Universiteit Leiden

Master Computer Science

Big Five Personality Trait Prediction from Text Using Language Technologies

Name:	Shuang Fan
Student ID:	s3505847
Date:	28/08/2024
Specialisation:	Data Science
1st supervisor:	Dr.ir. Joost Broekens
2nd supervisor:	Prof.Dr. Suzan Verberne

Master's Thesis in Computer Science

Leiden Institute of Advanced Computer Science (LIACS)
Leiden University
Niels Bohrweg 1
2333 CA Leiden
The Netherlands

Abstract

The prediction of personality traits from text data has significant implications for fields such as human resources, psychological assessment, and personalized content delivery. This study explores the efficacy of BERT models in predicting the Big Five personality traits using a subset of the myPersonality dataset, which was augmented with synthetic data generated by GPT-4. Initial experiments with the original dataset revealed significant overfitting, as indicated by high Mean Squared Error (MSE) and low R^2 scores. Additional experiments using the IMDB dataset, while not directly related to personality, served as a benchmark to understand the data requirements for BERT's effective training. Then we augmented the dataset, resulting in improved performance, particularly on larger test sets. However, when evaluated on unseen subjects, the model's overfitting persisted, suggesting limitations in generalization. These experiments highlighted the necessity of a sufficiently large and diverse dataset for robust model performance. Our findings underscore the critical role of dataset size and diversity in training reliable personality prediction models and the potential of data augmentation to enhance generalization. Future research should refine augmentation techniques and explore more diverse datasets to further improve model robustness and applicability.

Contents

1 Introduction

The analysis of personality traits through automated methods holds significant potential for improving professional interactions, particularly in fields like recruitment and human resources [8]. The ability to infer personality traits from text data, such as CVs and LinkedIn profiles, offers companies a powerful tool for assessing candidates more efficiently and accurately. This technology can lead to more informed hiring decisions, better job-person fit, and ultimately, enhanced workplace dynamics [16]. In particular, the Big Five personality traits model [12], encompassing openness, conscientiousness, extraversion, agreeableness, and neuroticism, provides a well-established framework for characterizing human personality. By using this model, researchers can systematically study and understand personality traits.

Recent advancements in natural language processing (NLP), specifically the emergence of transformer models like BERT [6], have revolutionized the ability to extract meaning from textual data. This has significant implications for social signal processing, where understanding the subtle cues in human communication can greatly improve interaction dynamics. The application of NLP to analyze texts such as CVs and LinkedIn profiles to infer personality traits better help the companies to understand how it works, and how well it works.

Prior work has indicated the viability of using textual analysis for personality assessment [24], but the integration of robust machine learning models like BERT to enhance prediction accuracy in professional profiles is still relatively unexplored, particularly in conjunction with advanced feature extraction techniques [2]. Our research intends to fill this gap, investigating how BERT perform on small datasets. The questions we address through this work include: How accurately can a BERT model predict personality traits from professional profiles when only limited data is available?

In this work, we use a subset of the myPersonality dataset comprising 10,000 samples, enriched with additional annotated data, to predict the Big Five personality scores from the textual content of CVs and LinkedIn profiles. However, the myPersonality subset consists only of 250 participants, leading to a limited number of samples and scores. Additionally, many of these samples are extremely short, containing only one or two words, which complicates the task of predicting personality traits. These limitations highlight the importance of exploring strategies to mitigate overfitting and enhance model performance.

Our approach involves first fine-tuning the BERT model using the myPersonality subset. Given the challenges of working with small datasets, we explore various data augmentation methods to counter overfitting. To validate our approach, we also conducted preliminary experiments using the IMDB dataset, a larger and more established dataset, to assess the feasibility of fine-tuning the BERT model before applying these techniques to our target data.

