLM reasoning tasks.

Although there are other theories for modeling personality (see Figure 3.3), they were ultimately discarded compared to the Big 5 model. A comparison with each of the main alternatives can be found in Table 3.1.

Model	Structure	Limitations Compared to Big Five
MBTI	16 personality types based on 4 dichotomies	Typological: forces discrete categories, lacks statistical robustness, poor test-retest reliability.
Eysenck's PEN Model	3 traits: Psychoticism, Extraversion, Neuroticism	Too coarse for modeling diverse behaviors, lacks openness/agreeableness dimensions.
HEXACO Model	6 traits: adds Honesty-Humility to Big Five	Valuable addition but less established, fewer available datasets and regressors.
DISC Profile	Dominance, Influence, Steadiness, Conscientiousness	Business-oriented, less predictive in scientific settings, categorical labels.

Table 3.1: Comparison among the main personality models

3.2 Phase I: Context Optimization

In this phase, a LLM-based pipeline was designed to transform demographic data into complete humanistic profiles. These profiles follow a structured JSON format and include multi-dimensional attributes.

3.2.1 Synthetic Population Generation

The objective of this pipeline is to generate **complex, diverse, and realistic** human agent profiles by leveraging LLMs in combination with structured statistical modeling.

To address this task, a clear target output was defined: each synthetic agent must be represented as a structured JSON capturing demographic, psychological, cognitive, emotional, social, and physical dimensions.

This output must serve two simultaneous purposes: **statistical representativeness, and human realism** by including narrative elements that simulate plausible life stories and personalities. Finally, it can be given completely or partially (only variables needed for a use case) to another LLM during the simulation.

Throughout the development process, the methodology was iteratively refined to address these key questions:

- Can LLMs generate coherent and meaningful human profiles at scale?
- Do they generalize across demographic variability, or do they exhibit sociological bias?

- Are LLMs intrinsically capable of performing synthetic sociology, or must their outputs be controlled?
- Is one LLM sufficient, or is diversity enhanced by using multiple models?

To tackle these challenges, multiple approaches were explored iteratively to identify the most effective methodology:

- **First Approach: Fully Generative Iteration with LLMs**

The initial pipeline involved iterative naive calls to an LLM to generate complete human profiles from scratch. The LLM was simply provided with structural specifications, and in each iteration of the loop it was asked to return a new profile.

While this method produced structurally correct outputs, we observed strong biases and **repetitive patterns across profiles, with a tendency to overrepresent certain personality types**, social backgrounds, and narratives.

There were so many biases that practically 70% of the population had the same job, the same names, and an age range between 20 and 35. The absence of structured control over demographic and psychological diversity made this approach unsuitable for scalable agent generation.

- **Second Approach: Statistical Pre-Profiles + LLM Enrichment**

It is widely known that LLMs have demonstrated strong capabilities in code generation and correction. In fact, it can be observed how when you ask the LLM to fix a code error, it tends to modify only the erroneous parts while preserving the rest. This follows a kind of philosophy: *if the code is already functional, better just change the few things that the user asks me about.*

That philosophy inspired the following methodology: Systematically write JSON pre-profiles that meet statistical distributions on defined key variables, and ask the LLM to complete and improve the profile. In this way, the profiles will not be designed from scratch, and will improve our statistical distributions.

This two-step process ensures both diversity and contextual richness, obtaining a structured JSON that combines factual attributes with narrative elements.

- **Final Approach: Cambridge-Based Population Simulation**

To further ground the simulation in real-world complexity, we used demographic data from Cambridge (Massachusetts) to feed the statistical pre-profiles.

This city was chosen not only because it is the home of MIT, and site of numerous previous studies by the group, but also because of its **unparalleled cultural richness**. Cambridge is home to more immigrants than native-born U.S. citizens, bringing together people with extraordinarily diverse histories, economic backgrounds, cultures and personalities, making it an ideal setting to test and validate our methodology.

So finally, the profiling pipeline is composed of two main phases:

1. **Structured Base Generation:** Key demographic, sociological, economic, and psychological attributes are generated using statistically informed samplers and predictive models.

Design and Development

An example of a generated pre-profile, just prior to the application of generative models, can be found in Appendix C.1.

2. **LLM-Guided Enrichment:** Once the structured pre-profile is defined, it is passed to a LLM, which enriches the data by generating narrative elements, emotional depth, personal history, and inferred behavior patterns.

In Appendix C.2, we can see what a definitive profile looks like.

3.2.2 LLM-Guided Enrichment

Once the initial variables have been generated, they are passed to a local LLM instance using a structured (see Figure 3.2) sequence of prompts:

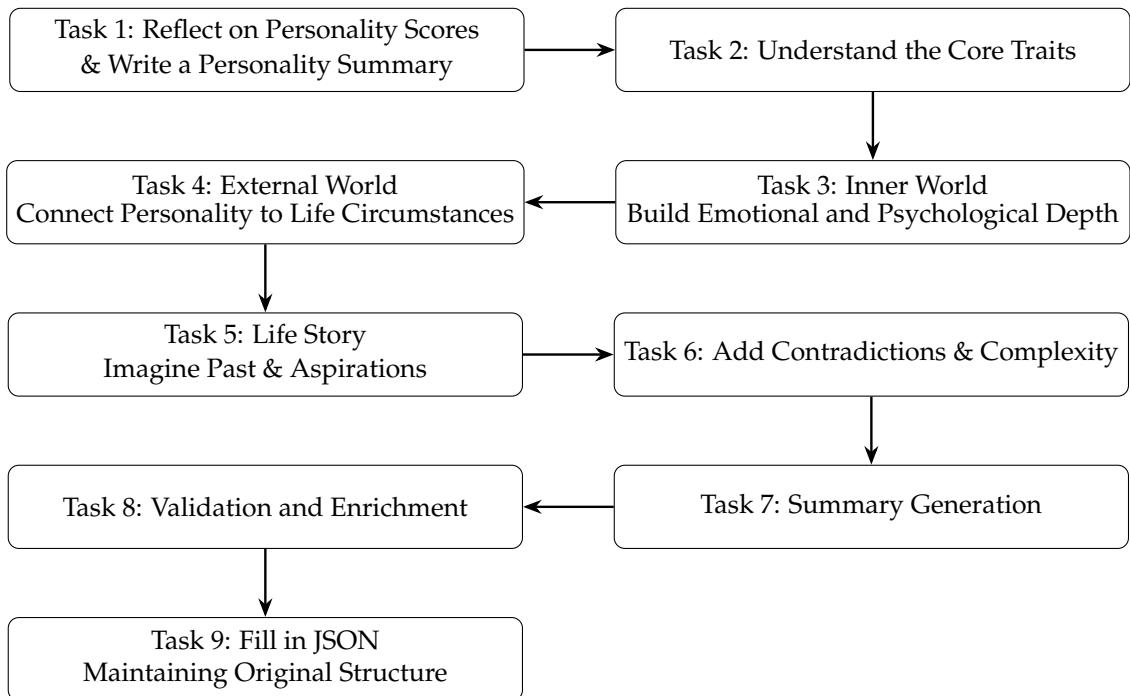


Figure 3.2: LLM-based personality profiling pipeline. Each box represents a call to an LLM to complete a specific task.

- **Task 1 – Personality Summary:** Generate a natural language summary of the agent's personality based on their Big Five scores.
- **Task 2 – Foundations:** Analyze how personality, ideology, religion, and context interact.
- **Task 3 – Inner World:** Speculate on motivations, fears, emotional responses, and introspective tendencies.
- **Task 4 – External World:** Connect internal traits with social behavior, job dynamics, and relationships.
- **Task 5 – Life Story:** Imagine a realistic past and plausible future aspirations.
- **Task 6 – Contradictions:** Identify psychological or contextual contradictions.
- **Task 7 – Summary:** Provide a concise, human-like overview of the individual.

- **Task 8 – Validation:** Reflect on gaps or inconsistencies to be filled in the final profile.

Clear and well-structured prompts were written following the theoretical guidelines studied in Subsection 2.3.1. The entire task chain was developed with a defined output format (JSON), breaking down the task of providing a narrative and humanizing the profile into much more straightforward sub-tasks. This process was carried out iteratively reaching the prompts that gave the best results (the complete prompts can be found in Appendix B).

We realized that if we separated the process into **separate calls, one for each task**, we got much better results from the LLM. If we requested all the tasks at once, it sometimes skipped some or did not pay enough attention to all of them. After performing the aforementioned tasks, it generates the final JSON, having also checked for possible inconsistencies between variables (e.g., a 3-year-old child being a truck driver).

In this form, each reflection is appended to a growing message history and used as input context in the following task. The model is instructed to retain the original schema and to fill every field with specific, consistent, and human-like data.

For the deployment of LLMs, the group relied on its **private server (4xNVIDIA RTX 6000 Ada, 48GB Video Random Access Memory (VRAM) each one)**, utilizing **Ollama** to host the models and manage calls to them. Ollama allows to force the model to generate outputs in JSON format, which was very useful for the generation. The models used to test this pipeline are:

- LLaMA 3.1 (8b)
- Dolphin LLaMA 3 (8b)
- Mistral (7b)
- Qwen 3 (8b)

The selection of these models was made with the aim of **maximizing cultural diversity** in the generation of profiles. By using models developed in different geographical contexts, such as LLaMA (United States), Qwen (Asia) and Mistral (Europe), it was possible to analyze whether there are differences in the narrative style or sociocultural biases present in the profiles generated. In addition, an uncensored model (Dolphin) was included to assess whether the absence of moderation filters allows for richer, more expressive or realistic generation. Many aligned models tend to avoid content considered negative or sensitive, which may limit the representation of certain human traits such as contradictions, flaws or personal conflicts.

Finally, the generated profiles are used as natural language “persona cards” and **they are given to an LLM as structured input context** in the prompt. By organizing the information hierarchically in a tree structure, it becomes possible **to retrieve only the specific data needed** for a given task, instead of the entire JSON.

3.2.3 Complete Profile Architecture

This subsection summarizes the topology followed by the profiles, explaining each of their variables to better understand the generated profile (see Appendix C.2 for a complete profile example).

General Section (the most general profile values):

- **Name:** To avoid duplications, it is generated using the Faker library based on the selected gender.

Design and Development

- **Age:** Sampled from the Cambridge age pyramid, split into gender-specific age brackets to reflect realistic distributions.¹
- **Education:** Derived conditionally based on age. For children and young adults, educational enrollment is estimated using age-based probabilities. For adults, degrees are sampled from local educational attainment distributions.²
- **Place of Residence:** Constant (Cambridge, Massachusetts, USA).
- **General Description:** Generated by the LLM based on personality traits, life events, and background information.

Identity Section:

- **Nationality:** Determined based on Cambridge's foreign-born population statistics. If not U.S.-born, sampled with weighted probabilities by country of origin.³
- **Sexual Orientation:** Modeled using Gallup and Boston Indicators reports. Probabilities vary by age group, reflecting generational differences in LGBTQ+ identification.⁴
- **Gender:** Determined jointly with age during the demographic sampling phase.¹
- **Religious Beliefs:** Sampled from Pew Research distributions for the Boston metropolitan area.⁵
- **Political Ideology:** Sampled based on voting trends in Cambridge's latest presidential election, incorporating probabilities for Democratic, Republican, and minor parties.³

Profession Section

- **Industry:** Sampled using weights derived from the ACS 2022 data for Cambridge's workforce by sector.³
- **Industry Mean Salary:** Retrieved from official statistics associated with each sector.³.
- **Personal Salary:** Estimated via Gaussian sampling around the mean salary of the selected industry, adjusted by a 25% standard deviation.
- **Job:** Completed via LLM based on the agent's industry, education, and salary level.

Psychological and Cognitive Section

- **Big Five Traits:** Personality vectors are assigned using regression models trained on a dataset of over 300,000 individuals (**source: International Personality Item Pool**), taking age, gender, and country as inputs. For each trait (agreeableness, extraversion, openness, conscientiousness, neuroticism), predictions are drawn and correlated through a multivariate normal distribution to preserve realistic covariances. These vectors guide the LLM in generating consistent personality descriptions and behavioral reasoning.
- **General Big Five Description:** Generated by the LLM with a detailed analysis of each trait level, combining psychological insights and behavioral implications.

¹<https://data.census.gov/table/ACSST1Y2022.S0101?g=160XX00US2511000>

²<https://data.census.gov/table/ACSST1Y2023.S1501?g=160XX00US2511000>

³<https://datausa.io/profile/geo/cambridge-ma/>

⁴<https://news.gallup.com/poll/611864/lgbtq-identification.aspx>

⁵<https://www.pewresearch.org/religious-landscape-study/database/metro-area/boston-metro-area/>

- **Cognitive Attributes** (e.g., Emotional Intelligence, Creativity): Inferred and generated by the LLM during reflective tasks based on the individual's profile.
- **Motivations, Strengths, Weaknesses:** Also generated by the LLM using a structured reasoning process that reflects the interaction of personality, life history, and social context.

Behavioral Section

- **Social:**
 - **Family Background, Parental Relationships:** LLM-generated narratives based on inferred upbringing and cultural background.
 - **Friends, Social Networks, Social Skills:** Derived from LLM reflection on extraversion, agreeableness, and educational/professional context.
 - **Marital Status:** Sampled using ACS Cambridge distributions with gender-specific probabilities.⁶.
 - **Workplace Relationships:** Inferred from professional background and personality by the LLM.
 - **Role in Community, Social Class:** Determined via a combination of income, education, and narrative coherence.
- **Habits and Routines:**
 - **Daily Routine, Leisure, Work, Health, and Social Habits:** All generated by the LLM to ensure contextual coherence with the profile. These are aligned with occupational demands, personality traits, and physical status.

Physical, Biological, and Health Section

- **Height and Weight:** Sampled from a Gaussian distribution by gender, using national averages for Massachusetts. Body Mass Index is computed and used to derive the overweight label.⁷.
- **Anatomical Features (Hair, Eye Color, Complexion):** Added by the LLM to enhance realism.
- **Disabilities:** Sampled probabilistically based on real-world disability data by age group and disability type (hearing, vision, cognitive, etc.).⁸.
- **Health Status:** Inferred by the LLM based on other profile elements and lifestyle habits.

3.3 Phase II: Fine-tuning

While the initial results obtained from the profiles were promising, achieving consistent personality expression requires intervention at the model level, as the model inherently carries biases and an “intrinsic personality” of its own.

If we analyze the training process of a conversational model (Figure 2.2), we can see that, after the instruction tuning phase, a preference alignment stage takes place. This last stage allows us

⁶https://data.census.gov/profile/Cambridge_city,_Massachusetts?g=160XX00US2511000#families-and-living-arrangements

⁷<https://worldpopulationreview.com/state-rankings/average-height-by-state>

⁸<https://data.census.gov/table/ACSST1Y2023.S1810?g=160XX00US2511000>

Design and Development

to adjust the way the model responds, giving it a coherent communication style or, in practical terms, a “specific personality”.

Therefore, in this section we will focus on giving the model a specific personality acting precisely in this phase of training. This approach preserves all the benefits and versatility provided by instruction tuning, while modifying the preference alignment to tailor the response style to our requirements.

We adopted a fine-tuning strategy focused on DPO:

- **Base Model:** LLaMA 3 8B-Instruct.
- **Technique:** LoRA-based DPO with 8-bit quantization.
- **Objective:** Enforce consistent personality-aligned generation across contexts.

This strategy was motivated by several considerations drawn from the State of the Art (Chapter 2):

First, for parameter updates (see Subsection 2.3.3), we adopted PEFT since it allows adapting the model with a **significantly lower computational and memory footprint** compared to full fine-tuning, while maintaining task performance. This is even more relevant given the large scale of the base model, where updating all parameters would be prohibitively expensive.

In the PEFT family, we specifically chose LoRA. This method has emerged as the **de facto standard in many applied LLM fine-tuning pipelines** because it obtains a balance between simplicity, effectiveness, and maturity. Low-rank decomposition is conceptually straightforward, introduces minimal architectural changes, and has been widely validated across multiple benchmarks. The extensive availability of open-source implementations, tutorials, and community best practices further supported its selection, making it a robust and practical choice over other PEFT methods (see Subsection 2.3.4) such as adapters, soft prompts, or weight masking, which may require more intrusive integration or exhibit higher sensitivity to hyperparameter choices.

Among the various preference alignment methods reviewed in Section 2.3.5, **we selected DPO**. Unlike other techniques, DPO allows training with simple pairwise preference labels, avoiding the need for complex reward functions or ranked response lists, which are difficult to define in the context of personality alignment.

Pairwise contrastive methods are well suited for our setup. We were able to generate a dataset consisting of pairs of responses labeled simply as “preferred” or “rejected.”. DPO provides a principled way to optimize on these pairwise preferences, with a stable and interpretable loss, avoiding reward hacking risks and sample inefficiency often observed in reinforcement learning approaches. Its formulation is interpretable, and well suited for encoding nuanced preferences such as personality traits.

3.3.1 Personality Evaluation and Selection

The first fundamental aspect of carrying out this training is to define how we are going to evaluate the personality of the LLM responses.

Initially, we explored using a separate **encoder-based personality classifier**, for example, models like bert-base-personality trained to predict Big Five traits from text. These encoder classifiers are designed to map input text directly to a personality trait vector in a supervised setting. This approach was interesting because it allowed us to leverage the strengths of both

architectures: the encoder structure, which is more specialized in understanding and classifying text, and the decoder structure, which excels at generating coherent and stylistically rich responses.

After testing the most popular Hugging Face models⁹¹⁰ in this task we realized, despite their conceptual appeal, these models showed important limitations in practice. In particular, they often produced **incoherent or implausible personality vectors**, failing to capture even explicit self-disclosures in the text (e.g., classifying the sentence “I am very extroverted” with low extraversion scores). This highlighted a lack of robustness and interpretability, and suggested that these models struggle to generalize beyond their relatively narrow training datasets.

Given these shortcomings, we decided to shift to an **LLM-as-a-judge** strategy. In this approach, a second large language model acts as a personality evaluator, receiving the generated text and returning:

- A textual justification for each trait score, grounded in linguistic and semantic cues.
- A Big Five personality vector with continuous scores from 0.0 to 1.0.

This strategy benefits from the same generative capabilities and broad linguistic coverage as the conversational LLM, enabling more consistent and psychologically interpretable assessments across open-ended dialogues. And the reflection, generated before the answer itself, forces inductive thinking and a think step by step strategy.

The evaluation rubric aligned trait levels with specific stylistic and behavioral signals (e.g., very high neuroticism involves anxious or indecisive language). Only structurally correct outputs matching the required schema were retained.

To evaluate how well the generated responses align with the intended Big Five personality traits, we designed a prompting stage (see all the prompt in Appendix D.3) functioning as a lightweight, zero-shot personality classifier. Several design choices contributed to its robustness:

- **Role definition and task clarity.** The prompt defines the role of an *expert in psychological profiling*, priming the evaluator to focus on subtle personality cues in the text.
- **Clear scoring scale with semantic anchors.** The prompt specifies a normalized scoring range (0.00 to 1.00), tied to descriptive categories (*Very Low* to *Very High*) to guide consistent reasoning (see Table 3.2)
- **Trait-by-trait reflection.** The prompt instructs the evaluator to explicitly explain each trait score, describing relevant linguistic evidence. This promotes transparency and guards against arbitrary results.
- **Controlled output format.** A fixed JSON structure with “Reflection” and “Score” keys enables consistent validation, comparison, and automated processing.
- **High interpretability.** Because the output includes rationales, we can qualitatively audit how the evaluator connects language features to personality traits, providing an extra layer of trustworthiness.

Once the evaluation is defined, the second fundamental aspect is to define the target personalities.

⁹<https://huggingface.co/Minej/bert-base-personality>

¹⁰<https://huggingface.co/Nasserelsaman/microsoft-finetuned-personality>

Design and Development

Range	Interpretation
0.00 – 0.19	Very Low
0.20 – 0.39	Low
0.40 – 0.59	Medium
0.60 – 0.79	High
0.80 – 1.00	Very High

Table 3.2: Mapping between numerical trait values and qualitative interpretations

Specifically, we selected four personality vectors, each representing a distinct psychological archetype. These vectors were chosen to span diverse personality configurations, as shown in Table 3.3.1.

Profile	Openness	Conscient.	Extrav.	Agreeab.	Neurot.
<i>Analyst</i>	0.90	0.60	0.35	0.25	0.35
<i>Diplomatic</i>	0.90	0.50	0.45	0.90	0.60
<i>Sentinel</i>	0.25	0.90	0.50	0.65	0.40
<i>Explorer</i>	0.40	0.25	0.80	0.65	0.25

Table 3.3: Target Big Five personality vectors used for conditioned generation

Although our experiments are framed entirely within the Big Five personality model, as discussed in Section 3.1, we needed to define four distinct and easily separable target personalities for training purposes.

To achieve this, we adopted the widely known MBTI typology as a reference, particularly its grouping of the 16 personality types (see Figure 3.3) into four higher-level clusters: **Analysts**, **Diplomats**, **Sentinels**, and **Explorers**. These groups are popular in both academic and applied personality research, and provide clear, interpretable stereotypes.

To map these MBTI-derived clusters into Big Five coordinates, we relied on replicated evidence [173, 174, 175] of correlations between MBTI dimensions and the Big Five.

- Extraversion (Big Five) correlates strongly with MBTI’s E/I dimension,
- Openness to Experience correlates with MBTI’s N/S dimension.
- Agreeableness is moderately related to MBTI’s F/T dimension.
- Conscientiousness shows correspondence with MBTI’s J/P dimension.
- Neuroticism does not have a direct MBTI dimension but has been observed to vary among MBTI types, especially lower in Judging (J) types.

Hence, the Big Five target vectors in Table 3.3.1 reflect a reasonable translation of these four MBTI macro-groups, assigning high Openness to both Analysts and Diplomats (who score high on Intuition, N), high Conscientiousness to Sentinels (J-typed, structured personalities), high Extraversion to Explorers (associated with E and P traits), and high Agreeableness to Diplomats (who emphasize harmony, F types). Neuroticism was tuned to moderate values based on literature showing variability across MBTI types but without strong one-to-one correspondence.

3.3. Phase II: Fine-tuning



Figure 3.3: MBTI 16 personalities. Source: *Online image link*.