1.1 Problem Statement

The Big Five personality traits model—often referred to as the OCEAN model [21], encompassing openness, conscientiousness, extraversion, agreeableness, and neuroticism—has been foundational in personality research, offering critical insights into human behavior and

text	OPN	EXT	CONS	NEUR	ARG
likes the sound of thunder	4.40	2.65	3.25	3.00	3.15
little things give you away.	4.60	2.15	2.90	2.15	4.10

Table 1: Example of myPersonality dataset

interpersonal dynamics. Prior research has extensively used the myPersonality dataset to predict these traits using various machine learning approaches [23], demonstrating notable success, particularly with deep learning models such as CNN+LSTM [24]. Despite these advancements, the generalizability and accuracy of personality predictions from text remain constrained by data availability and quality.

The enactment of GDPR in 2018 significantly restricted access to comprehensive personality datasets like myPersonality, which previously facilitated extensive studies with its rich, annotated data. This limitation poses a significant challenge for advancing research in this field, as the current subset of the dataset—comprising only 10,000 statuses from 250 users—restricts the depth and diversity of training data. Consequently, the robustness and scalability of models trained on this limited dataset are potentially compromised.

This project proposes to use the capabilities of BERT, a state-of-the-art NLP model, to enhance the prediction accuracy of Big Five personality traits from professional texts such as LinkedIn profiles and CVs based on small dataset training. By training BERT on the available myPersonality subset, this study aims to explore how effectively modern NLP techniques can bridge the gap left by the absence of larger, more diverse datasets.

The problem we are addressing involves predicting personality traits based on textual data using machine learning. Specifically, the task is to predict the Big Five personality traits: Openness, Conscientiousness, Extraversion, Agreeableness, and Neuroticism (OCEAN) from user-generated text samples. The input to the model is a text sample from the myPersonality dataset. The label space consists of five continuous numeric values, one for each of the Big Five personality traits. These values represent the degree to which each personality trait is exhibited by the user who generated the text. This setup defines a multi-output regression task, where the goal is to predict five separate continuous outputs (one for each personality trait) based on the input text.

Furthermore, recognizing the inherent challenges in collecting new, compliant personality data, this project also proposes the development of an artificial dataset using large language models (LLM) [9]. This initiative aims to replicate the depth and variety of the original myPersonality dataset, thereby providing a sustainable solution for ongoing research in personality prediction using textual analysis.

This dual approach not only addresses the immediate challenges posed by data limitations but also sets the groundwork for more robust, salable models that are based on modern LLM-based data augmentation techniques. This research contributes to the fields of sentiment analysis and natural language processing by demonstrating the applicability of advanced NLP techniques in real-world scenarios and by innovating in the ethical creation and use of synthetic data for sensitive applications like personality assessment.

We address our main research questions from proposal. Also there are sub-questions from the process of experiments. The main research question of this work is as follows:

1. **How valid and reliable is BERT in predicting personality traits based on self-written texts in CVs and LinkedIn profiles?**
    - This question aims to evaluate the capability of BERT model when applied to the task of inferring personality traits from written content typically found in resumes and social media profiles.
2. **To what extent can BERT capture the relationship between text and OCEAN scores within the myPersonality dataset?**
3. **To what extent can BERT capture the relationship between text and labels in a larger and more comprehensive text-to-score dataset?**
    - We will explore whether BERT can effectively understand and predict continuous scores from text in a dataset known for its clear relationships between text and sentiment, such as the IMDB dataset. Additionally, it seeks to determine what dataset size is sufficient to train BERT without causing overfitting, ensuring robust model performance.
4. **What is the effect of data augmentation on the BERT model's performance?**
    - We would investigate this by exploring three methods: adding Gaussian noise to the data, aggregating data according to user ID, and augmenting the dataset using large language models (LLM). Our goal was to determine how these different augmentation techniques impact the model's ability to learn and converge during training, particularly in comparison to the original dataset.

1.2 Contributions

We fine-tune BERT models to extract the relation between text and the big five model scores in the myPersonality dataset: BERT model with one linear activation function layer, fixed weights BERT model with one activation function layer and fixed weights BERT model with two linear layers to extract the relation between text and big five model scores in myPersonality dataset and compare their quality.