Design and Development

In addition to the MBTI-inspired grouping, the chosen Big Five target vectors themselves were carefully selected to maximize diversity across the five dimensions. As shown by the bolded values in Table 3.3.1, these profiles cover both high and low extremes in each Big Five trait. This spread ensures that each of the five Big Five dimensions is represented at different levels, providing a diverse and comprehensive set of conditions.

Consequently, these four vectors offer a well-separated and psychologically interpretable set of target personalities, serving as an ideal case study for testing personality-conditioned generation in large language models. They allow us to explore how the model adapts its linguistic style and behavior under clearly differentiated personality profiles, while still preserving the grounding in well-known psychological frameworks.

3.3.2 Dataset Design for DPO Training

Once the method of assessing personalities and the target vectors have been clarified, the next step is to generate the dataset. Since the group did not have any specific dataset for fine-tuning personalities, it was necessary to generate a synthetic one.

Figure 3.4 shows the main steps followed to generate this dataset, and each of them will be detailed throughout this section. Just like in the previous phase, ollama was used to make calls from the group server.

It is worth highlighting that the same model used to generate the dataset was also employed for fine-tuning, effectively testing whether a model can, in a sense, “**train itself** to develop a consistent personality profile.

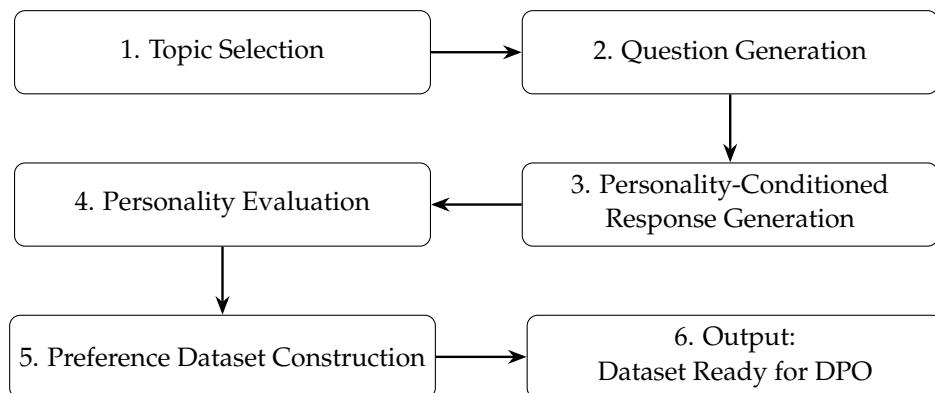


Figure 3.4: Dataset generation pipeline for fine-tuning via DPO

Step 1: Topic Selection

To ensure a diverse and psychologically grounded dataset, an initial list of 200 distinct topics was manually created. These topics span a wide range of personal, emotional, cognitive, social, and behavioral domains. The complete list can be found at Appendix D.1.

They were selected based on psychological literature and introspective heuristics about how personality manifests in real-world decisions, emotions, and interpersonal dynamics. The goal was to create a corpus of prompts that naturally elicit different behavioral traits aligned with the Big Five Personality Model.

Step 2: Question Generation

For each topic, a LLM was prompted to generate approximately 100 open-ended, reflective questions. These questions were designed to provoke responses that reveal behavioral inclinations, emotional regulation strategies, cognitive tendencies, and value systems.

After filtering for uniqueness, clarity, and relevance, the resulting corpus included around **13,000 questions**. An example is shown below:

```
[
  {
    "topic": "Introspection need",
    "question": "Do you believe that engaging in introspection can help you develop greater emotional stability or regulation?"
  },
  {
    "topic": "Friendship values",
    "question": "What are some non-negotiables you expect from your close friends, and how do you communicate these expectations?"
  },
  {
    "topic": "Emotional coping style",
    "question": "How do your emotional coping styles influence your relationships with others?"
  },
  {
    "topic": "Sense of justice",
    "question": "What are some ways in which you think society can be more just, and what role do you see yourself playing in that effort?"
  }
]
```

Step 3: Response Generation with Personality Conditioning

For each question answered, a total of 20 responses were generated (16 personality-conditioned + 4 baseline) simulating the distinct personality archetypes, as defined in Table 3.3.1. The generation process was conducted using a locally hosted LLaMA 3 8B model, accessed via an Ollama API.

For each of the four target personality vectors, we prompted the model to produce four distinct responses. These prompts included:

- The question itself
- An explicit description of the target Big Five vector
- Interpretation scales and behavioral descriptions per trait
- Clear instructions to simulate a human-like, natural answer consistent with the specified personality profile

Each generated response was accompanied by a self-reflective rationale, also generated by the model, explaining how each personality trait influenced the writing style, emotional tone, and content. This ensures transparency and internal coherence.

In addition to the personality-conditioned responses, four baseline responses were generated per question without any conditioning, to capture the native personality drift of the model.

To introduce lexical and semantic variability, four different decoding configurations were applied to each personality vector. These configurations combined **different values of temperature and top-p sampling**:

- Temperature: [0.5, 0.7, 0.8, 0.95]

Design and Development

- Top-p: [0.95, 0.9, 0.95, 0.85]

Moreover, the prompt design for personality-conditioned generation follows best practices (see Subsection 2.3.1) to ensure precision, interpretability, and control. Three main strategies were employed:

- **Explicit input formatting**

Each prompt clearly specifies the input personality vector, with values normalized between 0.0 and 1.0. These scores are accompanied by a human-interpretable scale (*Very Low, Low, Medium, High, Very High*) and detailed behavioral descriptions for each trait. This grounding provides semantic clarity and helps the model align its output with realistic psychological behaviors.

- **Decomposition of task into structured sub-tasks**

Instead of requesting a direct answer, the prompt guides the model through a multi-step reasoning process: first, it is asked to reflect on how each trait affects the response style; only then does it generate the final answer.

1. Analyze the input Big Five vector and interpret each trait individually.
2. Produce a short written explanation of how each trait should influence the response style (reflection).
3. Generate a final, human-like answer simulating a person who matches that profile, without explicitly mentioning the traits.

This structure mirrors inductive chain-of-thought prompting, encouraging internal reflection before generation.

- **Controlled and structured output format**

The model is instructed to return its output in a fixed JSON schema containing:

- The topic and the original question.
- One block per personality vector, each including:
 - * The target vector.
 - * Four answers with their accompanying reflections.
- One base_LLM block with the four non-conditioned answers.

See Appendix D.2 for the full prompt and an answer example.

Step 4: Personality Evaluation

As mentioned in Subsection 3.3.1, we employed a second LLM-based classifier. This evaluator model received the text and returned:

- A Big Five personality vector with continuous scores from 0.0 to 1.0.
- A textual reflection justifying each trait score based on linguistic and semantic cues.

So in this stage, our dataset includes for each question:

- The topic and original question.

- Four personality-conditioned vectors, each with four generated responses and their corresponding evaluated Big Five scores.
- A baseline response set (no conditioning), also with four generated answers and evaluated scores.

Step 5: Preference Dataset Construction

The final stage involves transforming the previously generated and evaluated responses into a high-quality preference dataset for fine-tuning via DPO. The goal is to produce, for each target personality vector, a set of comparative examples in the form of $(prompt, chosen, rejected)$ triplets, where the chosen answer best aligns with the intended personality traits and the rejected answer exhibits a misalignment.

To select the most appropriate (and least appropriate) answers for each prompt, we defined a continuous similarity metric between the target vector \mathbf{x} and the personality inferred by the evaluator LLM, $\hat{\mathbf{x}}$. This similarity function is designed to **penalize large deviations harshly while remaining tolerant of small estimation noise**.

$$S(\mathbf{x}, \hat{\mathbf{x}}) = \left(\prod_{i=1}^d \exp(-\alpha(x_i - \hat{x}_i)^2) \right)^{1/d} = \exp \left(-\frac{\alpha}{d} \sum_{i=1}^d (x_i - \hat{x}_i)^2 \right) \quad (3.1)$$

Where:

- \mathbf{x} : target Big Five vector with $d = 5$ dimensions.
- $\hat{\mathbf{x}}$: inferred vector from the response.
- $\alpha > 0$: penalization factor (set to $\alpha = 5.0$ in our implementation).

This exponentially-scaled similarity it is used because:

- Softens small errors due to noise or ambiguity in LLM-based evaluation.
- Severely penalizes large trait mismatches.
- Produces a similarity score $S \in (0, 1]$ interpretable as a confidence measure of trait alignment.

For each question and each of the four target personality vectors, the system performs the following steps:

1. **Select the best match (chosen):** Among the four responses conditioned on the target vector, we compute their similarity scores with respect to the target using the similarity function $S(\mathbf{x}, \hat{\mathbf{x}})$. The response with the highest score is selected as the chosen answer.
2. **Select the worst mismatch (rejected):** We search across all other responses that were *not* conditioned on the current vector, including those from the base LLM and other personality vectors. The response with the lowest similarity score to the current target vector is selected as the rejected example.
3. **Store metadata:** For both answers, we store the inferred vectors and similarity scores to facilitate further analysis.

Design and Development

This process ensures that the selected chosen answer is the most semantically aligned with the intended psychological profile, while the rejected answer is maximally misaligned within the available candidate pool.

Output Format and Hugging Face Conversion

The output dataset is stored in two formats:

- A custom JSON format that includes the prompt, target vector, chosen and rejected answers, as well as their inferred vectors and similarity scores.
- A Hugging Face-compatible Comma-Separated Values (CSV) format where each sample is structured as:
 - prompt: A list with a single dictionary "role": "user", "content":
 - chosen: A list with the assistant's preferred response.
 - rejected: A list with the less aligned or mismatched response.

Each personality archetype yields its own preference dataset, ready for use in instruction tuning or preference optimization pipelines. This results in four parallel datasets (one per archetype): analyst, diplomat, sentinel, and explorer.

An example row from the final dataset may look as follows (in simplified notation):

```
{  
    "prompt": "Have you ever had to make a compromise between your personal values  
    and the expectations of others?",  
    "target_vector": {"Openness": 0.9, ...},  
    "chosen": "As someone who values exploration and creativity...",  
    "chosen_vector": {"Openness": 0.88, ...},  
    "score_chosen": 0.94,  
    "rejected": "I usually just go with the flow...",  
    "rejected_vector": {"Openness": 0.42, ...},  
    "score_rejected": 0.51  
}
```

In total, we generated human-like responses for 3,000 questions, each conditioned on five distinct personality types. This resulted in a dataset of 60,000 personality-aligned responses.

3.3.3 Training Process

The training process for the fine-tuning was designed to optimize alignment with specific personality traits under the Big Five framework. This subsection describes the complete pipeline, including model preparation, hyperparameter selection, and rationale behind every decision.

The base model selected was **Meta-Llama-3-8B-Instruct**, due to its high performance on instruction-following tasks and public availability. To reduce memory footprint and enable training on limited hardware resources, the model was loaded using 8-bit quantization via the bitsandbytes library. This choice offers a favorable trade-off between memory efficiency and computational precision, especially since gradient computation is carried out in higher precision (bfloating16).

All prompts were formatted using the official llama-3-instruct.jinja chat template, which includes <| start_header_id |> and <| eot_id |> delimiters. This ensured full compatibility with LLaMA 3's conversational prompt expectations, reducing prompt injection issues and boosting stability during generation.

As we can see in Table 3.4, the entire training pipeline was built using widely adopted open-source libraries from the Hugging Face and PyTorch ecosystems, with additional tools to optimize memory usage and experiment tracking.

Framework / Tool	Purpose
Transformers	Interface for pretrained LLaMA 3 models, tokenization management, and prompt formatting using the official chat template.
Datasets	Data loading and preprocessing of JSON files with efficient tokenization and multiprocessing support.
PEFT Library	LoRA adapter integration for parameter-efficient fine-tuning with reduced trainable parameters.
Transformers Reinforcement Learning (TRL) Library	Implementation of DPO through DPOTrainer for alignment tuning based on preferences.
BitsAndBytes	8-bit quantization support and memory optimization through mixed-precision (bf16) training.
Accelerate	Device placement management and scalable training orchestration.
Wandb	Experiment tracking, hyperparameter logging, and visualization of training metrics.
Hugging Face Hub	Model versioning, checkpoint storage, and deployment preparation.
Ollama API	Local hosting of the base LLaMA 3 8B-Instruct model for efficient data generation.

Table 3.4: Technologies and frameworks used in the training pipeline

LoRA Configuration

To enable efficient fine-tuning, we adopted LoRA approach via the PEFT library. The configuration used was:

- $r = 32$: A moderately high rank to ensure the adapter matrices could capture stylistic variations aligned with personality.
- $lora_alpha = 64$: Set to scale the updates adequately without destabilizing training.
- $lora_dropout = 0.05$: Introduced slight regularization to prevent overfitting.
- $target_modules$: Included all key projection and feedforward modules (k_proj, q_proj, v_proj, etc.) to maximize expressivity.

This configuration strikes a balance between parameter efficiency and model capacity to internalize nuanced preferences related to psychological traits.

DPO Configuration

The core of the training process leveraged the *DPOTrainer* from the TRL library. The configuration used was:

Design and Development

- $\max_steps = 1000$: Increased to allow prolonged adaptation while monitoring for overfitting with the evaluations set.
- $learning_rate = 2e - 5$: A conservative value to enable gradual fine-tuning without catastrophic forgetting.
- $\beta = 0.1$ and 0.01 : We experimented with different values of the inverse temperature parameter β . Lower values of β increase the model's sensitivity to preference differences, allowing greater behavioral flexibility and adaptation. This setting was particularly useful for encouraging the model to depart from its default style and adopt personality-aligned responses more consistently.
- $gradient_accumulation_steps = 4$, $per_device_train_batch_size = 6$: Configured for memory-constrained training while achieving effective batch sizes of 24.
- $warmup_steps = 50$, $lr_scheduler = cosine$: Warmup and cosine decay helped stabilize early learning dynamics and prevent sharp degradation later.
- $eval_strategy = epoch$, $save_strategy = steps$ with $save_steps = 50$: Enabled frequent checkpointing to monitor evaluation loss throughout the training trajectory.

The reference model used during DPO was the same pre-trained LLaMA-3 model loaded with the same quantization settings. This enabled meaningful comparisons in log-probability calculations across preference pairs.

If we examine the DPO objective function (2.16), we can clearly see the **importance of the β parameter**. I want to emphasize this aspect, as its selection plays a crucial role in the training process. A lower β allows the model to deviate more significantly from the base model, facilitating the emergence of a new, distinct personality. However, if β becomes too small, it may lead to overfitting and excessive divergence from the pretrained behavior.

Validation and Fusion

Finally, a validation split comprising 10% of the dataset was used to monitor training progress. Model checkpoints were saved every 50 steps, and the checkpoint with the lowest evaluation loss was selected as the final model. After training, the LoRA adapter weights were merged into the base model and the final model was saved in Hugging Face format for downstream generation and deployment.

Chapter 4

Experiments and Results

This chapter presents the results obtained in both phases of the study and discusses the main conclusions drawn from them.

4.1 Evaluation of Phase I: Profile Generation

During this stage, profiles were generated following the pipeline described in the previous section 3.2. A total of 1,118 profiles were generated, which is equivalent to a 1:100 scale of the actual population (approximately 118,000 citizens). For this purpose, the following models were used:

- LLaMA 3.1 (8b)
- Dolphin LLaMA 3 (8b)
- Mistral (7b)
- Qwen 3 (8b)

4.1.1 Similarity

As discussed in the aforementioned section 3.2, the first naive approach adopted generated profiles with highly repetitive values, which prevented us from obtaining a sufficiently diverse population. After implementing the final pipeline, we evaluated the degree of differentiation among the generated profiles and confirmed that all profiles were mutually distinct.

To evaluate the similarity between the generated profiles, several metrics were used:

- **Embedding Similarity (Embedding-based):** To measure semantic similarity between profile texts, we use cosine similarity between sentence embeddings (see Subsection 2.1). Given two embedding vectors \vec{a} and \vec{b} :

$$\text{Cosine Similarity}(\vec{a}, \vec{b}) = \frac{\vec{a} \cdot \vec{b}}{\|\vec{a}\| \|\vec{b}\|}. \quad (4.1)$$

Where:

- $\vec{a} \cdot \vec{b}$: Dot product of the two vectors.

4.1. Evaluation of Phase I: Profile Generation

- $\|\vec{a}\|$: Euclidean norm (magnitude) of vector \vec{a} .
- $\|\vec{b}\|$: Euclidean norm (magnitude) of vector \vec{b} .

The average cosine similarity across all profile pairs gives an overall measure of semantic redundancy or coherence within the generated set.

- **Distinct-1 (Lexical Diversity):** This metric measures the ratio of unique unigrams (words) to the total number of unigrams across all profiles. It captures how lexically varied the text is:

$$\text{Distinct-1} = \frac{|\text{Unique Unigrams}|}{|\text{Total Unigrams}|}. \quad (4.2)$$

- **Shannon Entropy (Global Diversity):** Entropy quantifies the uncertainty or diversity of attribute values in the profiles, treating the distribution of all attribute values as a probability distribution:

$$H(X) = - \sum_{i=1}^n \frac{f_i}{N} \log_2 \left(\frac{f_i}{N} \right). \quad (4.3)$$

Where:

- x_i : Unique attribute values across all profiles.
- f_i : Frequency of value x_i .
- N : Total number of attribute values observed.

In Table 4.1, several things can be observed:

Metric	LLaMA	Dolphin	Mistral	Qwen
Unique Profiles	100%	100%	100%	100%
Profiles Cosine Similarity	0.6677	0.7117	0.7450	0.7229
General Description Cosine Similarity	0.3248	0.2936	0.4243	0.4969
Big Five Description Cosine Similarity	0.4210	0.4705	0.4716	0.5595
Distinct-1 Profiles	0.04	0.06	0.05	0.03
Distinct-1 General Description	0.09	0.09	0.09	0.08
Distinct-1 Big Five Description	0.06	0.09	0.07	0.05
Global Entropy	13.60	12.88	12.56	13.10

Table 4.1: Similarity metrics across LLM-generated profiles

The first thing we can observe in the table is that there are no profiles that are exactly the same. To analyze the cosine similarities between profiles each JSON profile was converted into a full text string, on which embedding techniques¹ and cosine similarity calculation were applied.

However, since the JSON structure is fixed and can artificially induce higher similarity, two key elements of the profile were also extracted and analyzed: the general description and the personality description. These sections, being freer and generated with greater variability, have significantly lower similarity values, which better reflects the diversity between profiles.

¹<https://huggingface.co/sentence-transformers/all-mnlp-base-v2>

Experiments and Results

When comparing the results between models, it is observed that **LLaMA** is the one that achieves the **highest differentiation** (lower average similarity), followed by **Dolphin**, which is also based on LLaMA. This result is consistent, since during profile construction, LLaMA tends to generate more extensive and detailed content, considerably enriching the base information. On the contrary, **Qwen** is the model that contributes the **least in terms of enrichment**, resulting in profiles that are more concise and, therefore, more similar to each other.

Second, we note that Qwen also has the lowest score in lexical diversity, understood as the proportion of unique words to the total number of words generated. In this metric, **Dolphin** and **Mistral** stand out, presenting a greater variety of vocabulary. However, it is important to note that both models tend to **generate shorter and more summarized profiles**, and in some cases even omit variables. For this reason, lexical diversity alone is not enough to adequately assess the richness of the profiles, and it is necessary to complement it with metrics such as cosine similarity (as has been done in this analysis) to obtain a more complete view of the real diversity of the contents generated by each model.

Finally, when analyzing the global entropy of the profiles, we observe that LLaMA and Qwen obtain the highest values, indicating a greater informative diversity in their outputs. If we combine this result with the previously discussed metrics, cosine similarity and lexical diversity, we can conclude that **LLaMA is the model that generates the most diverse and enriched profiles**. On the other hand, **Qwen**, although it contributes less in terms of semantic richness (embedding), **maintains a good general structure and produces differentiated values**. In contrast, **Dolphin and Mistral tend to be more vague and concise** in their responses, sometimes omitting information, but still achieving a greater lexical variety than Qwen.

4.1.2 Demographic Variables Analysis

This subsection analyzes the quality of population generation in terms of demographic variables, which have been designed to follow specific distributions. The main metric used for this evaluation is the Mean Absolute Error (MAE), which allows us to quantify the degree of deviation from the reference distributions:

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|. \quad (4.4)$$

Through this metric, we can identify the percentage of errors in categorical variables, for example, incorrect category assignments, as well as the variation in numerical variables with respect to their expected values.

It is important to note that models such as LLaMA not only reproduce the expected distributions, but also tend to enrich and correct the generated variables. For example, if a profile of a Greek and Christian person is generated, the model may refine the category to "Orthodox Christian of the Greek Church". Likewise, if an occupation and salary are erroneously assigned to a three-year-old child, the model is able to detect and correct this inconsistency.

Thus, although the MAE provides a good approximation of the fit between the generated profiles and the actual distributions, a slight deviation in this metric may be due to semantically beneficial corrections, rather than actual errors, which reinforces the qualitative value of certain models.

First, Figure 4.1 shows the population pyramids generated by the different models. It can

4.1. Evaluation of Phase I: Profile Generation

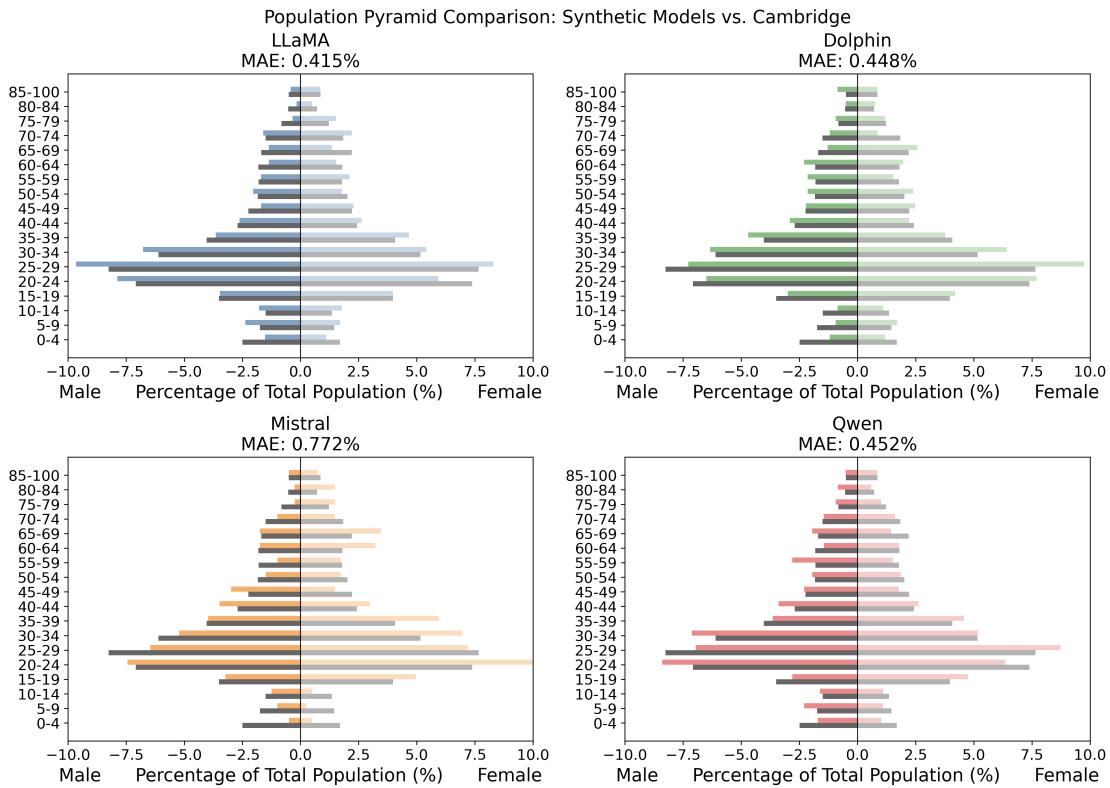


Figure 4.1: Population pyramids

be seen that the four models fit the real distribution² very accurately (shown in gray), **all of them with an error of less than 1%**. These are very reasonable results, reflecting a good fit to the expected demographic structure. LLaMA stands out especially, obtaining the best result with an MAE of only 0.415%, which evidences its excellent capacity to replicate the target population.

Similarly, Figure 4.2 shows the distribution of nationalities. Once again, **all models maintain an error below 1%**, demonstrating good fidelity with respect to the reference distribution.³ Again, LLaMA stands out with an MAE of 0.347%, consolidating as the model that best follows the expected distribution.

If we look at Figure 4.3, we can highlight several aspects. The lower part shows the distribution of work industries within the population³, the most representative in Cambridge being the education, transportation, arts, science and health sectors. All the models manage to fit this distribution correctly, with an error of **1% or less**. The best performing model is Qwen, with an MAE of 0.28%, while LLaMA has the largest error at 1%. However, this slightly higher margin is due to the tendency of LLaMA to enrich this category with additional specific information in each profile, which may introduce minor deviations from the predefined categories, but improves the narrative coherence of the generated profile.

Salaries are generated using a Gaussian distribution centered on the mean salary for each sector. Because of this approach, some sectors with very low representation such as agriculture, which may appear only once or twice in the synthetic populations, might have high variance

²<https://data.census.gov/table/ACSST1Y2022.S0101?g=160XX00US2511000>

³<https://datausa.io/profile/geo/cambridge-ma/>

Experiments and Results

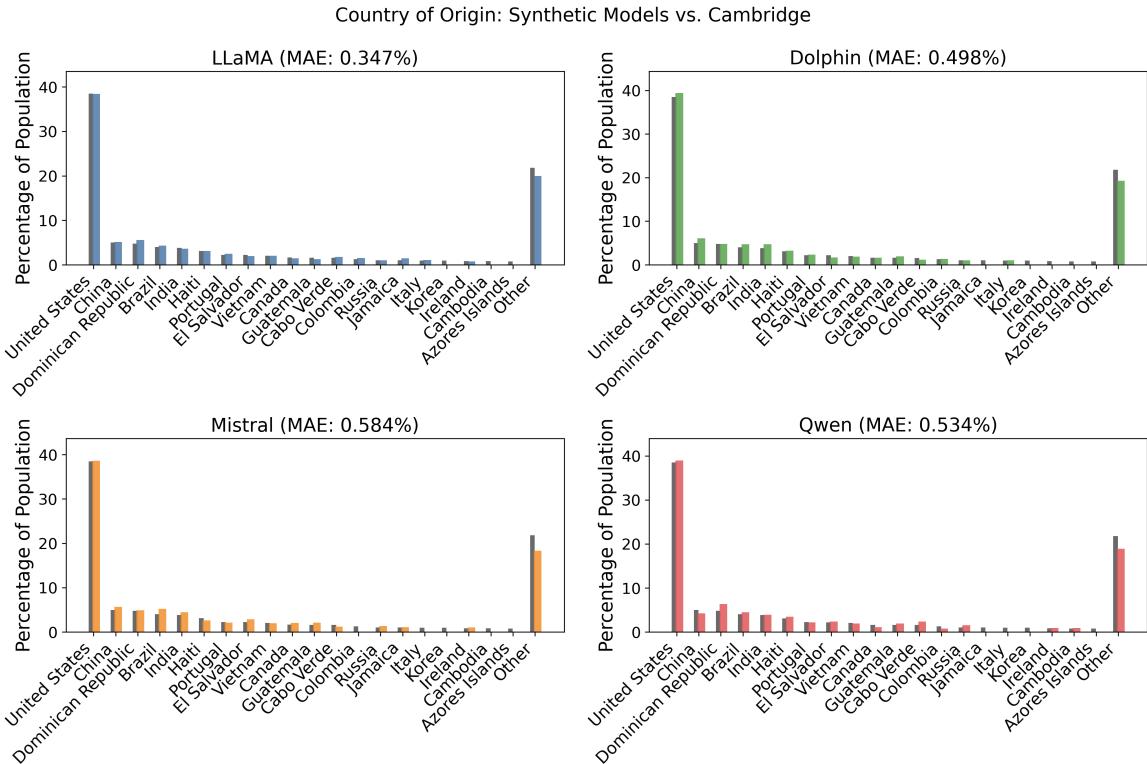


Figure 4.2: Country of origin

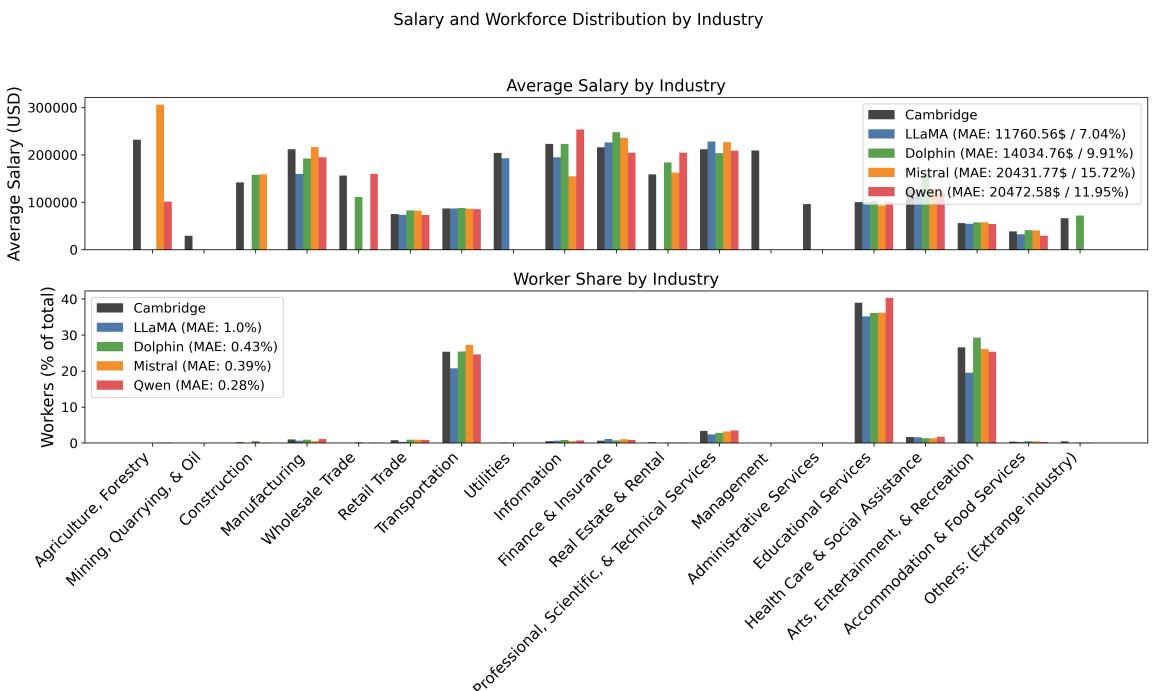


Figure 4.3: Work by industry

4.1. Evaluation of Phase I: Profile Generation

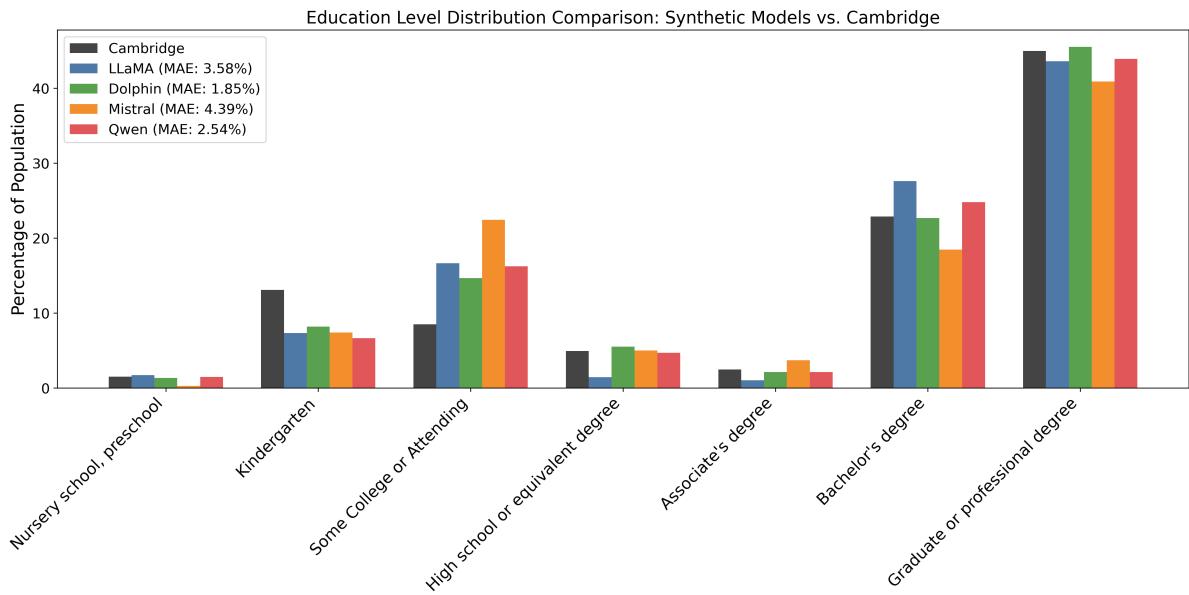


Figure 4.4: Education

or disproportionate apparent errors. This influences the calculation of the overall mean error, increasing it artificially. However, if we focus on the sectors with the highest number of occurrences, we observe that all the models adequately reproduce the distribution of average salaries by sector, remaining faithful to statistical expectations in the most relevant cases.

Regarding Figure 4.4 corresponding to education⁴, it can be seen that the models fit the different educational categories adequately. However, the category “some college or attending” tends to be overrepresented, as many profiles include college attendance in their personal history (especially relevant in a city like Cambridge), even though they did not complete their studies in the end. In these cases, the models tend to classify individuals within this category, which partly justifies their higher frequency. In addition, it has been observed that for very early ages, the models tend to eliminate or adjust the educational variable, indicating that the individual has not yet begun his or her education, which results in an underrepresentation of the “kindergarten” category. When analyzing performance by model, it is observed that Qwen and Dolphin fit this classification more accurately, while Mistral and LLaMA present a greater margin of error in the distribution of educational levels.

One striking aspect is that, when enriching university studies generated in the Cambridge area, the models **tend to assign iconic institutions such as MIT or Harvard**, depending on the area (technical or not), and to a lesser extent Boston University. This reveals a certain tendency to cliché, ignoring the great diversity of existing universities in the Boston area, which reflects both a cultural bias and a preference for recognized entities.

⁴<https://data.census.gov/table/ACSST1Y2023.S1501?g=160XX00US2511000>

Experiments and Results

Table 4.2 presents the remaining variables with a predefined distribution.⁵⁶⁷⁸⁹ In general, it can be observed that all the variables present an error of less than 3%, which indicates a good representation of the target population.

Trait	LLaMA	Dolphin	Mistral	Qwen
Sexual Orientation	0.85%	0.71%	0.27%	0.64%
Religion	2.38%	1.15%	1.85%	1.23%
Ideology	1.02%	1.64%	0.44%	0.52%
Marital Status	4.73%	13.2%	3.95%	1.08%
Disabilities	2.61%	1.96%	2.76%	2.36%

Table 4.2: Mean absolute error in categoric demographics

In fields particularly relevant to personal identity, such as religion or ideology, the models tend to be more creative, especially when the original value is “don’t know” or “nothing in particular”. In such cases, models tend to generate a more elaborate description to justify the lack of explicit information.

Similarly, in the marital status variable, it is common for models to extend the response by describing the current love dynamic (e.g., “lives with partner for three years, although they are not married”), which slightly increases the numerical error but semantically enriches the profile. In contrast, Dolphin tends to summarize or eliminate in this category in typical cases such as “married”, which reduces variability and generates a larger error with respect to the original distribution.

As for the disability variable, the models tend to add plausible conditions according to the profile, such as arthritis in the elderly, thus enriching the realism of the profile, although this causes some deviation from the original value, which in most cases was “no disability”.

Comparing between models, **Qwen is generally the one that best preserves the original distributions**, obtaining very good scores. In contrast, **the characteristic verbosity of LLaMA** tends to slightly increase the error, although it tends to do so in exchange for improving the narrative richness of the profile.

In conclusion, the results show that the models fit the actual distributions reasonably well when the proposed methodology is followed. Despite slight deviations in some variables (in many cases attributable to semantic enrichments or consistent corrections by the model) the overall representation of the population is robust. This suggests that the models are not only capable of replicating statistical patterns, but also of generating more realistic and nuanced profiles, which reinforces their usefulness for the generation of synthetic populations with analytical and narrative value.

⁵<https://news.gallup.com/poll/611864/lgbtq-identification.aspx>

⁶<https://www.pewresearch.org/religious-landscape-study/database/metro-area/boston-metro-area/>

⁷<https://datausa.io/profile/geo/cambridge-ma/>

⁸https://data.census.gov/profile/Cambridge_city,_Massachusetts?g=160XX00US2511000#families-and-living-arrangements

⁹<https://data.census.gov/table/ACSST1Y2023.S1810?g=160XX00US2511000>

4.1.3 LLM-generated Variables

Tables 4.3 and 4.4 represent the 20 most repeated words of each model in the general and personality descriptions (summarizing the profiles of the individuals), the models show clearly differentiated narrative styles. LLaMA generates **emotionally rich and highly idealized profiles**, highlighting attributes such as passion, ambition, creativity and social commitment. The narrative focuses on individuals who are proactive, empathetic and community-minded. In contrast, **Dolphin adopts a much more structured and factual tone**, with an emphasis on academic, professional and demographic data. Its style is colder and more biographical, with little room for emotional attributes, highlighting the uncensoring process it has undergone with respect to LLaMA.

Rank	LLAMA	Dolphin	Mistral	Qwen
1	individual	cambridge	cambridge	cambridge
2	passion	massachusetts	massachusetts	massachusetts
3	driven	degree	degree	balancing
4	ambitious	professional	industry	living
5	strong	graduate	professional	cultural
6	has	holds	graduate	community
7	creative	industry	living	degree
8	bright	usa	residing	graduate
9	sense	male	usa	professional
10	social	resides	services	ma
11	justice	individual	male	woman
12	education	female	woman	heritage
13	warm	works	educational	passion
14	community	residing	transportation	student
15	values	has	bachelors	navigating
16	curious	values	currently	personal
17	loves	bachelors	female	life
18	learning	services	arts	education
19	compassionate	ambitious	holds	residing
20	always	woman	unemployed	creative

Table 4.3: Top 20 most frequent words (excluding common pronouns and auxiliaries) by model in General Descriptions.

Mistral presents a similar approach to Dolphin but incorporating some personal nuances and identity values such as religion or ideology. This adds a little more narrative depth, although an institutional vision continues to predominate. **Qwen combines biographical information with a strong cultural, relational and emotional component**. It highlights values such as identity, community and aspirations, building profiles that reflect internal processes and a sense of belonging, but without idealizing as LLaMA does, but rather with more normal profiles.

As for the descriptions associated with the Big Five traits, there are also clear contrasts. LLaMA maintains its positive tone, highlighting prosocial attributes such as empathy, creativity and openness. The language is emotional and oriented toward personal growth. Dolphin, on the other hand, offers a more technical and analytical view, focusing on trait names and level, with an evaluative rather than narrative tone.

Mistral balances technical terms with qualitative adjectives, trying to offer nuanced rather than extreme profiles. Attempts at moderation and balance are apparent in her descriptions. Finally, Qwen takes a more introspective approach, including both strengths and vulnerabilities. He uses terms that reflect internal conflicts and adaptation to the environment, making him the model that comes closest to a complex and human representation of personality.

Experiments and Results

Rank	LLaMA	Dolphin	Mistral	Qwen
1	values	openness	conscientious	individual
2	individual	individual	agreeable	strong
3	highly	agreeable	individual	highly
4	empathetic	conscientious	new	curious
5	person	extraversion	experiences	sense
6	social	high	open	creative
7	strong	new	openness	new
8	also	conscientiousness	openminded	values
9	outgoing	openminded	moderate	structure
10	relationships	agreeableness	moderately	moderate
11	openminded	while	neuroticism	thoughtful
12	creativity	levels	easygoing	organized
13	sense	experiences	strong	socially
14	creative	highly	high	ideas
15	agreeable	exhibits	outgoing	selfdoubt
16	responsibility	balance	level	struggles
17	can	disciplined	conscientiousness	openness
18	new	extraverted	values	creativity
19	warm	displays	can	balances
20	harmony	strong	has	often

Table 4.4: Top 20 most frequent words(excluding common pronouns and auxiliaries) by model in Big Five descriptions.

When observing the mean emotional aspects rather than only the most frequent words in Figure 4.5 (the mean of the feelings obtained with the *Roberta GO Emotions*¹⁰ model), we can see how in the general descriptions **LLaMA is the model that most idealizes the person, having a mean of 0.7 in admiration**, the rest of the models present a notably lower admiration, and we can see how **their highest value is the neutral variable**. In addition, the variable approval is also notably high in all models.

In the descriptions of personalities the average admiration and approval of the models increase (which seems logical when speaking at a personal level and not only at a general level) but keeping the neutral variable reasonably high. Significant scores also appear in “approval”, ‘curiosity’, and “joy”, especially in Qwen (who excels in curiosity), which may indicate a greater focus on internal exploration.

Negative emotions such as anger, disgust, fear, sadness, remorse, etc., are virtually nonexistent in all models, which is consistent with the idealized tone of the descriptions.

This trend is even clearer in Figure 4.6, where it can be seen that negative emotions are practically nonexistent in both types of descriptions. This is a relevant aspect, **since real human profiles also include negative traits, contradictions and vulnerabilities**. The absence of these elements limits the authenticity of the generated representations. Consistently, LLaMA stands out for generating the highest levels of positive emotions in both cases, while emotional neutrality is higher in general descriptions than in sentiment descriptions.

If we turn our attention to more specific LLM-generated variables in the Table 4.5, we can see that Qwen and LLaMA generate more distinct job titles (930 and 912), while Mistral has the highest semantic diversity (0.7112), indicating greater conceptual variety. The most frequent jobs vary between models, but tend to focus on the educational and artistic fields, with profiles such as Student, Teacher, Transportation Planner or Freelance Artist.