We train three models on three dataset generated from different augmentation treatments of myPersonality dataset. Our results show that glue the text based on user will not help to converge. Neither do the adding Gaussian noise to the scores.

Additionally, We replicate MLphile's work[1] of classification task on the IMDB dataset and fine-tune the BERT model with a regression task. The model converged without overfitting. This experiment demonstrates that BERT can converge without overfitting when trained on sufficiently large datasets, allowing us to identify an optimal dataset size (approximately 10k samples) that effectively balances model complexity and generalization.

[1] https://github.com/MLphile/BERT_on_Movie_Reviews?tab=readme-ov-file

To address the challenge of limited data in personality prediction, we augment the myPersonality dataset using the ChatGPT-4 model, expanding it to the identified optimal size. This augmented dataset was then employed to train the BERT model, significantly improving its performance.

In conclusion, we present a robust methodology for mitigating overfitting in regression tasks involving BERT models, particularly in scenarios where the number of data points is limited. Our approach highlights the importance of both model architecture and data augmentation in achieving reliable and generalizable personality predictions from text.

1.3 Thesis Structure

The remainder of this work is organized as follows: Section 2 introduces the background and related work, including an overview of the Big Five personality traits model and recent advancements in NLP with a focus on BERT. Section 3 provides details about the myPersonality dataset and IMDB dataset, including a preliminary analysis and statistics. Section 4 explains our proposed methods in detail, including the fine-tuning of the three BERT models and the data augmentation techniques employed. Section 5 presents our experiments and results, including the fine-tuning BERT model results, replication of MLphile's work on the IMDB dataset and the identification of an optimal dataset size. Finally, we address our research questions, discuss the implications of our findings, and outline potential improvements for future work in Section 6, concluding the thesis in Section 7.

2 Background and Related Work

2.1 The Big Five Personality Traits Model

The Big Five Personality Traits Model, also known as the Five Factor Model (FFM), is one of the most widely accepted frameworks for understanding human personality [3]. This model identifies five core traits that serve as the building blocks of personality. These traits are Openness to Experience (Openness), Conscientiousness, Extraversion, Agreeableness, and Neuroticism, commonly remembered by the acronym OCEAN.

- Openness to Experience (Openness): Openness to Experience describes the extent to which an individual is imaginative, curious, and open-minded. People who score high in openness are typically more creative, willing to engage in novel experiences, and open to new ideas. They tend to have a broad range of interests and are more likely to seek out new and different experiences.
- Conscientiousness: Conscientiousness reflects an individual's degree of organization, dependability, and discipline. Highly conscientious people are typically reliable, well-organized, and diligent. They are known for their strong sense of duty and their ability to plan and follow through on tasks.
- Extraversion: Extraversion is characterized by an individual's sociability, energy, and assertiveness. Extroverted individuals are often outgoing, talkative, and enjoy being in social settings. They are energized by interactions with others and are often perceived as enthusiastic and active.
- Agreeableness: Agreeableness measures the quality of an individual's interpersonal interactions, including their cooperativeness, kindness, and compassion. People who are high in agreeableness are typically warm, friendly, and considerate. They are more likely to be empathetic, trusting, and cooperative in their interactions with others.
- Neuroticism: Neuroticism refers to the tendency to experience negative emotions, such as anxiety, depression, and anger. Individuals high in neuroticism are more likely to experience mood swings, stress, and emotional instability. They may react more strongly to stressors and be more susceptible to feelings of worry and sadness.