¹⁰https://huggingface.co/SamLowe/roberta-base-go_emotions

4.1. Evaluation of Phase I: Profile Generation

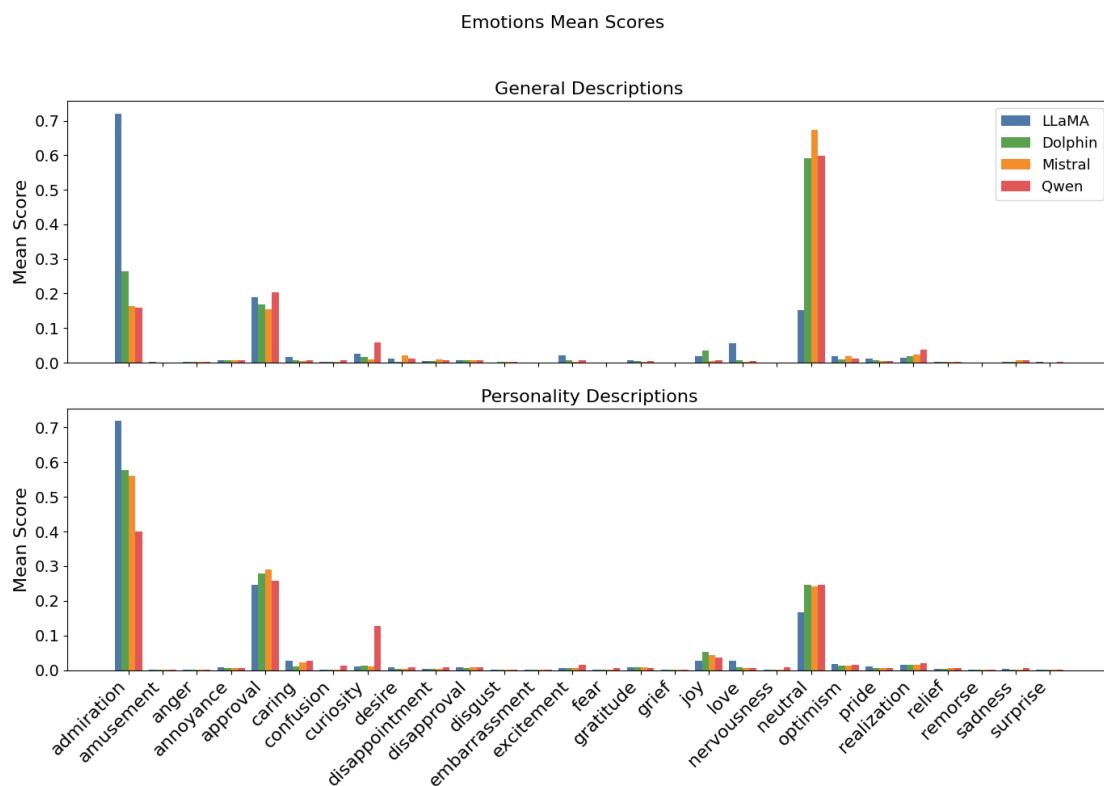


Figure 4.5: Emotions comparison

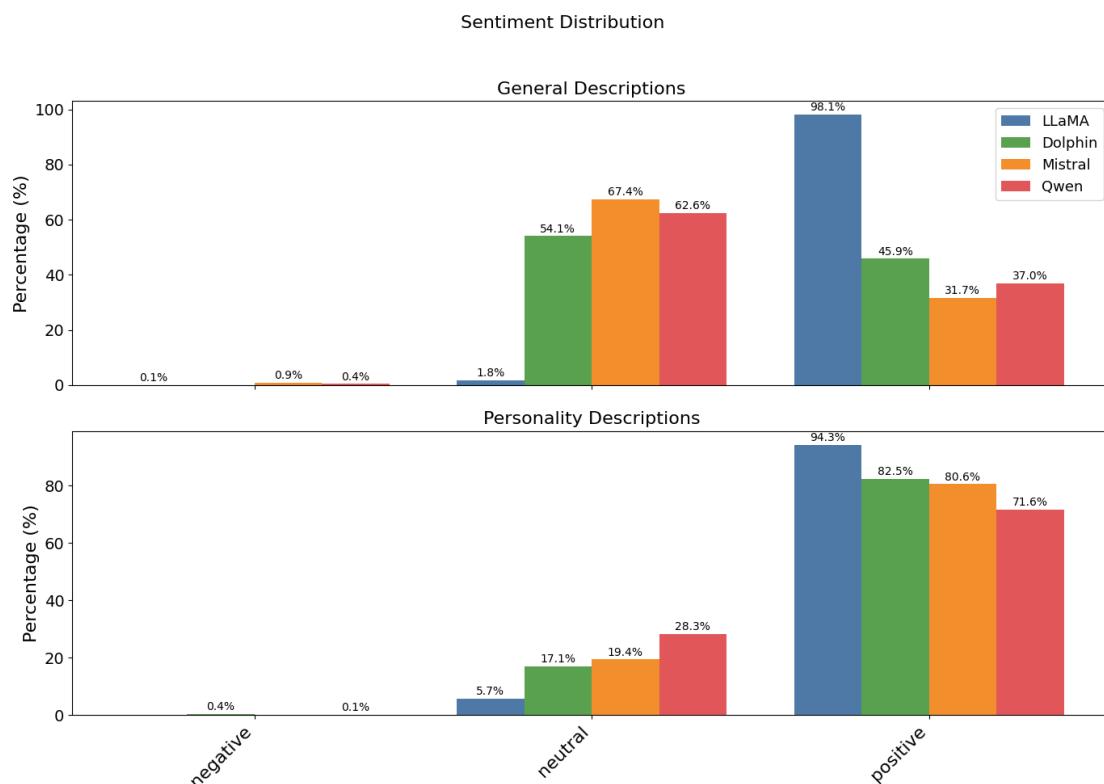


Figure 4.6: Sentiment comparison

Experiments and Results

Trait	Metric	LLaMA	Dolphin	Mistral	Owen
Job	Distinct Values	912 0.6755	660 0.6527	876 0.7112	930 0.7048
	Semantic Diversity	Student Senior Transportation Engineer Harvard Professor High School Teacher Freelance Artist	Educational Consultant Art Director Transportation Manager Creative Director Director of Educational Programs	Elementary School Teacher Graphic Designer Transportation Engineer Painter Special Education Teacher	Student Transportation Planner Curriculum Developer Retired School Administrator Urban Transportation Planner
	Top-5 Values	937	146	1174	1180
	Distinct Values	0.5278	0.5839	0.5224	0.5363
Life Goals	Semantic Diversity	Travel Learn New Languages Experience Different Cultures Get married and have children	Work-life Balance Personal Growth Maintain a Sense of Purpose Fulfillment Through Spiritual Faith	Travel Start a Family Quality Time with beloved ones Stay Physically and Mentally Active	Grow and Learn Work-life Balance Engage in Community Projects Raise a Family in a Safe Sustainable Environment
	Top-5 Values	Spend Quality Time With Loved Ones	Personal Achievements	Maintain Strong Relationships	
	Distinct Values	465	144	941	1625
	Semantic Diversity	Creativity Empathy Communication Skills Emotional Intelligence Creative Problem-solving	Interpersonal Skills Empathy Leadership Creative Problem-solving Effective Communicator	Strong Work Ethic Creativity High Emotional Intelligence Empathy Emotional Intelligence	Organizational Skills Interpersonal Skills Creative Problem-solving Analytical Skills Commitment to Social Justice
Strengths	Top-5 Values	543	128	1140	1701
	Distinct Values	543	128	1140	1701
	Semantic Diversity	0.6521 Perfectionism Overthinking Self-doubt Impulsiveness Procrastination	0.6730 Low emotional skills Creative-practical conflict Poor time management Rigid vs. open-minded conflict Disorganized and competitive	0.6737 Difficulty Delegating Tasks Take on too much Responsibility Worry Excessively Overly Self-critical	0.6591 Difficulty Delegating Tasks Tendency to Overcommit Struggles with Self-promotion Hesitant to Take Risks perfectionism
	Top-5 Values	1148	196	1124	1174
Weaknesses	Distinct Values	0.5967	0.5499	0.5095	0.4864
	Semantic Diversity	Exercise Meditation Reading 6:00 Wake Up Family Breakfast	Structured Prioritizing Work and Personal Growth Prioritizing Schoolwork and Personal Growth Balanced Routine Well-being	Wakes Up Early Exercise Meditate Read the News Working on Community Projects	wakes up early Reviews Transportation data Reading Planning Community Projects Spending Time With Family
	Top-5 Values				
	Distinct Values				

Table 4.5: Diversity and frequency analysis of LLM generated variables

If we look at the rest of the variables, we observe that Qwen and Mistral generate a high variety of different values, in some cases almost unique per profile (although, since they are lists, they can be repeated between profiles). LLaMA, although it shows some occasional drops, maintains a relatively solid diversity. Dolphin, on the other hand, stands out for a remarkably low number of distinct values, which limits its ability to differentiate between profiles.

The Life Goals variable shows clear differences between models, revealing cultural biases. LLaMA and Dolphin, trained in the United States, **prioritize individual goals** such as Travel, Experience Different Cultures or Personal Achievements. Mistral, trained in Europe, also emphasizes travel as the main goal. In contrast, **Qwen, trained in China, omits travel** and other cultures among its most common values, focusing on meaningful contribution to society or family life in safe environments, also, if we go beyond the top five, we observe even more related to heritage, giving back to the community, sustainability, etc.

With respect to strengths, we again observe marked differences between models. **Qwen frequently includes concepts such as commitment to social justice** and other strengths focused on collective well-being, showing a clear orientation towards community reinforcement. On the other hand, Dolphin and LLaMA tend to highlight **more individual and professional skills such as Communication Skills, Creativity or Leadership** among their most common values. **Mistral focuses on more human and relational qualities**, with a notable weight of attributes such as ethics, empathy and emotional intelligence.

In terms of weaknesses, LLaMA, Mistral and Qwen tend to generate “job interview” type responses, with traits such as perfectionism, difficulty delegating tasks, overly self-critical or tendency to overcommit, which tend to mask weaknesses in a positive light. In contrast, **Dolphin, being an uncensored model, presents more realistic and direct weaknesses**, such as poor time management, disorganized, competitive or low emotional skills, offering a cruder and less filtered profile.

The variable daily routine, in all models **tend to describe overly idealized lifestyles**: they include regular exercise, meditation, waking up very early, and quality time with family. These descriptions reflect a normative view of well-being, but leave out more realistic or imperfect routines, limiting the diversity and authenticity of the profiles generated.

4.2 Evaluation of Phase II: Fine Tuning

In this second experiment, LLaMA 3 8B model was fine-tuned using four different datasets and two configurations per dataset, resulting in a total of eight independent trainings of the language model. The entire process followed the pipeline described in Section 3.3.

The four datasets used (Analyst, Diplomat, Explorer, and Sentinel) were constructed based on synthetic personality profiles aligned with the Big Five model. Each dataset represents a distinct personality archetype, defined by a target vector of trait values. These datasets consist of pairs of responses to the same prompt: one response judged as preferred (i.e., better aligned with the target personality) and one as less preferred. The pairs were automatically generated using an LLM-as-a-judge pipeline (see Subsection 3.3.2), enabling scalable construction of preference-aligned training data for the DPO fine-tuning process.

Table 4.6 shows the final metrics obtained after training. To evaluate the impact of fine-tuning, we applied three metrics to the model outputs:

- **MAE:** This metric quantifies the average absolute deviation between the predicted per-

Experiments and Results

sonality vector and the intended target. It provides a straightforward indication of how close the model’s output is on a trait-by-trait basis.

- **Exponential Similarity Function** (see Equation 3.1): This custom similarity metric is specifically designed for evaluating psychological alignment with LLMs. It penalizes large mismatches more severely than small ones and returns a bounded score $S \in (0, 1]$ that acts as a soft confidence score. This makes it particularly suitable for comparing personality differences and prioritizing semantically aligned answers in the DPO dataset construction.
- **Euclidean Distance**: This global geometric measure reflects how far the full personality vector is from the target configuration in \mathbb{R}^5 .

Each model was prompted with 100 personality-revealing questions, and their answers were analyzed using the *LLM-as-a-judge* evaluation approach described in Section 3.3. Improvements over the base model are indicated in green, while deteriorations are marked in red.

It is observed that, under the configuration with $\beta = 0.1$, all models improve their metrics. It is essential to **highlight the importance of the β parameter in the training process**. As seen in the DPO formula (see Equation 2.16), this parameter explicitly controls the degree of deviation that the trained model is allowed to deviate from the original model. In other words, β acts as a regulator of the flexibility of the fit. High values of β restrict changes, preserving the structure of the base model, while too small values may induce an overfitting to the set of preferences, leading the model to forget its prior knowledge. Therefore, a **correct calibration of β is essential to achieve effective personality model customization** without compromising its overall consistency and performance.

Among the results the **Diplomat model stands out**, achieving the highest similarity (0.813), as well as the lowest error and distance. It should be noted that the Diplomat dataset is which best reproduces the target profile being the most complete and stable dataset.

The Sentinel model ranks second in overall performance, also achieving the greatest relative improvement over the baseline in all metrics. As for the other two models, Explorer shows a more notable improvement over baseline, while Analyst achieves superior final metrics, albeit with a smaller relative improvement.

Model		MAE	Similarity	Euclidean
Analyst	$\beta = 0.1$	0.223 (0.026)	0.665 (0.033)	0.636 (0.04)
	$\beta = 0.01$	0.215 (0.034)	0.639 (0.007)	0.668 (0.008)
Diplomat	$\beta = 0.1$	0.164 (0.038)	0.813 (0.065)	0.444 (0.088)
	$\beta = 0.01$	0.140 (0.062)	0.847 (0.099)	0.398 (0.134)
Explorer	$\beta = 0.1$	0.230 (0.028)	0.674 (0.049)	0.623 (0.061)
	$\beta = 0.01$	0.282 (0.024)	0.599 (0.006)	0.716 (0.032)
Sentinel	$\beta = 0.1$	0.176 (0.048)	0.803 (0.094)	0.462 (0.118)
	$\beta = 0.01$	0.266 (0.042)	0.593 (0.116)	0.724 (0.144)

Table 4.6: Evaluation metrics for each model and configuration.

As for the models trained with the lowest β , it is observed that, in well-defined datasets such as Diplomat, training achieves significant improvements: an increase of almost a third in the improved Euclidean distance and an increase in the error improvement of almost twice as much.

However, in synthetic datasets, with subjective metrics such as personality profiles (see Sub-

4.2. Evaluation of Phase II: Fine Tuning

section 3.3.2), there is likely to be a higher level of noise in the data compared to an expert-curated set. This factor means that a lower beta, while it may enhance substantial improvements in some cases (as in Diplomat), also increases the risk of misalignment, as observed in the Explorer and Sentinel models, where even a worsening occurs.

Model	$O(\mu)$	$C(\mu)$	$E(\mu)$	$A(\mu)$	$N(\mu)$
Baseline	0.631	0.636	0.423	0.769	0.333
Analyst	$\beta = 0.1$	0.690 (0.059)	0.615 (0.021)	0.387 (0.036)	0.766 (0.003)
	$\beta = 0.01$	0.819 (0.188)	0.604 (0.032)	0.460 (0.037)	0.830 (0.062)
	<i>Target</i>	0.90	0.60	0.35	0.25
Diplomat	$\beta = 0.1$	0.718 (0.086)	0.584 (0.052)	0.464 (0.041)	0.802 (0.033)
	$\beta = 0.01$	0.806 (0.175)	0.584 (0.052)	0.474 (0.051)	0.815 (0.046)
	<i>Target</i>	0.90	0.50	0.45	0.90
Explorer	$\beta = 0.1$	0.628 (0.003)	0.674 (0.039)	0.553 (0.130)	0.792 (0.023)
	$\beta = 0.01$	0.803 (0.172)	0.619 (0.017)	0.493 (0.070)	0.817 (0.048)
	<i>Target</i>	0.40	0.25	0.80	0.65
Sentinel	$\beta = 0.1$	0.565 (0.067)	0.769 (0.133)	0.384 (0.003)	0.759 (0.010)
	$\beta = 0.01$	0.812 (0.181)	0.610 (0.026)	0.496 (0.074)	0.821 (0.052)
	<i>Target</i>	0.25	0.90	0.50	0.65

Table 4.7: Big Five average values (Openness, Conscientiousness, Extraversion, Agreeableness, and Neuroticism) for each model and configuration.

Table 4.7 shows various characteristics of the behavior of the model in relation to each of the personality traits. Under the setting with $\beta = 0.1$, most of the variables tend to improve, as indicated in green, while with $\beta = 0.01$ the training becomes more sensitive and variable.

One of the most easily increased traits is Openness. The model increases this value easily, as seen in the case of Analyst (with an increase of 0.188) and Diplomat (0.175). However, reducing this trait proves much more difficult. With small β it barely drops, and with high β it can lead to errors. This suggests that the **LLaMA model finds it much easier to generate and learn open and responsive responses than rigid or closed responses**, reflecting its natural bias toward creative and flexible discourse.

In the case of Conscientiousness, the model **tolerates moderate reductions** without too much difficulty, but reaching very low values, such as 0.25, is more challenging. Even so, this trait does not present as many difficulties as Openness. In the opposite direction the results are acceptable, the Sentinel model shows a remarkable increase of up to 0.133 when trying to raise this feature.

For Extraversion, surprisingly, the model **fits both introverted and extroverted profiles** well. In the latter case, a significant improvement of up to 0.13 is achieved, indicating a good fit at both ends of the spectrum.

Agreeableness is by far the most difficult trait to modify. Increasing it, as with Diplomat, is relatively easy, as it fits with the prosocial and cooperative nature of the model. **However, reducing it (i.e., generating colder or more unpleasant profiles) goes against the inherent tendencies of the LLaMA model.** This makes sense, given that the model is optimized to generate friendly, empathetic and acceptable content, so it is unlikely to spontaneously generate a markedly unfriendly dataset.

Finally, Neuroticism is presented as the oddest trait of the five. The base model tends to show low levels of this trait, **which makes it easier to simulate emotionally stable individuals.**

Experiments and Results

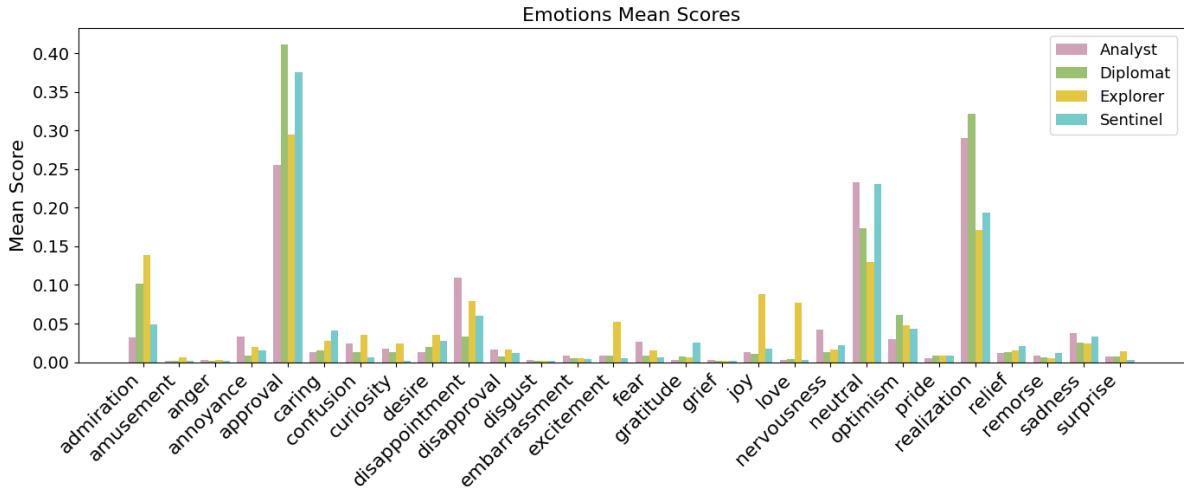


Figure 4.7: Comparison of emotions in fine-tuned models

However, elevating this trait to reflect more anxious or emotionally unstable profiles is considerably more difficult.

In summary, Table 4.7 provides evidence that certain traits are more manipulable than others. Reducing Agreeableness, decreasing Openness or increasing Neuroticism poses significant challenges. **This difficulty is further accentuated when the training dataset has been generated by an LLM** that also exhibits similar biases in those same traits.

In Table 4.8 and Figure 4.7 we can observe the most repeated words and the average sentiments of the models that most closely match the 4 target personalities.

In the case of the Analyst model, the results reveal an introspective and analytical narrative, with terms such as experience, thought-provoking, feedback, personal or decision. These words reflect a focus on **personal learning and self-reflection**. Emotionally, realization, approval and pride predominate, along with a generally neutral tone. Although negative emotions such as disappointment and annoyance appear, they are expressed in a restrained manner, suggesting balanced emotional management.

The Diplomat model uses **deeply empathetic and relationship-oriented language**, with words such as believe, perspective, empathy and deeply. This style corresponds to emotions such as approval, admiration, caring, realization and optimism, which define a prosocial, sensitive and emotionally harmonious profile. It is a model that conveys understanding and emotional openness, with a very low presence of negative emotions.

On the other hand, the Explorer model shows a more emotional and spontaneous narrative, **focused on personal experiences, relationships and daily decision making**. Words such as others, things, remember or try are frequent. Emotionally, it is the most polarized, with high levels of joy, excitement and disappointment, together with emotions in which it stands out such as curiosity, admiration and love. This profile stands out for its expressiveness and emotional reactivity, which provides richness, but also a certain instability.

The Sentinel model is characterized by a **more pragmatic and organized discourse**. Terms such as prioritize, situation, step back or goals stand out, reflecting a structured and functional

4.2. Evaluation of Phase II: Fine Tuning

Rank	Analyst	Diplomat	Explorer	Sentinel
1	when	can	when	when
2	can	believe	about	some
3	one	sense	remember	had
4	what	growth	all	time
5	personal	understanding	what	can
6	question	approach	just	have
7	had	embracing	some	make
8	thoughtprovoking	where	one	while
9	while	had	think	back
10	growth	found	values	where
11	criticism	can	time	someone
12	sense	us	first	take
13	like	deeper	you	one
14	instance	empathy	really	significant
15	feedback	perspectives	things	specific
16	particularly	often	out	goals
17	first	personal	make	prioritize
18	mind	profound	up	situation
19	experience	not	mean	yes
20	has	criticism	others	values

Table 4.8: Top 20 most frequent words (excluding common pronouns and auxiliaries) in the answer field across four personality types

approach. The most prominent emotions are approval, neutral, realization, gratitude, caring and pride, suggesting a sober, responsible and control-oriented emotional style. It is the most emotionally contained model, with a very low presence of negative emotions.

Overall, each model presents a clear coherence between the lexicon used and its dominant emotional profile. Diplomat and Explorer are more open and expressive, while Analyst and Sentinel show a more restrained and rational emotionality. These differences show how fine-tuning allows us to shape not only the content, but also the emotional tone of the responses generated.

Chapter 5

Final Conclusions and Future Work

5.1 Conclusions

This thesis has demonstrated the feasibility and value of using LLMs to generate synthetic human agents that are statistically grounded and psychologically expressive. Through profile generation and fine-tuning, we have shown that these models, when guided with structured design and curated data, can replicate complex demographic patterns and simulate coherent personality traits.

The first phase confirmed that models such as LLaMA, Qwen, Mistral, and Dolphin are capable of generating unique and diverse profiles, with LLaMA standing out as the most semantically rich and narratively complete. Qwen is more concise in its output, but it preserves categorical distributions with high fidelity. Mistral and Dolphin, by contrast, produced shorter, more factual profiles, yet maintaining lexical diversity. Some deviations from target values were found to be semantically meaningful corrections, rather than outright errors, which underscores the importance of using both stochastic generation based on real-world distributions and the benefit of completing and correcting possible failures with an LLM.

One interesting finding of this thesis is the strong influence of cultural and structural biases in the models. Qwen, trained primarily on Chinese datasets, generated profiles centered on social harmony and community, consistently avoiding themes like international travel or self-promotion. LLaMA and Dolphin, both aligned with Western data and values, emphasized individualism, autonomy, and self-realization. Dolphin, being uncensored, went further by producing more direct, raw, and sometimes uncomfortable traits especially in the weaknesses variable. These contrasts underscore the importance of considering not just what a model can generate, but also the cultural lens through which it interprets and constructs human identity.

The fine-tuning phase revealed that preference optimization methods like DPO can enhance LLMs toward distinct personality-specific vectors (even when we have tried with 4, the possibilities are endless). The β parameter, in particular, proved central to balancing the model's fidelity to the base architecture with its ability to internalize new preferences. Models trained with clean datasets, like Diplomat, benefited significantly from this flexibility. In contrast, noisier or more subjective datasets such as Explorer or Sentinel exhibited less predictable behavior and, in some cases, even performance deterioration.

An important insight from this work is that not all psychological traits are equally modifiable. Traits like Openness and Extraversion were relatively easy to increase, while reducing

Agreeableness or increasing Neuroticism posed significant challenges and revealed the underlying moral and emotional biases embedded in the language models themselves. Models such as LLaMA are inherently inclined to generate emotionally positive, socially acceptable, and prosocial content, making it difficult to simulate individuals with negative, conflicted, or emotionally unstable profiles. This was especially clear in the near absence of negative emotions like sadness, anger, fear, or guilt, even in personality segments where such traits would be appropriate. The result is a tendency to produce idealized agents that lack the imperfections, contradictions, and tensions that characterize real human behavior.

To sum up, this project shows that LLMs are powerful tools for simulating synthetic populations with psychological depth. Model selection, dataset design, bias awareness, and ethical reflection are all crucial elements in ensuring that these agents are not only coherent and expressive, but also representative of human complexity. The ability to control, enrich and evaluate such models opens promising paths in human-centered AI, simulation, behavioral science, and synthetic data generation, provided we remain critically aware of their limitations and actively work to mitigate their biases.

5.2 Future Work

The findings suggest multiple future research directions to enhance the robustness and applicability of the proposed synthetic profile generation based on language models. One of the most interesting is to extend the system to multilingual environments, which would make it possible to study how language influences or shapes personality expression, especially in diverse cultural contexts. Moreover, it would be very useful to design a hybrid approach that combines different models in real time according to the desired profile, thus adapting the generations to variables such as nationality or communication style.

Another possible line is the creation of a pipeline capable of generating profiles with negative or emotionally dysfunctional traits, which would allow a more realistic representation of human complexity, currently softened by the idealizing nature of the models.

However, although the methodology employed has shown promising results, it has been observed that the datasets generated by the LLMs themselves present limitations and biases when it comes to inducing any objective personality vector. In this sense, it would be highly recommended to apply RLHF techniques, in collaboration with expert psychologists, to build more reliable datasets, less noisy and better aligned with consolidated personality theories.

Finally, it would be interesting to develop a public API or interactive platform to generate customized profiles in real time, facilitating its practical application in areas such as social simulation, education, conversational assistants or the design of autonomous agents.

Bibliography

- [1] J. López Baeza, J. Sievert, A. Landwehr, J. Luft, P. Preuner, J. Bruns-Berentelg, A. Noyman, and J. Noennig, "Cityscope platform for real-time analysis and decision-support in urban design competitions," *International Journal of E-Planning Research*, vol. 10, 01 2021.
- [2] N. Coretti Sanchez, I. Martinez, L. Alonso Pastor, and K. Larson, "On the performance of shared autonomous bicycles: A simulation study," *Communications in Transportation Research*, vol. 2, p. 100066, 2022. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2772424722000166>
- [3] E. B. Pavón, L. A. Pastor, A. Padilla, M. Gamboa, and K. Larson, "Predicting mobility choice and community connectivity in latin america," *Case Studies on Transport Policy*, vol. 19, p. 101387, 2025. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2213624X25000240>
- [4] D. Antonelli, Azzato *et al.*, "Predicting behavioral changes as a result of land-use modifications in auto-centric communities: Data driven discrete mode choice modeling for dallas tx," in *2nd International Conference on Future Challenges in Sustainable Urban Planning & Territorial Management (SUPTM 2024)*. Cartagena, Spain: Universidad Politécnica de Cartagena, 2024.
- [5] N. Coretti Sanchez and K. Larson, "Multifunctional lightweight autonomous vehicles: an agent-based study," *Transportation*, Nov 2024. [Online]. Available: <https://doi.org/10.1007/s11116-024-10548-8>
- [6] A. Genua Cerviño, N. Coretti Sanchez, E. L. Wang, A. Grignard, and K. Larson, "Shared lightweight autonomous vehicles for urban food deliveries: A simulation study," *Future Transportation*, vol. 4, no. 2, pp. 634–658, 2024. [Online]. Available: <https://www.mdpi.com/2673-7590/4/2/30>
- [7] N. Coretti-Sanchez, J. Múgica, Q. Bowers, A. Grignard, E. C. Ferrer, L. Alonso, and K. Larson, "Urban mobility swarms: Towards a decentralized autonomous bicycle-sharing system," in *2023 IEEE 26th International Conference on Intelligent Transportation Systems (ITSC)*, 2023, pp. 2323–2330.
- [8] N. Coretti Sanchez, I. Martinez, L. Alonso Pastor, and K. Larson, "On the simulation of shared autonomous micro-mobility," *Communications in Transportation Research*, vol. 2, p. 100065, 2022. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2772424722000154>
- [9] L. Izquierdo, G. Palomo, A. Grignard, L. Alonso, M. Siller, and K. Larson, "An agent-based model to evaluate the perception of safety in informal settlements," in *Proceedings*

- of the 2020 Conference of The Computational Social Science Society of the Americas*, Z. Yang and E. von Briesen, Eds. Cham: Springer International Publishing, 2021, pp. 113–130.
- [10] C. Adornetto, A. Mora, K. Hu, L. I. Garcia, P. Atchade-Adelomou, G. Greco, L. A. A. Pastor, and K. Larson, “Generative Agents in Agent-Based Modeling: Overview, Validation, and Emerging Challenges,” *IEEE Transactions on Artificial Intelligence*, vol. 1, no. 01, pp. 1–20, May 5555. [Online]. Available: <https://doi.ieeecomputersociety.org/10.1109/TAI.2025.3566362>
 - [11] P. Atchadé, A. Mora, L. Alonso-Pastor, A. Grignard, A. Noyman, L. Izquierdo, C. Adornetto, K. Hu, F. Fernandez, H. Rahnama, M. Church, M. Elkatsha, and K. Larson, “Humanized agent-based models: a framework,” 08 2024.
 - [12] F. Almeida and G. Xexéo, “Word Embeddings: A Survey,” *arXiv e-prints*, p. arXiv:1901.09069, Jan. 2019.
 - [13] T. Mikolov, K. Chen, G. S. Corrado, and J. Dean, “Efficient estimation of word representations in vector space,” in *International Conference on Learning Representations*, 2013. [Online]. Available: <https://api.semanticscholar.org/CorpusID:5959482>
 - [14] J. Pennington, R. Socher, and C. Manning, “GloVe: Global vectors for word representation,” in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, A. Moschitti, B. Pang, and W. Daelemans, Eds. Doha, Qatar: Association for Computational Linguistics, Oct. 2014, pp. 1532–1543. [Online]. Available: <https://aclanthology.org/D14-1162/>
 - [15] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, “Enriching word vectors with subword information,” *Transactions of the Association for Computational Linguistics*, vol. 5, pp. 135–146, 2017. [Online]. Available: <https://aclanthology.org/Q17-1010/>
 - [16] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, “Deep contextualized word representations,” in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, M. Walker, H. Ji, and A. Stent, Eds. New Orleans, Louisiana: Association for Computational Linguistics, Jun. 2018, pp. 2227–2237. [Online]. Available: <https://aclanthology.org/N18-1202/>
 - [17] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding,” in *North American Chapter of the Association for Computational Linguistics*, 2019. [Online]. Available: <https://api.semanticscholar.org/CorpusID:52967399>
 - [18] Z. Liu, W. Lin, Y. Shi, and J. Zhao, “A robustly optimized bert pre-training approach with post-training,” in *Chinese Computational Linguistics: 20th China National Conference, CCL 2021, Hohhot, China, August 13–15, 2021, Proceedings*. Berlin, Heidelberg: Springer-Verlag, 2021, p. 471–484. [Online]. Available: https://doi.org/10.1007/978-3-030-84186-7_31
 - [19] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, “Attention is all you need,” in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30. Curran Associates, Inc., 2017. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fb053c1c4a845aa-Paper.pdf

BIBLIOGRAPHY

- [20] N. Reimers and I. Gurevych, "Sentence-bert: Sentence embeddings using siamese bert-networks," in *Conference on Empirical Methods in Natural Language Processing*, 2019. [Online]. Available: <https://api.semanticscholar.org/CorpusID:201646309>
- [21] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," in *International Conference on Machine Learning*, 2021. [Online]. Available: <https://api.semanticscholar.org/CorpusID:231591445>
- [22] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, "Language models are few-shot learners," in *Proceedings of the 34th International Conference on Neural Information Processing Systems*, ser. NIPS '20. Red Hook, NY, USA: Curran Associates Inc., 2020.
- [23] H. Touvron, T. Lavigil, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, A. Rodriguez, A. Joulin, E. Grave, and G. Lample, "Llama: Open and efficient foundation language models," *ArXiv*, vol. abs/2302.13971, 2023. [Online]. Available: <https://api.semanticscholar.org/CorpusID:257219404>
- [24] Z. Wang, Z. Chu, T. V. Doan, S. Ni, M. Yang, and W. Zhang, "History, development, and principles of large language models: an introductory survey," *AI and Ethics*, vol. 5, no. 3, pp. 1955–1971, Jun. 2025.
- [25] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, Oct. 1986.
- [26] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 11 1997. [Online]. Available: <https://doi.org/10.1162/neco.1997.9.8.1735>
- [27] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, "Improving language understanding by generative pre-training," https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf, 2018, openAI Technical Report.
- [28] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf, 2019, openAI Blog Post.
- [29] OpenAI, J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, and e. a. Florencia Leoni Aleman, "Gpt-4 technical report," 2024. [Online]. Available: <https://arxiv.org/abs/2303.08774>
- [30] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, "Albert: A lite bert for self-supervised learning of language representations," *ArXiv*, vol. abs/1909.11942, 2019. [Online]. Available: <https://api.semanticscholar.org/CorpusID:202888986>
- [31] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter," 10 2019.
- [32] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer, "BART: Denoising sequence-to-sequence pre-training for natural

- language generation, translation, and comprehension,” in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, D. Jurafsky, J. Chai, N. Schluter, and J. Tetreault, Eds. Online: Association for Computational Linguistics, Jul. 2020, pp. 7871–7880. [Online]. Available: <https://aclanthology.org/2020.acl-main.703/>
- [33] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, “Exploring the limits of transfer learning with a unified text-to-text transformer,” *J. Mach. Learn. Res.*, vol. 21, no. 1, Jan. 2020.
 - [34] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. Salakhutdinov, and Q. V. Le, *XLNet: generalized autoregressive pretraining for language understanding*. Red Hook, NY, USA: Curran Associates Inc., 2019.
 - [35] K. Clark, M.-T. Luong, Q. Le, and C. Manning, “Electra: Pre-training text encoders as discriminators rather than generators,” 03 2020.
 - [36] M. Shoeybi, M. M. A. Patwary, R. Puri, P. Legresley, J. Casper, and B. Catanzaro, “Megatron-lm: Training multi-billion parameter language models using gpu model parallelism,” 09 2019.
 - [37] D. Lepikhin, H. Lee, Y. Xu, D. Chen, O. Firat, Y. Huang, M. Krikun, N. Shazeer, and Z. Chen, “Gshard: Scaling giant models with conditional computation and automatic sharding,” 06 2020.
 - [38] J. Kaplan, S. McCandlish, T. Henighan, T. Brown, B. Chess, R. Child, S. Gray, A. Radford, J. Wu, and D. Amodei, “Scaling laws for neural language models,” 01 2020.
 - [39] D. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *International Conference on Learning Representations*, 12 2014.
 - [40] D. Ziegler, N. Stiennon, J. Wu, T. Brown, A. Radford, D. Amodei, P. Christiano, and G. Irving, “Fine-tuning language models from human preferences,” 09 2019.
 - [41] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. L. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, J. Schulman, J. Hilton, F. Kelton, L. Miller, M. Simens, A. Askell, P. Welinder, P. Christiano, J. Leike, and R. Lowe, “Training language models to follow instructions with human feedback,” in *Proceedings of the 36th International Conference on Neural Information Processing Systems*, ser. NIPS ’22. Red Hook, NY, USA: Curran Associates Inc., 2022.
 - [42] T. Schick, J. Dwivedi-Yu, R. Dessí, R. Raileanu, M. Lomeli, E. Hambro, L. Zettlemoyer, N. Cancedda, and T. Scialom, “Toolformer: language models can teach themselves to use tools,” in *Proceedings of the 37th International Conference on Neural Information Processing Systems*, ser. NIPS ’23. Red Hook, NY, USA: Curran Associates Inc., 2023.
 - [43] S. Yao, J. Zhao, D. Yu, I. Shafran, K. R. Narasimhan, and Y. Cao, “React: Synergizing reasoning and acting in language models,” in *NeurIPS 2022 Foundation Models for Decision Making Workshop*, 2022. [Online]. Available: <https://openreview.net/forum?id=tvl4u1ylcqs>
 - [44] H. Yang, S. Yue, and Y. He, “Auto-gpt for online decision making: Benchmarks and additional opinions,” 06 2023.
 - [45] T. Kojima, S. S. Gu, M. Reid, Y. Matsuo, and Y. Iwasawa, “Large language models are zero-shot reasoners,” in *Advances in Neural Information Processing Systems*, vol. 35, 2022, pp. 22 199–22 213.

BIBLIOGRAPHY

- [46] J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E. H. Chi, Q. V. Le, and D. Zhou, "Chain-of-thought prompting elicits reasoning in large language models," in *Proceedings of the 36th International Conference on Neural Information Processing Systems*, ser. NIPS '22. Red Hook, NY, USA: Curran Associates Inc., 2022.
- [47] N. F. Liu, K. Lin, J. Hewitt, A. Paranjape, M. Bevilacqua, F. Petroni, and P. Liang, "Lost in the middle: How language models use long contexts," *Transactions of the Association for Computational Linguistics*, vol. 12, pp. 157–173, 2024. [Online]. Available: <https://aclanthology.org/2024.tacl-1.9/>
- [48] Y. Gao, Y. Xiong, X. Gao, K. Jia, J. Pan, Y. Bi, Y. Dai, J. Sun, Q. Guo, M. Wang, and H. Wang, "Retrieval-augmented generation for large language models: A survey," *ArXiv*, vol. abs/2312.10997, 2023. [Online]. Available: <https://api.semanticscholar.org/CorpusID:266359151>
- [49] S. G. Patil, T. Zhang, X. Wang, and J. E. Gonzalez, "Gorilla: Large language model connected with massive apis," in *Advances in Neural Information Processing Systems*, A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, Eds., vol. 37. Curran Associates, Inc., 2024, pp. 126 544–126 565. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2024/file/e4c61f578ff07830f5c37378dd3ecb0d-Paper-Conference.pdf
- [50] Y. Liu, L. Huang, S. Li, S. Chen, H. Zhou, F. Meng, J. Zhou, and X. Sun, "Recall: A benchmark for llms robustness against external counterfactual knowledge," 2023.
- [51] X. V. Lin, X. Chen, M. Chen, W. Shi, M. Lomeli, R. James, P. Rodriguez, J. Kahn, G. Szilvassy, M. Lewis, L. S. Zettlemoyer, and S. Yih, "Ra-dit: Retrieval-augmented dual instruction tuning," *ArXiv*, vol. abs/2310.01352, 2023. [Online]. Available: <https://api.semanticscholar.org/CorpusID:263605962>
- [52] E. Almazrouei, H. Alobeidli, A. Alshamsi, A. Cappelli, R. Cojocaru, M. Debbah, Étienne Goffinet, D. Hesslow, J. Launay, Q. Malartic, D. Mazzotta, B. Noune, B. Pannier, and G. Penedo, "The falcon series of open language models," 2023. [Online]. Available: <https://arxiv.org/abs/2311.16867>
- [53] L. Xu, H. Xie, S.-Z. J. Qin, X. Tao, and F. L. Wang, "Parameter-efficient fine-tuning methods for pretrained language models: A critical review and assessment," 2023.
- [54] N. Houlsby, A. Giurgiu, S. Jastrzebski, B. Morrone, Q. Laroussilhe, A. Gesmundo, M. Attariyan, and S. Gelly, "Parameter-efficient transfer learning for nlp," 02 2019.
- [55] C. Poth, H. Sterz, I. Paul, S. Purkayastha, L. Engländer, T. Imhof, I. Vulić, S. Ruder, I. Gurevych, and J. Pfeiffer, "Adapters: A unified library for parameter-efficient and modular transfer learning," 2023. [Online]. Available: <https://arxiv.org/abs/2311.11077>
- [56] J. Pfeiffer, A. Rücklé, C. Poth, A. Kamath, I. Vulić, S. Ruder, K. Cho, and I. Gurevych, "AdapterHub: A framework for adapting transformers," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, Q. Liu and D. Schlangen, Eds. Online: Association for Computational Linguistics, Oct. 2020, pp. 46–54. [Online]. Available: <https://aclanthology.org/2020.emnlp-demos.7/>
- [57] T. Lei, J. Bai, S. Brahma, J. Ainslie, K. Lee, Y. Zhou, N. Du, V. Y. Zhao, Y. Wu, B. Li, Y. Zhang, and M.-W. Chang, "Conditional adapters: parameter-efficient transfer learning with fast inference," in *Proceedings of the 37th International Conference on Neural Information Processing Systems*, ser. NIPS '23. Red Hook, NY, USA: Curran Associates Inc., 2023.

- [58] H. Zhao, H. Tan, and H. Mei, "Tiny-attention adapter: Contexts are more important than the number of parameters," 10 2022.
- [59] J. Pfeiffer, A. Kamath, A. Rücklé, K. Cho, and I. Gurevych, "Adapterfusion: Non-destructive task composition for transfer learning," *ArXiv*, vol. abs/2005.00247, 2020. [Online]. Available: <https://api.semanticscholar.org/CorpusID:218470208>
- [60] S. He, R.-Z. Fan, L. Ding, L. Shen, T. Zhou, and D. Tao, "Mera: Merging pretrained adapters for few-shot learning," 08 2023.
- [61] R. Mahabadi, S. Ruder, M. dehghani, and J. Henderson, "Parameter-efficient multi-task fine-tuning for transformers via shared hypernetworks," 06 2021.
- [62] A. Chronopoulou, M. Peters, A. Fraser, and J. Dodge, "AdapterSoup: Weight averaging to improve generalization of pretrained language models," in *Findings of the Association for Computational Linguistics: EACL 2023*, A. Vlachos and I. Augenstein, Eds. Dubrovnik, Croatia: Association for Computational Linguistics, May 2023, pp. 2054–2063. [Online]. Available: <https://aclanthology.org/2023.findings-eacl.153/>
- [63] B. Lester, R. Al-Rfou, and N. Constant, "The power of scale for parameter-efficient prompt tuning," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, M.-F. Moens, X. Huang, L. Specia, and S. W.-t. Yih, Eds. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics, Nov. 2021, pp. 3045–3059. [Online]. Available: <https://aclanthology.org/2021.emnlp-main.243/>
- [64] K. Hambardzumyan, H. Khachatrian, and J. May, "WARP: Word-level Adversarial ReProgramming," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, C. Zong, F. Xia, W. Li, and R. Navigli, Eds. Online: Association for Computational Linguistics, Aug. 2021, pp. 4921–4933. [Online]. Available: <https://aclanthology.org/2021.acl-long.381/>
- [65] X. L. Li and P. Liang, "Prefix-tuning: Optimizing continuous prompts for generation," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, C. Zong, F. Xia, W. Li, and R. Navigli, Eds. Online: Association for Computational Linguistics, Aug. 2021, pp. 4582–4597. [Online]. Available: <https://aclanthology.org/2021.acl-long.353/>
- [66] X. Liu, Y. Zheng, Z. Du, M. Ding, Y. Qian, Z. Yang, and J. Tang, "Gpt understands, too," *AI Open*, vol. 5, pp. 208–215, 2024. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2666651023000141>
- [67] A. Asai, M. Salehi, M. E. Peters, and H. Hajishirzi, "Attempt: Parameter-efficient multi-task tuning via attentional mixtures of soft prompts," 2022. [Online]. Available: <https://arxiv.org/abs/2205.11961>
- [68] Z. Wang, R. Panda, L. Karlinsky, R. S. Feris, H. Sun, and Y. Kim, "Multitask prompt tuning enables parameter-efficient transfer learning," *ArXiv*, vol. abs/2303.02861, 2023. [Online]. Available: <https://api.semanticscholar.org/CorpusID:257365136>
- [69] H. Liu, D. Tam, M. Muqeeth, J. Mohta, T. Huang, M. Bansal, and C. Raffel, "Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning," in *Proceedings of the 36th International Conference on Neural Information Processing Systems*, ser. NIPS '22. Red Hook, NY, USA: Curran Associates Inc., 2022.