Many of the studies related to the Big Five Personality model are about learning styles and academic achievements [17]. One study [12] examines the relationship between personality traits, learning styles, and academic achievement among 308 undergraduate college students. Using the Five Factor Inventory to assess personality and the Inventory of Learning Processes to determine learning styles, the researchers found significant correlations between certain personality traits and learning styles, as well as their impact on grade point average (GPA). Also some studies aim at using state-of-art machine learning methods to do the personality traits prediction, particularly those utilizing pre-trained language models. A novel approach [13]: correlation analysis, helps in understanding how closely the predictions made by the model align with actual personality traits, while ablation studies involve systematically removing certain parts of the model to see how performance changes. These experiments showed that when SenticNet6 is integrated with language models, the model's

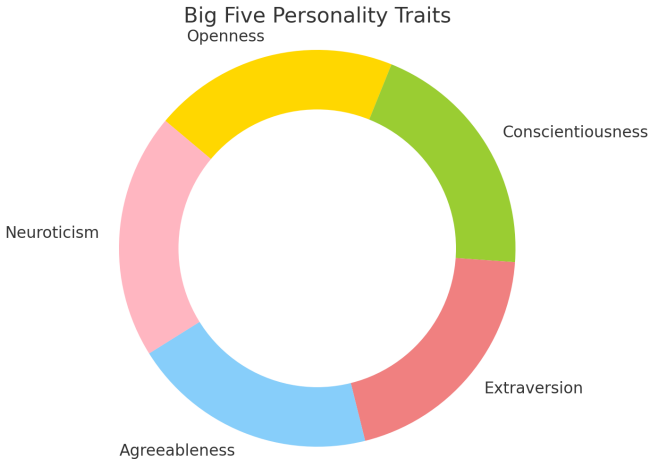

Figure 1: An infographic illustrating the Big Five Personality Traits

ability to predict personality traits improves significantly. This suggests that incorporating specialized sentiment knowledge can enhance the model's understanding of the emotional and psychological nuances present in text, leading to more accurate predictions of personality traits.

2.2 Related Work on Personality Trait Prediction

The myPersonality dataset is one of the most comprehensive collections of psychometric and social media profile data available for academic research. Originally derived from the myPersonality Facebook application, which operated between 2007 and 2012, this dataset includes data voluntarily contributed by millions of Facebook users. The participants completed various psychometric tests, including the Big Five personality test.

The myPersonality dataset has played a key role in advancing research across various disciplines, particularly in psychology, computational linguistics, and machine learning. The application of Convolutional Neural Networks (CNNs) and Long Short-Term Memory networks (LSTMs) in text analysis has provided new avenues for personality prediction. Several studies have employed these methods using the myPersonality dataset to extract and analyze linguistic features correlated with personality traits. Particularly, a study [24] advances the field of personality prediction by applying deep learning techniques to Facebook user data, using the Big Five Personality Model. Previous research on personality prediction often relied on methods such as LIWC (Linguistic Inquiry and Word Count) [25] and SPLICE (Sparse LInear Compositional Embedding) for linguistic analysis. However, LIWC and SPLICE have limitations in capturing deep contextual relationships within text. To address these limitations, this work moves beyond these traditional methods by employing a combination of Convolutional Neural Networks (CNN) [1] and Long Short-Term Memory networks (LSTM) [19]. This integrated model leverages CNNs to capture spatial patterns in text, such as local word groupings that might signify specific personality traits, while LSTMs focus on the sequential nature of language, maintaining the context over longer textual spans. Compared to other machine learning approaches, such as Naive Bayes, Support Vector Machines

(SVM), Logistic Regression, Gradient Boosting, Latent Dirichlet Allocation (LDA), and simpler models like CNN or LSTM alone, their combined CNN-LSTM model demonstrates superior performance. This approach achieves a significant improvement in prediction accuracy, with an average accuracy rate of 74.17%. This marks a substantial advancement over traditional methods and sets a new benchmark for future research in personality assessment through text analytics.

In addition, recognizing the importance of personality in communication, a recent study [18] presents a novel method for administering and validating personality tests using widely-used large language models (LLMs). By employing a structured prompting approach, the study subjected various LLMs to repeated personality assessments and psychometric tests. This approach is relevant to our work as it underscores the importance of understanding and modeling personality in text generation tasks, further informing our methods for personality trait prediction using BERT models. Another article [10] uses BERT to extract contextualized word embeddings from text for the purpose of automated personality detection. To achieve this, the study develops a model that combines BERT's contextualized embeddings with psycho-linguistic features, which are then fed into a Bagged-SVM classifier for personality trait prediction.

2.3 BERT Model

While the combination of CNN and LSTM has significantly improved the accuracy of personality predictions from social media data, the field continues to evolve with the introduction of more advanced models like BERT [6] (Bidirectional Encoder Representations from Transformers). Unlike CNN, which primarily captures spatial hierarchies, and LSTM, which processes data sequentially to capture temporal dependencies, BERT analyzes text in a more holistic manner.Traditional models like LSTM typically process text in one direction—either from the beginning to the end of a sentence (unidirectional) or from the end to the beginning. While LSTM processes text sequentially, meaning one token at a time and in order, BERT processes text in parallel. This means that when BERT encounters a word, it takes into account not only the words that come before it but also the words that come after it. This allows BERT to understand the full context of a word by looking at the words that come before and after it, making it particularly effective for tasks that require a deep understanding of language nuances. In the next section we will introduce BERT in detail.

Bidirectional Encoder Representations from Transformers (BERT) [6] is a groundbreaking model in the field of natural language processing (NLP), introduced by researchers at Google in 2018. BERT is built upon the Transformer architecture but is distinctively an encoder-only transformer model. This design allows BERT to generate embedding for input text, capturing rich contextual information. BERT's novel approach lies in its use of bidirectional training of Transformer, a type of attention mechanism that learns contextual relations between words (or sub-words) in a text. Unlike previous models that process words sequentially, either from left-to-right or combined left-to-right and right-to-left, BERT reads the entire sequence of words at once. This characteristic allows the model to capture a richer understanding of context and word relationships, making it highly effective for tasks that require a deep understanding of language context.

BERT is pre-trained on a large corpus of text using two self-supervised tasks: masked language modeling (MLM) and next sentence prediction (NSP). In MLM, BERT learns to predict randomly masked words in a sentence, which gives it a deep understanding of language structure. In NSP, the model learns to predict whether a sentence logically follows another, enhancing its ability to understand relationships between sentences. This pre-training prepares BERT to tackle a wide array of tasks with only a small amount of task-specific data, revolutionizing the efficiency and effectiveness of fine-tuning models for specific NLP tasks.

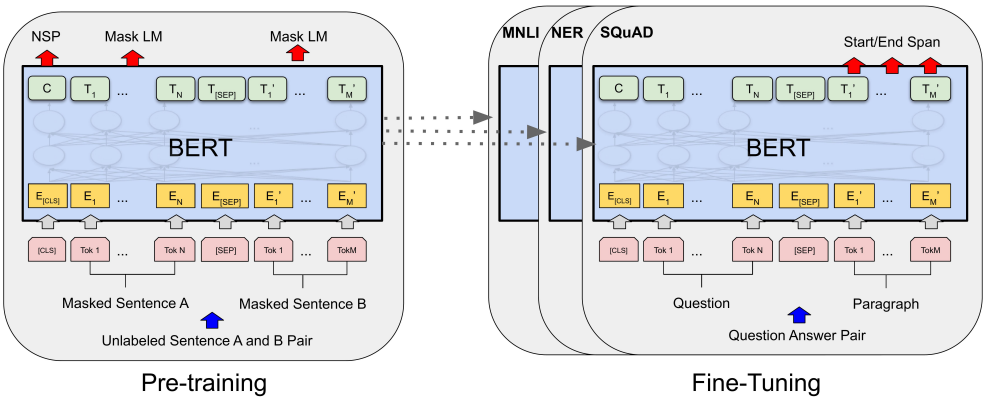

Figure 2: An overview of the BERT model architecture from [6]. The left part shows the pre-training phase with tasks such as Next Sentence Prediction (NSP) and Masked Language Model (Mask LM).