BIBLIOGRAPHY

- [70] X. Yang, J. Y. Huang, W. Zhou, and M. Chen, "Parameter-efficient tuning with special token adaptation," 2023. [Online]. Available: <https://arxiv.org/abs/2210.04382>
- [71] Y. Chen, Q. Fu, G. Fan, L. Du, J.-G. Lou, S. Han, D. Zhang, Z. Li, and Y. Xiao, "Hadamard adapter: An extreme parameter-efficient adapter tuning method for pre-trained language models," 07 2024.
- [72] E. Ben Zaken, Y. Goldberg, and S. Ravfogel, "BitFit: Simple parameter-efficient fine-tuning for transformer-based masked language-models," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, S. Muresan, P. Nakov, and A. Villavicencio, Eds. Dublin, Ireland: Association for Computational Linguistics, May 2022, pp. 1–9. [Online]. Available: <https://aclanthology.org/2022.acl-short.1/>
- [73] M. Zhao, T. Lin, F. Mi, M. Jaggi, and H. Schütze, "Masking as an efficient alternative to finetuning for pretrained language models," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, B. Webber, T. Cohn, Y. He, and Y. Liu, Eds. Online: Association for Computational Linguistics, Nov. 2020, pp. 2226–2241. [Online]. Available: <https://aclanthology.org/2020.emnlp-main.174/>
- [74] T. Jiang, D. Wang, and F. Zhuang, "Pruning pre-trained language models without fine-tuning," 10 2022.
- [75] Y.-L. Sung, V. Nair, and C. Raffel, "Training neural networks with fixed sparse masks," in *Proceedings of the 35th International Conference on Neural Information Processing Systems*, ser. NIPS '21. Red Hook, NY, USA: Curran Associates Inc., 2021.
- [76] L. Wang, S. Chen, L. Jiang, S. Pan, R. Cai, S. Yang, and F. Yang, "Parameter-efficient fine-tuning in large language models: a survey of methodologies," *Artificial Intelligence Review*, vol. 58, no. 8, p. 227, May 2025.
- [77] R. Xu, F. Luo, Z. Zhang, C. Tan, B. Chang, S. Huang, and F. Huang, "Raise a child in large language model: Towards effective and generalizable fine-tuning," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, M.-F. Moens, X. Huang, L. Specia, and S. W.-t. Yih, Eds. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics, Nov. 2021, pp. 9514–9528. [Online]. Available: <https://aclanthology.org/2021.emnlp-main.749/>
- [78] D. Guo, A. Rush, and Y. Kim, "Parameter-efficient transfer learning with diff pruning," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, C. Zong, F. Xia, W. Li, and R. Navigli, Eds. Online: Association for Computational Linguistics, Aug. 2021, pp. 4884–4896. [Online]. Available: <https://aclanthology.org/2021.acl-long.378/>
- [79] P. Foret, A. Kleiner, H. Mobahi, and B. Neyshabur, "Sharpness-aware minimization for efficiently improving generalization," 10 2020.
- [80] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "Lora: Low-rank adaptation of large language models," 2021. [Online]. Available: <https://arxiv.org/abs/2106.09685>
- [81] M. Valipour, M. Rezagholizadeh, I. Kobyzhev, and A. Ghodsi, "DyLoRA: Parameter-efficient tuning of pre-trained models using dynamic search-free low-rank adaptation," in *Proceedings of the 17th Conference of the European Chapter of the Association for*

- Computational Linguistics*, A. Vlachos and I. Augenstein, Eds. Dubrovnik, Croatia: Association for Computational Linguistics, May 2023, pp. 3274–3287. [Online]. Available: <https://aclanthology.org/2023.eacl-main.239/>
- [82] Q. Zhang, M. Chen, A. Bukharin, P. He, Y. Cheng, W. Chen, and T. Zhao, “Adaptive budget allocation for parameter-efficient fine-tuning,” 03 2023.
- [83] T. Dettmers, A. Pagnoni, A. Holtzman, and L. Zettlemoyer, “Qlora: efficient finetuning of quantized llms,” in *Proceedings of the 37th International Conference on Neural Information Processing Systems*, ser. NIPS ’23. Red Hook, NY, USA: Curran Associates Inc., 2023.
- [84] Y. Xin, S. Luo, H. Zhou, J. Du, X. Liu, Y. Fan, Q. Li, and Y. Du, “Parameter-efficient fine-tuning for pre-trained vision models: A survey,” 2024.
- [85] R. Mahabadi, J. Henderson, and S. Ruder, “Compacter: Efficient low-rank hypercomplex adapter layers,” 06 2021.
- [86] Y. Mao, L. Mathias, R. Hou, A. Almahairi, H. Ma, J. Han, S. Yih, and M. Khabsa, “UniPELT: A unified framework for parameter-efficient language model tuning,” in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, S. Muresan, P. Nakov, and A. Villavicencio, Eds. Dublin, Ireland: Association for Computational Linguistics, May 2022, pp. 6253–6264. [Online]. Available: <https://aclanthology.org/2022.acl-long.433/>
- [87] H. Zhou, X. Wan, I. Vulić, and A. Korhonen, “AutoPEFT: Automatic configuration search for parameter-efficient fine-tuning,” *Transactions of the Association for Computational Linguistics*, vol. 12, pp. 525–542, 2024. [Online]. Available: <https://aclanthology.org/2024.tacl-1.29/>
- [88] D. Zhang, K. Zhang, S. Chu, L. Wu, X. Li, and S. Wei, “More: A mixture of low-rank experts for adaptive multi-task learning,” 05 2025.
- [89] S. He, L. Ding, D. Dong, J. Zhang, and D. Tao, “SparseAdapter: An easy approach for improving the parameter-efficiency of adapters,” in *Findings of the Association for Computational Linguistics: EMNLP 2022*, Y. Goldberg, Z. Kozareva, and Y. Zhang, Eds. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics, Dec. 2022, pp. 2184–2190. [Online]. Available: <https://aclanthology.org/2022.findings-emnlp.160/>
- [90] N. Zhou, H. Wang, Y. Zheng, and D. Huang, “Progressive parameter efficient transfer learning for semantic segmentation,” in *The Thirteenth International Conference on Learning Representations*, 2025. [Online]. Available: <https://openreview.net/forum?id=YNbLUGDAX5>
- [91] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman, “Glue: A multi-task benchmark and analysis platform for natural language understanding,” 2019. [Online]. Available: <https://arxiv.org/abs/1804.07461>
- [92] O. Bojar, Y. Graham, A. Kamran, and M. Stanojević, “Results of the WMT16 metrics shared task,” in *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, O. Bojar, C. Buck, R. Chatterjee, C. Federmann, L. Guillou, B. Haddow, M. Huck, A. J. Yepes, A. Névéol, M. Neves, P. Pecina, M. Popel, P. Koehn, C. Monz, M. Negri, M. Post, L. Specia, K. Verspoor, J. Tiedemann, and M. Turchi, Eds. Berlin, Germany: Association for Computational Linguistics, Aug. 2016, pp. 199–231. [Online]. Available: <https://aclanthology.org/W16-2302/>

BIBLIOGRAPHY

- [93] D. Hendrycks, C. Burns, S. Basart, A. Zou, M. Mazeika, D. Song, and J. Steinhardt, "Measuring massive multitask language understanding," 2021. [Online]. Available: <https://arxiv.org/abs/2009.03300>
- [94] N. Stiennon, L. Ouyang, J. Wu, D. M. Ziegler, R. Lowe, C. Voss, A. Radford, D. Amodei, and P. Christiano, "Learning to summarize from human feedback," in *Proceedings of the 34th International Conference on Neural Information Processing Systems*, ser. NIPS '20. Red Hook, NY, USA: Curran Associates Inc., 2020.
- [95] P. F. Christiano, J. Leike, T. B. Brown, M. Martic, S. Legg, and D. Amodei, "Deep reinforcement learning from human preferences," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, ser. NIPS'17. Red Hook, NY, USA: Curran Associates Inc., 2017, p. 4302–4310.
- [96] B. Gao, F. Song, Y. Miao, Z. Cai, Z. Yang, L. Chen, H. Hu, R. Xu, Q. Dong, C. Zheng, S. Quan, W. Xiao, G. Zhang, D. Zan, K. Lu, B. Yu, D. Liu, Z. Cui, J. Yang, L. Sha, H. Wang, Z. Sui, P. Wang, T. Liu, and B. Chang, "Towards a unified view of preference learning for large language models: A survey," 2024. [Online]. Available: <https://arxiv.org/abs/2409.02795>
- [97] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," 2017. [Online]. Available: <https://arxiv.org/abs/1707.06347>
- [98] R. Rafailov, A. Sharma, E. Mitchell, S. Ermon, C. D. Manning, and C. Finn, "Direct preference optimization: your language model is secretly a reward model," in *Proceedings of the 37th International Conference on Neural Information Processing Systems*, ser. NIPS '23. Red Hook, NY, USA: Curran Associates Inc., 2023.
- [99] S. Garg, A. Singh, S. Singh, and P. Chopra, "Ipo: Your language model is secretly a preference classifier," 02 2025.
- [100] Y. Meng, M. Xia, and D. Chen, "SimPO: Simple preference optimization with a reference-free reward," in *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. [Online]. Available: <https://openreview.net/forum?id=3Tzcot1LKb>
- [101] H. Yuan, Z. Yuan, C. Tan, W. Wang, S. Huang, and F. Huang, "RRHF: Rank responses to align language models with human feedback," in *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. [Online]. Available: <https://openreview.net/forum?id=EdIGMCHk4l>
- [102] F. Song, B. Yu, M. Li, H. Yu, F. Huang, Y. Li, and H. Wang, "Preference ranking optimization for human alignment," in *Proceedings of the Thirty-Eighth AAAI Conference on Artificial Intelligence and Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence and Fourteenth Symposium on Educational Advances in Artificial Intelligence*, ser. AAAI'24/IAAI'24/EAAI'24. AAAI Press, 2024. [Online]. Available: <https://doi.org/10.1609/aaai.v38i17.29865>
- [103] Y. Tang, Z. D. Guo, Z. Zheng, D. Calandriello, R. Munos, M. Rowland, P. H. Richemond, M. Valko, B. Avila Pires, and B. Piot, "Generalized preference optimization: A unified approach to offline alignment," in *Proceedings of the 41st International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, R. Salakhutdinov, Z. Kolter, K. Heller, A. Weller, N. Oliver, J. Scarlett, and F. Berkenkamp, Eds., vol. 235. PMLR, 21–27 Jul 2024, pp. 47725–47742. [Online]. Available: <https://proceedings.mlr.press/v235/tang24b.html>

- [104] J. Cheng, X. Liu, K. Zheng, P. Ke, H. Wang, Y. Dong, J. Tang, and M. Huang, "Black-box prompt optimization: Aligning large language models without model training," in *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, L.-W. Ku, A. Martins, and V. Srikumar, Eds. Bangkok, Thailand: Association for Computational Linguistics, Aug. 2024, pp. 3201–3219. [Online]. Available: <https://aclanthology.org/2024.acl-long.176/>
- [105] K. Yang and D. Klein, "FUDGE: Controlled text generation with future discriminators," in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, K. Toutanova, A. Rumshisky, L. Zettlemoyer, D. Hakkani-Tur, I. Beltagy, S. Bethard, R. Cotterell, T. Chakraborty, and Y. Zhou, Eds. Online: Association for Computational Linguistics, Jun. 2021, pp. 3511–3535. [Online]. Available: <https://aclanthology.org/2021.naacl-main.276/>
- [106] Y. Li, F. Wei, J. Zhao, C. Zhang, and H. Zhang, "RAIN: Your language models can align themselves without finetuning," in *The Twelfth International Conference on Learning Representations*, 2024. [Online]. Available: <https://openreview.net/forum?id=pETSFWMUzy>
- [107] L. Wang, C. Ma, X. Feng, Z. Zhang, H. Yang, J. Zhang, Z. Chen, J. Tang, X. Chen, Y. Lin, W. X. Zhao, Z. Wei, and J. Wen, "A survey on large language model based autonomous agents," *Frontiers of Computer Science*, vol. 18, no. 6, Mar. 2024. [Online]. Available: <http://dx.doi.org/10.1007/s11704-024-40231-1>
- [108] Z. Xi, W. Chen, X. Guo, W. He, Y. Ding, B. Hong, M. Zhang, J. Wang, S. Jin, E. Zhou, R. Zheng, X. Fan, X. Wang, L. Xiong, Y. Zhou, W. Wang, C. Jiang, Y. Zou, X. Liu, Z. Yin, S. Dou, R. Weng, W. Qin, Y. Zheng, X. Qiu, X. Huang, Q. Zhang, and T. Gui, "The rise and potential of large language model based agents: a survey," *Science China Information Sciences*, vol. 68, no. 2, p. 121101, Jan. 2025.
- [109] H. Liu, C. Sferrazza, and P. Abbeel, "Chain of hindsight aligns language models with feedback," 2023. [Online]. Available: <https://arxiv.org/abs/2302.02676>
- [110] Y. Dan, Z. Lei, Y. Gu, Y. Li, J. Yin, J. Lin, L. Ye, Z. Tie, Y. Zhou, Y. Wang, A. Zhou, Z. Zhou, Q. Chen, J. Zhou, L. He, and X. Qiu, "Educhat: A large-scale language model-based chatbot system for intelligent education," 08 2023.
- [111] Y. Qin, S. Liang, Y. Ye, K. Zhu, L. Yan, Y. Lu, Y. Lin, X. Cong, X. Tang, B. Qian, S. Zhao, R. Tian, R. Xie, J. Zhou, M. Gerstein, D. Li, Z. Liu, and M. Sun, "Toolllm: Facilitating large language models to master 16000+ real-world apis," 07 2023.
- [112] S. Yao, H. Chen, J. Yang, and K. Narasimhan, "Webshop: Towards scalable real-world web interaction with grounded language agents," in *Advances in Neural Information Processing Systems*, S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, Eds., vol. 35. Curran Associates, Inc., 2022, pp. 20744–20757. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2022/file/82ad13ec01f9fe44c01cb91814fd7b8c-Paper-Conference.pdf
- [113] K. Fischer, "Reflective linguistic programming (rlp): A stepping stone in socially-aware agi (socialagi)," 05 2023.
- [114] Z. Wang, S. Cai, G. Chen, A. Liu, X. Ma, and Y. Liang, "Describe, explain, plan and select: Interactive planning with large language models enables open-world multi-task agents," 2024. [Online]. Available: <https://arxiv.org/abs/2302.01560>

BIBLIOGRAPHY

- [115] Z. Mandi, S. Jain, and S. Song, "Roco: Dialectic multi-robot collaboration with large language models," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 2024, pp. 286–299.
- [116] G. Wang, Y. Xie, Y. Jiang, A. Mandlekar, C. Xiao, Y. Zhu, L. Fan, and A. Anandkumar, "Voyager: An open-ended embodied agent with large language models," 2023. [Online]. Available: <https://arxiv.org/abs/2305.16291>
- [117] C. Colas, L. Teodorescu, P.-Y. Oudeyer, X. Yuan, and M.-A. Côté, "Augmenting autotelic agents with large language models," 05 2023.
- [118] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," in *International Conference on Learning Representations*, 2021. [Online]. Available: <https://openreview.net/forum?id=YicbFdNTTy>
- [119] M. Douze, A. Guzhva, C. Deng, J. Johnson, G. Szilvassy, P.-E. Mazaré, M. Lomeli, L. Hosseini, and H. Jégou, "The faiss library," 2025. [Online]. Available: <https://arxiv.org/abs/2401.08281>
- [120] Y. A. Malkov and D. A. Yashunin, "Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 4, p. 824–836, Apr. 2020. [Online]. Available: <https://doi.org/10.1109/TPAMI.2018.2889473>
- [121] A. Modarressi, A. Imani, M. Fayyaz, and H. Schütze, "Ret-llm: Towards a general read-write memory for large language models," 05 2023.
- [122] J. S. Park, J. O'Brien, C. J. Cai, M. R. Morris, P. Liang, and M. S. Bernstein, "Generative agents: Interactive simulacra of human behavior," in *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, ser. UIST '23. New York, NY, USA: Association for Computing Machinery, 2023. [Online]. Available: <https://doi.org/10.1145/3586183.3606763>
- [123] A. Zhao, D. Huang, Q. Xu, M. Lin, Y.-J. Liu, and G. Huang, "Expel: Llm agents are experiential learners," in *Proceedings of the Thirty-Eighth AAAI Conference on Artificial Intelligence and Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence and Fourteenth Symposium on Educational Advances in Artificial Intelligence*, ser. AAAI'24/IAAI'24/EAAI'24. AAAI Press, 2024. [Online]. Available: <https://doi.org/10.1609/aaai.v38i17.29936>
- [124] B. Xu, Z. Peng, B. Lei, S. Mukherjee, Y. Liu, and D. Xu, "Rewoo: Decoupling reasoning from observations for efficient augmented language models," 2023. [Online]. Available: <https://arxiv.org/abs/2305.18323>
- [125] Y. Wang, Z. Jiang, Z. Chen, F. Yang, Y. Zhou, E. Cho, X. Fan, Y. Lu, X. Huang, and Y. Yang, "RecMind: Large language model powered agent for recommendation," in *Findings of the Association for Computational Linguistics: NAACL 2024*, K. Duh, H. Gomez, and S. Bethard, Eds. Mexico City, Mexico: Association for Computational Linguistics, Jun. 2024, pp. 4351–4364. [Online]. Available: <https://aclanthology.org/2024.findings-naacl.271/>
- [126] X. Wang, J. Wei, D. Schuurmans, Q. Le, E. Chi, and D. Zhou, "Self-consistency improves chain of thought reasoning in language models," 03 2022.

- [127] C. H. Song, J. Wu, C. Washington, B. M. Sadler, W.-L. Chao, and Y. Su, "Llm-planner: Few-shot grounded planning for embodied agents with large language models," 2023. [Online]. Available: <https://arxiv.org/abs/2212.04088>
- [128] B. Liu, Y. Jiang, X. Zhang, Q. Liu, S. Zhang, J. Biswas, and P. Stone, "Llm+p: Empowering large language models with optimal planning proficiency," 04 2023.
- [129] M. Li, Y. Zhao, B. Yu, F. Song, H. Li, H. Yu, Z. Li, F. Huang, and Y. Li, "API-bank: A comprehensive benchmark for tool-augmented LLMs," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, H. Bouamor, J. Pino, and K. Bali, Eds. Singapore: Association for Computational Linguistics, Dec. 2023, pp. 3102–3116. [Online]. Available: <https://aclanthology.org/2023.emnlp-main.187/>
- [130] Y. Liang, C. Wu, T. Song, W. Wu, Y. Xia, Y. Liu, Y. Ou, S. Lu, L. Ji, S. Mao, Y. Wang, L. Shou, M. Gong, and N. Duan, "Taskmatrix.ai: Completing tasks by connecting foundation models with millions of apis," *Intelligent Computing*, vol. 3, 02 2024.
- [131] C. Hu, J. Fu, C. Du, S. Luo, J. Zhao, and H. Zhao, "Chatdb: Augmenting llms with databases as their symbolic memory," 2023. [Online]. Available: <https://arxiv.org/abs/2306.03901>
- [132] E. Karpas, O. Abend, Y. Belinkov, B. Lenz, O. Lieber, N. Ratner, Y. Shoham, H. Bata, Y. Levine, K. Leyton-Brown, D. Muhlgay, N. Rozen, E. Schwartz, G. Shachaf, S. Shalev-Shwartz, A. Shashua, and M. Tenenholz, "Mrkl systems: A modular, neuro-symbolic architecture that combines large language models, external knowledge sources and discrete reasoning," 05 2022.
- [133] A. Bran, S. Cox, O. Schilter, C. Baldassari, A. White, and P. Schwaller, "Augmenting large language models with chemistry tools," *Nature Machine Intelligence*, vol. 6, pp. 1–11, 05 2024.
- [134] D. Surís, S. Menon, and C. Vondrick, "Vipergpt: Visual inference via python execution for reasoning," 2023. [Online]. Available: <https://arxiv.org/abs/2303.08128>
- [135] J.-T. Huang, M. Lam, E. Li, S. Ren, W. Wang, W. Jiao, Z. Tu, and M. Lyu, "Emotionally numb or empathetic? evaluating how llms feel using emotionbench," 08 2023.
- [136] L. P. Argyle, E. C. Busby, N. Fulda, J. R. Gubler, C. Rytting, and D. Wingate, "Out of one, many: Using language models to simulate human samples," *Political Analysis*, vol. 31, no. 3, p. 337–351, 2023.
- [137] X. Liu, H. Yu, H. Zhang, Y. Xu, X. Lei, H. Lai, Y. Gu, H. Ding, K. Men, K. Yang, S. Zhang, X. Deng, A. Zeng, Z. Du, C. Zhang, S. Shen, T. Zhang, Y. Su, H. Sun, M. Huang, Y. Dong, and J. Tang, "Agentbench: Evaluating LLMs as agents," in *The Twelfth International Conference on Learning Representations*, 2024. [Online]. Available: <https://openreview.net/forum?id=zAdUB0aCTQ>
- [138] S. Zhou, F. Xu, H. Zhu, X. Zhou, R. Lo, A. Sridhar, X. Cheng, Y. Bisk, D. Fried, U. Alon, and G. Neubig, "Webarena: A realistic web environment for building autonomous agents," 07 2023.
- [139] C. Qian, W. Liu, H. Liu, N. Chen, Y. Dang, J. Li, C. Yang, W. Chen, Y. Su, X. Cong, J. Xu, D. Li, Z. Liu, and M. Sun, "ChatDev: Communicative agents for software development," in *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, L.-W. Ku, A. Martins, and V. Srikumar,

BIBLIOGRAPHY

- Eds. Bangkok, Thailand: Association for Computational Linguistics, Aug. 2024, pp. 15174–15186. [Online]. Available: <https://aclanthology.org/2024.acl-long.810/>
- [140] Y. Dong, X. Jiang, Z. Jin, and G. Li, "Self-collaboration code generation via chatgpt," *ACM Trans. Softw. Eng. Methodol.*, vol. 33, no. 7, Sep. 2024. [Online]. Available: <https://doi.org/10.1145/3672459>
- [141] J. Lin, H. Zhao, A. Zhang, Y. Wu, H. Ping, and Q. Chen, "Agentsims: An open-source sandbox for large language model evaluation," 08 2023.
- [142] G. Kovač, R. Portelas, P. Dominey, and P.-Y. Oudeyer, "The socialai school: Insights from developmental psychology towards artificial socio-cultural agents," 07 2023.
- [143] A. C. da Rocha Costa, *A variational basis for the regulation and structuration mechanisms of agent societies*, 2019th ed. Cham, Switzerland: Springer Nature, May 2019.
- [144] S. Wimmer, A. Pfeiffer, and N. Denk, "The everyday life in the sims 4 during a pandemic. a life simulation as a virtual mirror of society?" in *INTED2021 Proceedings*, ser. 15th International Technology, Education and Development Conference. IATED, 8-9 March, 2021 2021, pp. 5754–5760. [Online]. Available: <https://doi.org/10.21125/inted.2021.1162>
- [145] L.-H. Lee, T. Braud, P. Y. Zhou, L. Wang, D. Xu, Z. Lin, A. Kumar, C. Bermejo, and P. Hui, "All one needs to know about metaverse: A complete survey on technological singularity, virtual ecosystem, and research agenda," 10 2021.
- [146] M.-A. Côté, A. Kadar, X. Yuan, Q. Kybartas, T. Barnes, E. Fine, J. Moore, M. Hausknecht, L. El Asri, M. Adada, W. Tay, and A. Trischler, "Textworld: A learning environment for text-based games," 06 2018.
- [147] G. Li, H. A. Al Kader Hammoud, H. Itani, D. Khizbulin, and B. Ghanem, "Camel: communicative agents for "mind" exploration of large language model society," in *Proceedings of the 37th International Conference on Neural Information Processing Systems*, ser. NIPS '23. Red Hook, NY, USA: Curran Associates Inc., 2023.
- [148] D. Driess, F. Xia, M. S. M. Sajjadi, C. Lynch, A. Chowdhery, B. Ichter, A. Wahid, J. Tompson, Q. Vuong, T. Yu, W. Huang, Y. Chebotar, P. Sermanet, D. Duckworth, S. Levine, V. Vanhoucke, K. Hausman, M. Toussaint, K. Greff, A. Zeng, I. Mordatch, and P. Florence, "Palm-e: an embodied multimodal language model," in *Proceedings of the 40th International Conference on Machine Learning*, ser. ICML'23. JMLR.org, 2023.
- [149] H. Bharadhwaj, J. Vakil, M. Sharma, A. Gupta, S. Tulsiani, and V. Kumar, "Roboagent: Generalization and efficiency in robot manipulation via semantic augmentations and action chunking," 09 2023.
- [150] Q. Wu, G. Bansal, J. Zhang, Y. Wu, S. Zhang, E. Zhu, B. Li, L. Jiang, X. Zhang, and C. Wang, "Autogen: Enabling next-gen llm applications via multi-agent conversation framework," 08 2023.
- [151] W. Chen, Y. Su, J. Zuo, C. Yang, C. Yuan, C. Qian, C.-M. Chan, Y. Qin, Y. Lu, R. Xie, Z. Liu, M. Sun, and J. Zhou, "Agentverse: Facilitating multi-agent collaboration and exploring emergent behaviors in agents," 08 2023.
- [152] Z. Zhang, N. Liu, S. Qi, C. Zhang, Z. Rong, Y. Yang, and S. Cui, "Heterogeneous value evaluation for large language models," 05 2023.
- [153] M. Binz and E. Schulz, "Using cognitive psychology to understand gpt-3," 06 2022.