For our research, BERT's capabilities make it an ideal candidate for analyzing text data to extract personality traits. Its deep contextual learning allows it to interpret the nuances and complexities of language used in CVs and LinkedIn profiles, which is critical for accurate personality assessment. By fine-tuning BERT on the myPersonality dataset, we use its pre-trained contextual embeddings to predict the Big Five personality traits, thereby enhancing the predictive power and reliability of our results. The application of BERT in personality prediction can address some of the limitations faced by CNN-LSTM architectures, such as handling complex sentence structures and idiomatic expressions more effectively. Moreover, BERT's pre-training on a vast corpus enables it to start with a rich understanding of language, which can be fine-tuned to specific tasks like personality prediction with relatively less data than what is traditionally required for training deep learning models from scratch.

BERT is pre-trained on large corpora using two self-supervised tasks: Masked Language Modeling (MLM) and Next Sentence Prediction (NSP). After pre-training, BERT can be fine-tuned on a specific downstream task, such as text classification, sentiment analysis, or, in our case, predicting personality traits from text. Fine-tuning a BERT model is supposed to be the process of adapting a pre-trained BERT model to a specific downstream task. This process allows the model to learn from task-specific data on top of its general language understanding gained during pre-training, enhancing its performance on the given task.

The process of input preparation converts the task-specific data into a format that BERT can process. This typically involves tokenizing the text into token IDs and adding special

tokens like [CLS] and [SEP]. For sentence classification tasks, the input format is [CLS] sentence [SEP]. This usually involves adding a fully connected layer on the output of the [CLS] token. And for a regression task, this should be a linear layer and a multiplier to make sure the result is in the right range. Then use an optimizer (such as Adam [11]) with a suitable learning rate to update the model parameters. A smaller learning rate is usually adopted during fine-tuning to prevent drastic changes in pre-trained weights. Training the model should use the task-specific dataset (myPersonality dataset), gradually adjusting the model parameters. The training process typically involves multiple epochs, where the entire training data is traversed once per epoch.

2.4 Large Language Models for Data Augmentation

In recent years, the generation of synthetic data has become an effective method to address the challenges of limited and costly conversational datasets. Traditionally, dialogue data was collected through crowd-sourcing, which is resource-intensive and difficult to scale. To overcome these limitations, researchers have developed techniques to create synthetic dialogue data by augmenting existing datasets or converting text into conversational formats. This approach [22] has proven beneficial in enhancing the training of models, particularly in domains with scarce data. Our work draws on these methods, applying similar strategies to augment and improve the training of BERT models for personality trait prediction.

The field of data augmentation in natural language processing (NLP) is still in its early stages, especially when compared to its more mature counterpart in computer vision (CV) [20]. By examining the use of prior knowledge in self-supervised learning alongside conventional data augmentation methods, the article explores diverse strategies for enhancing model performance in natural language processing tasks.

Another relevant study [14] discusses the importance of data pre-processing in machine learning, highlighting the need to address issues such as noise, corruption, and inconsistencies in raw data. It emphasizes that poor data quality can lead to inaccurate predictions, making pre-processing essential for improving dataset quality. The paper reviews various pre-processing techniques, including classification, clustering, and data augmentation methods to enhance model performance without distorting the original data. In connection to our work, this paper inspired the idea of adding Gaussian noise as a data augmentation technique in our experiments. By incorporating this method, we aimed to improve the robustness of our models, similar to how image data is augmented to reduce dependency on training data and enhance machine learning model performance.

The challenges of limited sample sizes in NLP tasks, particularly in few-shot learning scenarios, are further explored in another study [5]. It highlights the importance of data augmentation as a strategy to increase sample size and capture data in-variance. However, existing text data augmentation methods often struggle with maintaining correct labeling (faithfulness) or generating diverse enough samples (compactness). The paper introduces a new approach, AugGPT, which leverages large language models like ChatGPT to rephrase sentences into multiple conceptually similar but semantically diverse versions. This method is shown to improve the effectiveness of text classification tasks by enhancing the diversity

and accuracy of augmented data. This concept directly influenced our work, where we applied a similar strategy to rephrase sentences and generate augmented samples for training, effectively improving model performance.