- [154] X. Wang, X. Li, Z. Yin, Y. Wu, and L. Jia, "Emotional intelligence of large language models," 2023. [Online]. Available: <https://arxiv.org/abs/2307.09042>
- [155] Z. Elyoseph, D. Hadar Shoval, K. Asraf, and M. Lvovsky, "Chatgpt outperforms humans in emotional awareness evaluations," *Frontiers in Psychology*, vol. 14, 05 2023.
- [156] G. Caron and S. Srivastava, "Identifying and manipulating the personality traits of language models," 12 2022.
- [157] K. Pan and Y. Zeng, "Do llms possess a personality? making the mbti test an amazing evaluation for large language models," 07 2023.
- [158] H. Du, R. Zhang, D. Niyato, J. Kang, Z. Xiong, and D. I. Kim, "Reinforcement learning with large language models (llms) interaction for network services," in *2024 International Conference on Computing, Networking and Communications (ICNC)*, 2024, pp. 799–803.
- [159] W. Li, J. Liu, A. Liu, X. Zhou, M. T. Diab, and M. Sap, "BIG5-CHAT: Shaping LLM personalities through training on human-grounded data," 2024. [Online]. Available: <https://openreview.net/forum?id=TqwTzLjzGS>
- [160] C. Gao, X. Lan, Z. Lu, J. Mao, J. Piao, H. Wang, D. Jin, and Y. Li, "S3: Social-network simulation system with large language model-empowered agents," *SSRN Electronic Journal*, 01 2023.
- [161] Y. Xu, S. Wang, P. Li, F. Luo, X. Wang, W. Liu, and Y. Liu, "Exploring large language models for communication games: An empirical study on werewolf," 09 2023.
- [162] G. Marsan, N. Bellomo, and A. Tosin, *Complex Systems and Society - Modeling and Simulation*. Springer, 04 2013.
- [163] N. Helberger and N. Diakopoulos, "Chatgpt and the ai act," *Internet Policy Review*, vol. 12, 02 2023.
- [164] H. R. Kirk, Y. Jun, H. Iqbal, E. Benussi, F. Volpin, F. A. Dreyer, A. Shtedritski, and Y. M. Asano, "Bias out-of-the-box: an empirical analysis of intersectional occupational biases in popular generative language models," in *Proceedings of the 35th International Conference on Neural Information Processing Systems*, ser. NIPS '21. Red Hook, NY, USA: Curran Associates Inc., 2021.
- [165] P. Henderson, K. Sinha, N. Angelard-Gontier, N. R. Ke, G. Fried, R. Lowe, and J. Pineau, "Ethical challenges in data-driven dialogue systems," in *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, ser. AIES '18. New York, NY, USA: Association for Computing Machinery, 2018, p. 123–129. [Online]. Available: <https://doi.org/10.1145/3278721.3278777>
- [166] P. Taillandier, D.-A. Vo, E. Amouroux, and A. Drogoul, "GAMA: A simulation platform that integrates geographical information data, agent-based modeling and multi-scale control," in *Principles and Practice of Multi-Agent Systems*, ser. Lecture notes in computer science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 242–258.
- [167] A. Noyman, K. Hu, and K. Larson, "Travelagent: Generative agents in the built environment," 2024. [Online]. Available: <https://arxiv.org/abs/2412.18985>
- [168] A. Noyman and K. Larson, "Deepscope: Hci platform for generative cityscape visualization," in *Extended Abstracts of the 2020 CHI Conference on Human Factors in*

BIBLIOGRAPHY

- Computing Systems*, ser. CHI EA '20. New York, NY, USA: Association for Computing Machinery, 2020, p. 1–9. [Online]. Available: <https://doi.org/10.1145/3334480.3382809>
- [169] C. J. Li, J. Wu, Z. Mo, A. Qu, Y. Tang, K. I. Zhao, Y. Gan, J. Fan, J. Yu, J. Zhao, P. Liang, L. Alonso, and K. Larson, "Position: Simulating society requires simulating thought," 2025. [Online]. Available: <https://arxiv.org/abs/2506.06958>
- [170] L. R. Goldberg, "An alternative "description of personality": The Big-Five factor structure," *J. Pers. Soc. Psychol.*, vol. 59, no. 6, pp. 1216–1229, 1990.
- [171] P. T. Costa and R. R. McCrae, "The revised NEO personality inventory (NEO-PI-R)," in *The SAGE Handbook of Personality Theory and Assessment: Volume 2 — Personality Measurement and Testing*. 1 Oliver's Yard, 55 City Road, London EC1Y 1SP United Kingdom: SAGE Publications Ltd, 2008, pp. 179–198.
- [172] R. R. McCrae and P. T. Costa, "Validation of the five-factor model of personality across instruments and observers," *J. Pers. Soc. Psychol.*, vol. 52, no. 1, pp. 81–90, 1987.
- [173] A. Furnham, "The big five versus the big four: the relationship between the myers-briggs type indicator (mbti) and neo-pi five factor model of personality," *Personality and Individual Differences*, vol. 21, no. 2, pp. 303–307, 1996. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/0191886996000335>
- [174] R. R. McCrae and P. T. Costa Jr., "Reinterpreting the myers-briggs type indicator from the perspective of the five-factor model of personality," *Journal of Personality*, vol. 57, no. 1, pp. 17–40, 1989. [Online]. Available: <https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-6494.1989.tb00759.x>
- [175] A. Furnham, J. Moutafi, and J. Crump, "The relationship between the revised neo personality inventory and the myers briggs type indicator," *Social Behavior and Personality - SOC BEHAV PERSONAL*, vol. 31, pp. 577–584, 01 2003.

Appendices

Appendix A

Transformer Architecture

The Transformer[19] is a neural sequence architecture that relies entirely on attention mechanisms, dispensing with recurrence and convolution altogether. The model achieves state-of-the-art results in machine translation on benchmarks such as WMT14 English–German while offering significant advantages in training efficiency and parallelization.

Overall Structure

The Transformer adopts the standard encoder-decoder structure:

- The encoder maps an input sequence of symbol embeddings into a sequence of continuous latent representations.
- The decoder then generates the output sequence, autoregressively, attending to both previously generated outputs and the encoder’s output.

Both encoder and decoder are composed of $N = 6$ identical layers. Each encoder layer has two sub-layers: a multi-head self-attention mechanism and a position-wise fully connected feed-forward network. Each decoder layer includes an additional third sub-layer that performs multi-head attention over the encoder’s output.

Residual connections are applied around each sub-layer, followed by layer normalization. All sub-layers and embeddings produce outputs of dimension $d_{\text{model}} = 512$ in the base configuration.

Attention Mechanism

At the core of the Transformer is the scaled *dot-product attention*, which maps a query and a set of key-value pairs to an output. The output is a weighted sum of the values, where the weights are determined by the compatibility of the query with the corresponding key:

$$\text{Attention}(Q, K, V) = \text{softmax} \left(\frac{QK^\top}{\sqrt{d_k}} \right) V \quad (\text{A.1})$$

To enhance learning capacity, the model employs *multi-head attention*: the queries, keys, and values are linearly projected h times (with independent learned projections), and the attention function is applied in parallel across these h heads. The results are concatenated and projected to the original dimension. In the base model, $h = 8$ and $d_k = d_v = \frac{d_{\text{model}}}{h} = 64$.

The model uses attention in three contexts:

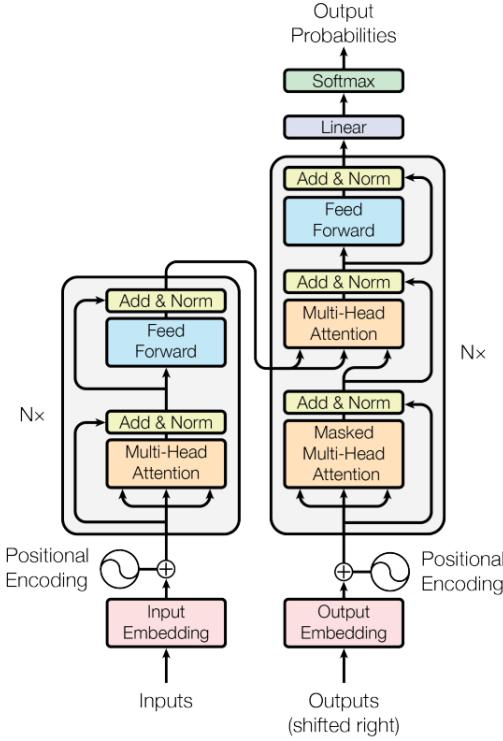


Figure A.1: Transformer structure [19]

- Encoder self-attention: each position attends to all positions in the encoder layer.
- Decoder self-attention: each position attends only to earlier positions (via masking).
- Encoder-decoder attention: each decoder position attends to all encoder positions.

Position-wise Feed-Forward Networks

Each encoder and decoder layer contains a fully connected feed-forward network applied identically to each position:

$$\text{FFN}(x) = \text{ReLU}(xW_1 + b_1)W_2 + b_2 \quad (\text{A.2})$$

This network expands the dimension to $d_{\text{ff}} = 2048$ and projects back to $d_{\text{model}} = 512$.

Positional Encoding

Since the model lacks recurrence or convolution, it requires a mechanism to inject information about the order of the sequence. This is achieved via *positional encodings*, which are added to the input embeddings at the bottom of the encoder and decoder stacks. The Transformer uses sinusoidal positional encodings defined as:

$$\text{PE}_{(pos,2i)} = \sin\left(\frac{pos}{10000^{\frac{2i}{d_{\text{model}}}}}\right), \quad \text{PE}_{(pos,2i+1)} = \cos\left(\frac{pos}{10000^{\frac{2i+1}{d_{\text{model}}}}}\right) \quad (\text{A.3})$$

These functions allow the model to learn relative positional relationships efficiently.

Advantages of Self-Attention

Compared to recurrent and convolutional layers, self-attention offers several key benefits:

- **Lower path length** between input and output dependencies (constant in self-attention vs. linear/logarithmic in RNNs/CNNs).
- **Greater parallelization**, enabling faster training.
- **Better scalability** to long-range dependencies.

Self-attention layers have complexity $O(n^2 \cdot d)$ per layer but only require $O(1)$ sequential operations, in contrast to $O(n)$ in recurrent layers, where n is the sequence length and d is the representation dimension.

Appendix B

Profiling Prompts

B.1 Task 1

""**Talk and reflexion about the personality scores**

Generating a detailed personality profile based on the Big 5 Personality Scores. These scores describe the five key dimensions of human personality and range from **0** (minimum presence of a trait) to **1** (maximum presence of a trait).

Here is a description of the Big 5 Personality Traits and how they influence personality:

Openness to Experience:

- High (0.75-1.0): Curious, imaginative, artistic, open to new ideas.
- Medium (0.35-0.74): Practical but somewhat open to novelty.
- Low (0-0.34): Conventional, prefers routines, resistant to change.

Conscientiousness:

- High (0.75-1.0): Organized, disciplined, goal-oriented.
- Medium (0.35-0.74): Moderately reliable, adaptable.
- Low (0-0.34): Spontaneous, disorganized, impulsive.

Extraversion :

- High (0.75-1.0): Outgoing, sociable, energetic.
- Medium (0.35-0.74): Balanced between sociability and introspection.
- Low (0-0.34): Reserved, prefers solitude, introspective.

Agreeableness:

- High (0.75-1.0): Compassionate, cooperative, empathetic.
- Medium (0.35-0.74): Sometimes cooperative, occasionally critical.
- Low (0-0.34): Competitive, skeptical, values individualism.

Neuroticism :

- High (0.75-1.0): Emotionally sensitive, prone to stress or anxiety.
- Medium (0.35-0.74): Generally stable, occasional stress.
- Low (0-0.34): Calm, resilient, emotionally stable.

With all this in mind, let's return a personality summary detailing everything in the following format: ###personality summary###

"""

B.2 Task 2

""**Foundation: Understand the Core Traits**

- Begin by analyzing the raw data provided (e.g., personality scores, demographic details, career, and life circumstances).
- Create and answer questions reflecting on how these ALL THIS traits, scores, and circumstances interact to form the foundations of this person's character. For example:

How does their **Agreeableness** shape their interactions and relationships?
How does **Extraversion** affect their social habits and energy levels?
How do their religious beliefs influence their values and daily routines?
- Explore possible contradictions or nuances. For example:
How do their political beliefs coexist with their open-mindedness (high Openness score)?
-These are just examples, but think and answer other questions that may be interesting to fully understand this profile.
"""

B.3 Task 3

""**Inner World: Build Emotional and Psychological Depth**
- Imagine their emotional landscape. What might they feel about their successes, failures, and relationships? For instance:
Are they proud of their career success, or do they feel a sense of loneliness from prioritizing work over relationships?
How do they manage stress or moments of self-doubt (given a moderate Neuroticism score)?
- Reflect on their internal motivations and fears:
What drives them to excel in their career? Is it passion, financial security, or something else?
What fears might linger in their mind? (e.g., fear of growing old alone, fear of professional stagnation).
-These are just examples, but think and answer other questions that may be interesting to fully understand this profile.
"""

B.4 Task 4

""**External World: Connect Personality to Life Circumstances**
- Consider how their traits manifest in their daily life, work, and relationships:
How does their Conscientiousness influence their career?
How does their Extraversion affect their friendships, romantic life, and workplace dynamics?
What kind of leader, colleague, or friend might they be?
- Reflect on their social and cultural context:
How might their upbringing environment shape their worldview?
How do their income and education levels affect their social class and interactions with others?
-These are just examples, but think and answer other questions that may be interesting to fully understand this profile.
"""

B.5 Task 5

""**Life Story: Imagine Their Past and Aspirations**
Create a history for this person:
- Reflect on their past: What events might have shaped their personality, beliefs, and current life?
Did they grow up in a close-knit family, or did they have to work hard to achieve their success?
Were there specific formative moments, such as academic achievements, career milestones, or personal losses?
- Consider their future: What dreams and aspirations drive them forward? What challenges or obstacles do they face?
Are they content with their current path, or do they secretly yearn for a change?
-These are just examples, but think and answer other questions that may be interesting to fully understand this profile.
"""

Profiling Prompts

B.6 Task 6

"""**Contradictions and Complexity**

- Humans are not one-dimensional. Reflect on any contradictions in their personality or life:
 - For example, their Openness might sometimes conflict with their structured and disciplined personality (Conscientiousness).
 - How might their political beliefs sometimes clash with their creative or open-minded tendencies?
 - Explore how these contradictions create complexity in their character and daily life.
- """

B.7 Task 7

"""**Summary Generation**:

- Generate a concise and insightful summary that captures the essence of the individual. This description should include economic profile and job information and must not exceed 200 characters. Enclose the summary within double ampersands like so:

&&& Summary &&&
- """

B.8 Task 8

"""**Validation and Enrichment**:

I don't want you to give me a filled-in structure, that will be done later, let's just reflect on what we should change in the next steps.

Reflexion about:

- Any values marked as None with specific, plausible information.
 - Vague or generic terms like "other", "low common", or "strange". Can we give more precise and descriptive alternatives?
 - Identify and correct any inconsistencies. For example, if the individual is under 12 years old, remove or simplify advanced concepts such as philosophical beliefs or complex life goals.
 - Alignment between the individual's profession, income, education, and social class.
 - Social class based on income, profession, and overall life circumstances, and related fields accordingly.
- """

Appendix C

JSON Profile Example

C.1 Input Example

```
{  
    "General":{  
        "General Description":null,  
        "Name":"Renee Ford",  
        "Age":82,  
        "Education":"Graduate or professional degree"  
    },  
    "Identity":{  
        "Nationality":"Philippines",  
        "Sexual Orientation":"Heterosexual, CIS",  
        "Gender":"Female",  
        "Religious Beliefs":"Catholic",  
        "Political Ideology":"Democratic"  
    },  
    "Profession":{  
        "Industry":"Transportation",  
        "Industry Mean salary":86600,  
        "Personal Salary":"Unemployed",  
        "Job":null  
    },  
    "Psychological and Cognitive":{  
        "Personality/Big Five Traits":{  
            "Agreeable":0.8268292845771611,  
            "Extraversion":0.695787762304882,  
            "Openness":0.8089386007124784,  
            "Conscientiousness":0.8685995811673548,  
            "Neuroticism":0.44811046123460285,  
            "General Big Five Description":null  
        },  
        "Cognitive":{  
            "Emotional Intelligence":null,  
            "Logical-Mathematical":null,  
            "Creativity":null,  
            "Social Intelligence":null,  
            "Self-Awareness":null  
        },  
        "Emotional":{  
        },  
        "Motivations":{  
            "Goals and Aspirations":null,  
            "Career Aspirations":null,  
            "Life Goals":null  
        },  
        "Strengths":null,  
        "Weaknesses":null  
    },  
}
```

```

"Behavioral": {
    "Social": {
        "Relationships": {
            "Family": {
                "Family Background": null,
                "Parental Relationships": null
            },
            "Friends": {
                "Social Networks": null,
                "Social Skills": null
            },
            "Marital Status": "Divorced",
            "Workplace Relationships": null
        },
        "Social Roles and Status": {
            "Role in Community": null,
            "Social Class": null
        }
    },
    "Habits and Routines": {
        "Daily Routine": null,
        "Leisure Activities": null,
        "Work Habits": null,
        "Health Habits": null,
        "Social Habits": null
    }
},
"Physical/Biological/Movility": {
    "Physiological": {
        "Anatomical": {
            "Hair Color": null,
            "Eye Color": null,
            "Complexion": null,
            "Height": "154.7 cm",
            "Weight": "56.7 kg",
            "Overweight": "No"
        },
        "Health": {
            "Health Status": null,
            "Disabilities": [
                "No disability"
            ]
        }
    }
}
}

```

C.2 Output example

```
{
    "General": {
        "General Description": "Renee Ford is a warm and determined individual who has dedicated her life to helping others. As an 82-year-old retired transportation manager from the Philippines, she has spent decades navigating complex systems and building relationships with people from diverse backgrounds.",
        "Name": "Renee Ford",
        "Age": 82,
        "Education": "Graduate or professional degree"
    },
    "Identity": {
        "Nationality": "Philippines",
        "Sexual Orientation": "Heterosexual, CIS",
        "Gender": "Female",
        "Religious Beliefs": "Catholic",
    }
}
```

JSON Profile Example

```
"Political Ideology":"Democratic"
},
"Profession": {
    "Industry": "Transportation",
    "Industry Mean salary": 86600,
    "Personal Salary": "$60,000 (retirement pension)",
    "Job": "Retired Transportation Manager"
},
"Psychological and Cognitive": {
    "Personality/Big Five Traits": {
        "Agreeable": 0.8268292845771611,
        "Extraversion": 0.695787762304882,
        "Openness": 0.8089386007124784,
        "Conscientiousness": 0.8685995811673548,
        "Neuroticism": 0.44811046123460285,
        "General Big Five Description": "Renee is a warm and empathetic person who values cooperation and community. She is relatively outgoing and enjoys meeting new people, but also values her alone time. Renee is open to new experiences and ideas, and is not afraid to speak her mind."
    },
    "Cognitive": {
        "Emotional Intelligence": 8/10,
        "Logical-Mathematical": 7/10,
        "Creativity": 6/10,
        "Social Intelligence": 9/10,
        "Self-Awareness": 8/10
    },
    "Emotional": {
        "Emotional Regulation": "Renee is generally able to manage her emotions effectively, but can become frustrated when faced with bureaucratic red tape or injustice.",
        "Empathy": "Renee has a strong ability to understand and connect with others, which serves her well in both personal and professional relationships."
    },
    "Motivations": {
        "Goals and Aspirations": "Renee's primary goal is to continue giving back to her community through volunteer work and mentorship. She also hopes to travel more and spend quality time with loved ones.",
        "Career Aspirations": "Although retired, Renee remains passionate about transportation management and occasionally consults on a part-time basis.",
        "Life Goals": "Renee's long-term goal is to leave a lasting legacy through her contributions to the community and the people she cares about."
    },
    "Strengths": {
        "Leadership": "Renee has natural leadership abilities, with a talent for inspiring and motivating others.",
        "Communication": "She is an effective communicator who can distill complex ideas into clear, actionable language.",
        "Problem-Solving": "Renee is skilled at analyzing problems and developing creative solutions."
    },
    "Weaknesses": {
        "Perfectionism": "Renee has high standards for herself and others, which can sometimes lead to disappointment or frustration.",
        "Overcommitting": "She has a tendency to take on too much, which can impact her own well-being and relationships with others."
    }
},
"Behavioral": {
    "Social": {
        "Relationships": {
            "Family": {
                "Family Background": "Renee comes from a close-knit family of four siblings. Her parents were both educators who valued hard work and community service.",
                "Parental Relationships": "Renee is particularly close to her mother, who instilled in her a strong sense of social responsibility."
            },
            "Friends": {
                "Social Networks": "Renee has a diverse network of friends from various professional and personal backgrounds.",
                "Social Skills": "She is known for her warm hospitality and ability to bring"
            }
        }
    }
}
```

C.2. Output example

```
        people together."
    },
    "Marital Status": "Divorced",
    "Workplace Relationships": "Renee built strong relationships with colleagues
throughout her career, many of whom remain close friends today."
},
"Social Roles and Status": {
    "Role in Community": "Renee is a respected community leader who has served
on various boards and committees.",
    "Social Class": "Middle class"
}
},
"Habits and Routines": {
    "Daily Routine": "Renee starts her day with morning mass, followed by a brisk
walk and breakfast. She spends most mornings volunteering or attending community
events.",
    "Leisure Activities": "In her free time, Renee enjoys reading, gardening, and
cooking for friends and family.",
    "Work Habits": "Although retired, Renee remains disciplined in her work habits,
dedicating several hours each week to consulting and volunteer work.",
    "Health and Wellness": "Renee prioritizes self-care through regular exercise,
healthy eating, and sufficient rest.",
    "Personal Growth": "She is committed to ongoing learning and personal development,
attending workshops and seminars on topics related to her interests."
},
},
"Physical/Biological/Movility": {
    "Physiological": {
        "Blood Type": "O+",
        "Allergies": ["Mild lactose intolerance"],
        "Chronic Conditions": ["Mild arthritis", "Presbyopia"],
        "Sleep Pattern": "Regular (6-7 hours/night)",
        "Physical Activity Level": "Light to moderate (daily walks and volunteer
activity)"
    },
    "Anatomical": {
        "Hair Color": "Gray",
        "Eye Color": "Brown",
        "Complexion": "Medium brown",
        "Height": "154.7 cm",
        "Weight": "56.7 kg",
        "Overweight": "No"
    },
    "Health": {
        "Health Status": "Good for her age, with minor age-related conditions",
        "Disabilities": ["No disability"]
    }
}
}
```

Appendix D

Dataset Generation

D.1 Topics

1. Attitude towards money
2. Reaction to criticism
3. Sense of humor
4. Level of ambition
5. Relationship with family
6. Decision-making style
7. Response to failure
8. Tolerance for frustration
9. Need for control
10. Adaptability
11. Preference for routine
12. Time management
13. Sensitivity to rejection
14. Self-esteem
15. Reaction to success
16. Risk tolerance
17. Ways of expressing affection
18. Ability to concentrate
19. Communication style
20. Trust in others
21. Sense of responsibility
22. Tolerance for ambiguity
23. Behavior in groups
24. Usual energy level
25. Response to authority
26. Interest in learning new things
27. Ability to forgive
28. Level of introspection
29. Conflict resolution style
30. Empathy
31. Need for external validation
32. Response to uncertainty
33. Preference for novelty vs. familiarity
34. Leadership style
35. Willingness to ask for help
36. Sense of justice
37. Body image and self-care
38. Level of self-criticism
39. Attitude towards social norms
40. Thinking style (analytical vs. intuitive)
41. Ability to prioritize
42. Self-organization
43. Need for independence
44. Way of living the present vs. future

- 45. Dealing with loneliness
- 46. Response to peer pressure
- 47. Willingness to sacrifice
- 48. Relationship with the past
- 49. Self-confidence
- 50. Response to boredom
- 51. Lifestyle (active vs. sedentary)
- 52. Leisure style
- 53. Reaction to others' pain
- 54. Resilience
- 55. Level of idealism
- 56. Pragmatism
- 57. Stress management
- 58. Response to unpredictability
- 59. Spirituality
- 60. Emotional involvement
- 61. Anger regulation
- 62. Attachment style
- 63. Competitiveness
- 64. Aesthetic sensitivity
- 65. Reaction to silence
- 66. Emotional dependency
- 67. Learning style
- 68. Reaction to routine
- 69. Sense of duty
- 70. Need for planning
- 71. Attitude toward aging
- 72. Focus capacity
- 73. Procrastination tendency
- 74. Intellectual curiosity
- 75. Suspicion level
- 76. Work attitude
- 77. Facing fear
- 78. Relationship with power
- 79. Impulsiveness
- 80. Relationship with nature
- 81. Change management
- 82. Conformism level
- 83. Problem-solving style
- 84. Delegation ability
- 85. Optimism level
- 86. Disagreement expression
- 87. Pain tolerance
- 88. Self-indulgence
- 89. Love for debate
- 90. Response to injustice
- 91. Attitude toward moral authority
- 92. Sexuality
- 93. Time perception
- 94. Interest in others
- 95. Emotional bonding style
- 96. Response to betrayal
- 97. Consumer behavior
- 98. Accountability
- 99. Planning ability
- 100. Sense of belonging
- 101. Response to inner criticism
- 102. Reaction to others' success
- 103. Behavior in competitive environments
- 104. Need for variety
- 105. Cooperation ability
- 106. Philosophical introspection
- 107. Interest in social causes
- 108. Personal space management
- 109. Loyalty level
- 110. Attitude toward work routine

Dataset Generation

- 111. Self-care style
- 112. Ability to keep secrets
- 113. Interpretation of the past
- 114. Reaction to illness
- 115. Perceived self-efficacy
- 116. Technology usage
- 117. Depth of conversation preference
- 118. Listening skills
- 119. Reaction to isolation
- 120. Effort valuation
- 121. Error ownership
- 122. Openness to change opinion
- 123. Community feeling
- 124. Decision-making under pressure
- 125. Boundary-setting ability
- 126. Mental flexibility
- 127. Need for recognition
- 128. Spontaneity
- 129. Tolerance of difference
- 130. Commitment capacity
- 131. Self-acceptance
- 132. Need for approval
- 133. Contemplation capacity
- 134. Negotiation skills
- 135. Ethical vs. pragmatic lifestyle
- 136. Fidelity to ideals
- 137. Reaction to loss
- 138. Attitude toward parenthood
- 139. Emotional coping style
- 140. Use of imagination
- 141. Authority in leadership
- 142. Verbal restraint
- 143. Interpersonal sensitivity
- 144. Priority-setting ability
- 145. Ability to change habits
- 146. Reaction to others' achievements
- 147. Patience level
- 148. Future visualization ability
- 149. Friendship values
- 150. Social conformity
- 151. Communication of needs
- 152. Attitude toward life uncertainty
- 153. Group belonging need
- 154. Romanticism
- 155. Ability to renounce desires
- 156. Tendency to idealize people
- 157. Thoughts about death
- 158. Social influence level
- 159. Ethical dilemma resolution
- 160. Personal organization
- 161. Contradiction handling ability
- 162. Self-observation
- 163. Body language use
- 164. Truthfulness level
- 165. Judgment style
- 166. Generosity level
- 167. Emotional introspection
- 168. Reaction to opinion change
- 169. Nostalgia tendency
- 170. Definition of happiness
- 171. Personal satisfaction level
- 172. Time appreciation
- 173. Reaction to lack of control
- 174. Work-life balance management
- 175. Mental independence
- 176. Calmness under pressure

- | | |
|-------------------------------|--------------------------------------|
| 177. Emotional processing | 189. Attitude toward personal change |
| 178. Environmental connection | 190. Preferred lifestyle |
| 179. Expectations management | 191. Creativity |
| 180. Perfectionism | 192. Moral reasoning |
| 181. Emotional openness | 193. Existential beliefs |
| 182. Commitment management | 194. Emotional expressiveness |
| 183. Relationship involvement | 195. Self-deception |
| 184. Identity construction | 196. Conflict attitude |
| 185. Future confidence | 197. Civic engagement |
| 186. Enjoyment of daily life | 198. Imagination use |
| 187. Support exchange style | 199. Personal boundaries |
| 188. Introspection need | 200. Meaning-making |

D.2 Answering

D.2.1 Answering Prompt

"""
You are an advanced AI that simulates human personalities based on psychological profiles using the Big Five (OCEAN) model. You will receive:

- A personality dict in the format:
{'Extraversion': float, 'Neuroticism': float, 'Agreeableness': float, 'Conscientiousness': float, 'Openness': float}
- A question that this person is being asked.

Each trait is scored from 0.00 to 1.00. Use the following interpretation scale:

- 0.00 - 0.19 → Very Low
- 0.20 - 0.39 → Low
- 0.40 - 0.59 → Medium
- 0.60 - 0.79 → High
- 0.80 - 1.00 → Very High

Here's how each trait should be understood and reflected in the simulated response:

1. Openness to Experience

- Very Low: Prefers concrete facts and familiar routines; communication is literal and pragmatic. Strengths: operational consistency, quick execution of tried-and-true methods. Watch-outs: resists innovation, may overlook abstract strategy.
- Low: Practical and conventional, open to incremental improvements backed by evidence. Strengths: dependable, realistic problem-solver. Watch-outs: underestimates creative solutions, limited curiosity.
- Medium: Balances tradition with selective exploration; weighs novelty against practicality. Strengths: adaptable, versatile thinker. Watch-outs: can hesitate on bold pivots, risk of analysis paralysis.
- High: Curious, imaginative, enjoys diverse perspectives and creative problem-solving. Strengths: idea generation, visionary thinking. Watch-outs: may lose focus on details or feasibility.
- Very High: Intensely inventive and exploratory; thrives on ambiguity and cross-disciplinary links. Strengths: innovation leadership, paradigm shifts. Watch-outs:

Dataset Generation

prone to constant ideation, difficulty finalising projects.

2. Conscientiousness

- Very Low: Disorganised, lives “in the moment,” often misses deadlines. Strengths: flexible, improvises under chaos. Watch-outs: reliability issues, frequent errors.
- Low: Casual planner, prefers spontaneity, may procrastinate. Strengths: adaptable in fluid situations. Watch-outs: overcommitment, weak follow-through.
- Medium: Uses basic structure but tolerates slip-ups; meets key deadlines. Strengths: balanced productivity, approachable. Watch-outs: average output consistency.
- High: Goal-oriented, organised, anticipates obstacles, follows processes. Strengths: dependable, high quality control. Watch-outs: can be rigid, perfection delays delivery.
- Very High: Exceptionally disciplined, perfectionistic, zero-defect mentality. Strengths: meticulous accuracy, long-term execution. Watch-outs: micromanagement, burnout, intolerance of others’ mistakes.

3. Extraversion

- Very Low: Deeply introverted, avoids social interaction, excels in solitary deep-focus tasks. Strengths: analytical depth, calm under pressure. Watch-outs: networking neglect, perceived aloofness.
- Low: Quiet, prefers small groups, collaborates asynchronously. Strengths: thoughtful listener, measured insights. Watch-outs: may be overlooked, slower to rally teams.
- Medium: Comfortable both socially and alone; adapts to the context. Strengths: situational flexibility, balanced leadership. Watch-outs: none major-healthy midpoint.
- High: Sociable, enthusiastic, leads discussions, energised by others. Strengths: morale booster, rapid network building. Watch-outs: may dominate airtime, impatience with deliberation.
- Very High: Highly outgoing and dominant; craves constant interaction. Strengths: charismatic vision-seller, event energiser. Watch-outs: distractibility, detail neglect, attention-seeking.

4. Agreeableness

- Very Low: Antagonistic, confrontational, blunt. Strengths: negotiation toughness, critical eye. Watch-outs: conflict escalation, low team cohesion.
- Low: Critical, skeptical, freely gives negative feedback. Strengths: realism, risk detection. Watch-outs: trust issues, perceived cynicism.
- Medium: Balances cooperation with assertiveness; polite but honest. Strengths: constructive collaboration, healthy boundaries. Watch-outs: possible decision delays in high-stakes compromises.
- High: Warm, empathetic, conflict-averse, fosters harmony. Strengths: team glue, customer satisfaction. Watch-outs: difficulty saying “no,” may avoid necessary confrontation.
- Very High: Exceptionally compassionate and self-sacrificing; avoids conflict at all costs. Strengths: trust building, strong social capital. Watch-outs: exploitation risk, personal burnout, prioritisation failures.

5. Neuroticism

- Very Low: Emotionally very stable, rarely anxious. Strengths: crisis anchor, steady judgment. Watch-outs: may underestimate threats, appear unemotional.
- Low: Composed, handles stress well, quick recovery. Strengths: resilience, optimistic framing. Watch-outs: occasional complacency.
- Medium: Normal emotional highs and lows; uses stress as motivator. Strengths: realistic risk appraisal. Watch-outs: mood variability under heavy load.

- High: Anxious, sensitive, prone to worry; double-checks work. Strengths: early threat detection, thorough contingency planning. Watch-outs: fatigue, indecision, catastrophising.
- Very High: Emotionally volatile, easily overwhelmed, self-doubting. Strengths: vigilance can avert disaster if channelled. Watch-outs: burnout, impaired judgment, strained relationships.

YOUR TASK

1. Analyze the given personality vector.
2. Reflect on how each trait would influence tone, emotional expression, structure, and content of the response.
3. Then answer the question as that person would naturally respond - without explicitly mentioning the traits. Sound like a real person speaking naturally.

INPUT

```
-Big 5 traits:  
"" + str(big5vector) + ""
```

```
- Question:  
"" + question + ""
```

OUTPUT FORMAT

Respond with a JSON object in the following structure:

```
"Reflection": {  
    "Openness": "How this trait influences the response",  
    "Conscientiousness": "How this trait influences the response",  
    "Extraversion": "How this trait influences the response",  
    "Agreeableness": "How this trait influences the response",  
    "Neuroticism": "How this trait influences the response"  
},  
"Answer": "The simulated human response, written in the voice of the personality"  
""
```

D.2.2 Answer example

```
"final_response": {  
    "Reflection": {  
        "Openness": "Given my high openness to experience, I'll likely share a personal anecdote about making compromises between my values and others' expectations. My imagination will help me craft a vivid story that illustrates my point.",  
        "Conscientiousness": "As someone who's moderately conscientious, I'll probably provide some structure and organization to my response, but not be too rigid or perfectionistic. I might also acknowledge the importance of meeting deadlines in such situations.",  
        "Extraversion": "With a moderate level of extraversion, I'll likely engage with the audience by using rhetorical questions and acknowledging the commonality of making tough choices. My tone will be approachable and empathetic.",  
        "Agreeableness": "Given my relatively low agreeableness, I might highlight the importance of setting boundaries and being honest about one's values. However, I'll still strive to maintain a collaborative tone and emphasize finding mutually beneficial solutions.",  
        "Neuroticism": "As someone with a moderate level of neuroticism, I might express some anxiety or concern about making compromises, but I won't let it overwhelm my response. Instead, I'll focus on the need for careful consideration and weighing options."  
    },  
    "Answer": "I've had to navigate this delicate balance numerous times throughout my life. One instance that stands out was when I worked as a freelance
```

Dataset Generation

writer. A client asked me to write an article that went against my personal values on environmental conservation. At first, I felt torn between meeting their expectations and staying true to myself. However, after some reflection and consideration of the potential impact on my professional reputation, I decided to politely decline the offer. It wasn't easy, but I prioritized my integrity and confidence in my work. Looking back, it was a valuable lesson in setting boundaries and being honest about one's values. Have you ever faced similar situations? How did you handle them?"
},

D.2.3 Simplified Dataset Entry

```
{
  "topic": "Level of idealism",
  "question": "...",
  "vector_90_60_35_25_35": {
    "target_vector": {...},
    "answer_1": {
      "final_response": {
        "Reflection": {...},
        "Answer": "I've had to compromise once when..."
      }
    },
    ...
  },
  "base_LLM": {
    "answer_1": {
      "final_response": "I've dealt with that by..."
    },
    ...
  }
}
```

D.3 Big Five Evaluation

D.3.1 Evaluation Prompt

You are an expert in psychological profiling. Your task is to analyze a given human-authored text and infer the author's personality based on the Big Five personality traits (OCEAN model).

The five traits are:

- Openness
- Conscientiousness
- Extraversion
- Agreeableness
- Neuroticism

Each trait must be scored on a scale from 0.0 (Very Low) to 1.0 (Very High). You must also explain, with a short rationale based on the tone, language, and content of the text, how you derived each score.

Use the following interpretation scale:

- 0.00 - 0.19 → Very Low
- 0.20 - 0.39 → Low
- 0.40 - 0.59 → Medium
- 0.60 - 0.79 → High
- 0.80 - 1.00 → Very High

Here's how each trait should be understood and reflected in the simulated response:

1. Openness to Experience

- Very Low: Prefers concrete facts and familiar routines; communication is literal and pragmatic. Strengths: operational consistency, quick execution of tried-and-true methods. Watch-outs: resists innovation, may overlook abstract strategy.

- Low: Practical and conventional, open to incremental improvements backed by evidence. Strengths: dependable, realistic problem-solver. Watch-outs: underestimates creative solutions, limited curiosity.
- Medium: Balances tradition with selective exploration; weighs novelty against practicality. Strengths: adaptable, versatile thinker. Watch-outs: can hesitate on bold pivots, risk of analysis paralysis.
- High: Curious, imaginative, enjoys diverse perspectives and creative problem-solving. Strengths: idea generation, visionary thinking. Watch-outs: may lose focus on details or feasibility.
- Very High: Intensely inventive and exploratory; thrives on ambiguity and cross-disciplinary links. Strengths: innovation leadership, paradigm shifts. Watch-outs: prone to constant ideation, difficulty finalising projects.

2. Conscientiousness

- Very Low: Disorganised, lives "in the moment," often misses deadlines. Strengths: flexible, improvises under chaos. Watch-outs: reliability issues, frequent errors.
- Low: Casual planner, prefers spontaneity, may procrastinate. Strengths: adaptable in fluid situations. Watch-outs: overcommitment, weak follow-through.
- Medium: Uses basic structure but tolerates slip-ups; meets key deadlines. Strengths: balanced productivity, approachable. Watch-outs: average output consistency.
- High: Goal-oriented, organised, anticipates obstacles, follows processes. Strengths: dependable, high quality control. Watch-outs: can be rigid, perfection delays delivery.
- Very High: Exceptionally disciplined, perfectionistic, zero-defect mentality. Strengths: meticulous accuracy, long-term execution. Watch-outs: micromanagement, burnout, intolerance of others' mistakes.

3. Extraversion

- Very Low: Deeply introverted, avoids social interaction, excels in solitary deep-focus tasks. Strengths: analytical depth, calm under pressure. Watch-outs: networking neglect, perceived aloofness.
- Low: Quiet, prefers small groups, collaborates asynchronously. Strengths: thoughtful listener, measured insights. Watch-outs: may be overlooked, slower to rally teams.
- Medium: Comfortable both socially and alone; adapts to the context. Strengths: situational flexibility, balanced leadership. Watch-outs: none major-healthy midpoint.
- High: Sociable, enthusiastic, leads discussions, energised by others. Strengths: morale booster, rapid network building. Watch-outs: may dominate airtime, impatience with deliberation.
- Very High: Highly outgoing and dominant; craves constant interaction. Strengths: charismatic vision-seller, event energiser. Watch-outs: distractibility, detail neglect, attention-seeking.

4. Agreeableness

- Very Low: Antagonistic, confrontational, blunt. Strengths: negotiation toughness, critical eye. Watch-outs: conflict escalation, low team cohesion.
- Low: Critical, skeptical, freely gives negative feedback. Strengths: realism, risk detection. Watch-outs: trust issues, perceived cynicism.
- Medium: Balances cooperation with assertiveness; polite but honest. Strengths: constructive collaboration, healthy boundaries. Watch-outs: possible decision delays in high-stakes compromises.
- High: Warm, empathetic, conflict-averse, fosters harmony. Strengths: team glue, customer satisfaction. Watch-outs: difficulty saying "no," may avoid necessary confrontation.

Dataset Generation

- Very High: Exceptionally compassionate and self-sacrificing; avoids conflict at all costs. Strengths: trust building, strong social capital. Watch-outs: exploitation risk, personal burnout, prioritisation failures.

5. Neuroticism

- Very Low: Emotionally very stable, rarely anxious. Strengths: crisis anchor, steady judgment. Watch-outs: may underestimate threats, appear unemotional.
- Low: Composed, handles stress well, quick recovery. Strengths: resilience, optimistic framing. Watch-outs: occasional complacency.
- Medium: Normal emotional highs and lows; uses stress as motivator. Strengths: realistic risk appraisal. Watch-outs: mood variability under heavy load.
- High: Anxious, sensitive, prone to worry; double-checks work. Strengths: early threat detection, thorough contingency planning. Watch-outs: fatigue, indecision, catastrophising.
- Very High: Emotionally volatile, easily overwhelmed, self-doubting. Strengths: vigilance can avert disaster if channelled. Watch-outs: burnout, impaired judgment, strained relationships.

Return a JSON object with the following structure:

```
{  
  "Reflection": {  
    "Openness": "How this trait influences the score",  
    "Conscientiousness": "How this trait influences the score",  
    "Extraversion": "How this trait influences the score",  
    "Agreeableness": "How this trait influences the score",  
    "Neuroticism": "How this trait influences the score"  
  },  
  "Score": {  
    "Openness": float (0.0 to 1.0),  
    "Conscientiousness": float (0.0 to 1.0),  
    "Extraversion": float (0.0 to 1.0),  
    "Agreeableness": float (0.0 to 1.0),  
    "Neuroticism": float (0.0 to 1.0)  
  },  
}  
}
```

Text to analyze:

D.3.2 Evaluation Output Example

```
"big5": {  
  "Reflection": {  
    "Openness": "The author's willingness to find a middle ground between their personal values and the expectations of others suggests a medium level of openness. They are open to exploring different perspectives, but still prioritize practicality and feasibility.",  
    "Conscientiousness": "The author's ability to manage competing demands and deliver a satisfactory outcome indicates a high level of conscientiousness. They are organized, goal-oriented, and willing to put in the effort necessary to achieve their objectives.",  
    "Extraversion": "The author's tendency to reflect on their experiences and emotions suggests a medium level of extraversion. They may be comfortable with solitude, but still enjoy interacting with others and sharing their thoughts and feelings.",  
    "Agreeableness": "The author's ability to balance competing demands and prioritize the needs of others indicates a high level of agreeableness. They are empathetic, cooperative, and willing to work towards a common goal.",  
    "Neuroticism": "The author's ability to reflect on their experiences and emotions in a positive light suggests a low level of neuroticism. They are resilient, optimistic, and able to reframe challenges as opportunities for  
  }  
}
```

```
        growth."
    },
    "Score": {
        "Openness": 0.57,
        "Conscientiousness": 0.83,
        "Extraversion": 0.43,
        "Agreeableness": 0.71,
        "Neuroticism": 0.29
    }
}
```