3 Data

3.1 MyPersonality Dataset

The myPersonality dataset is a widely used resource in psychological and computational research for studying personality traits [23]. The dataset consists of personality scores for over 6 million individuals, making it one of the largest datasets of its kind. However, the whole dataset was recalled by its developers and no longer available. Since then, for practical research purposes, smaller subsets are often used. A subset includes around 10,000 samples. The dataset was divided into training (80% of the total size) and test sets (20% of the total size) randomly, with handling errors. The least words of a row in the myPersonality dataset has 1 word, while the most words of a row has 4325 words. In the data processing phase, we encountered 176 items that required special error handling. These included both blank entries and items that could not be processed due to UTF-8 encoding issues. Additionally, The myPersonality dataset comprises 9742 text samples from 250 users, but it only includes 250 distinct sets of personality scores.

Description	**Size**
Training Dataset Size	7793 rows
Testing Dataset Size	1949 rows
Deleted Data Size	176 rows
Original Total Size	9918 rows
Processed Total Size	9742 rows
Number of Users	250

Table 2: myPersonality Dataset Details: numbers of data in training set, test set, deletion and total

3.2 IMDB Dataset

The IMDB dataset is a widely used resource in sentiment analysis and natural language processing (NLP) research [26]. This dataset consists of movie reviews from the Internet Movie Database (IMDB) and is commonly used for binary sentiment classification tasks, where the goal is to determine whether a given movie review is negative (score 0-4) or positive (score 7-10). The dataset includes 50,000 movie reviews, with an equal number of positive and negative reviews, making it well-balanced for binary classification tasks.

4 Methods

In this section, we first describe our approach to extracting and preprocessing data from the myPersonality and IMDB datasets. We then explain how to tokenize this data and use it to fine-tune BERT models. Following this, we detail the process of replicating the BERT classification task on the IMDB dataset and further fine-tuning the previously mentioned 3 BERT models using IMDB dataset. Finally, we discuss how we performed data augmentation

Description	Size
Positive Dataset Size	25000 rows
Negative Dataset Size	25000 rows
Training Dataset Size	25000 rows
Testing Dataset Size	25000 rows
Total Size	50000 rows

Table 3: IMDB Dataset Details: numbers of data in training set, test set, and total. Positive+Negative=Total, Training+Testing=Total.

on the myPersonality dataset and used the augmented data to further fine-tune the three BERT models.

4.1 Data Preprocessing

4.1.1 Data Extraction

To prepare the myPersonality dataset for further analysis, we extracted and separated the text data from the associated personality scores. We loaded the dataset with assuming the first row contained column headers and used a regular expression to match the user ID, text within double quotes, and the five personality scores. The pattern was used to effectively capture these elements. We extracted the matched fields into a new DataFrame with columns for *userid*, *text*, *e_score*, *n_score*, *a_score*, *c_score*, and *o_score*, and converted the score columns to numeric types for further analysis. After inspecting the data to ensure correctness, we saved the cleaned and structured data to a new CSV file, excluding any rows with missing values. This process ensured that the text and personality scores were properly separated and formatted for subsequent analysis and model training.

Field	Description
userid	Unique identifier for each user
text	Text data from the user's responses
e_score	Score for extraversion
n_score	Score for neuroticism
a_score	Score for agreeableness
c_score	Score for conscientiousness
o_score	Score for openness

Table 4: Fields in the Extracted Data

For the IMDB dataset, we aim to perform both classification and regression tasks. The original binary classification labels (negative and positive) are insufficient for our regression analysis. Therefore, we restructure the dataset to include a broader range of sentiment scores. The steps taken are as follows: We extract the text from 25,000 negative reviews, and each review is assigned an original sentiment score ranging from 1 to 4. Also we extract

the text from 25,000 positive reviews, and each review is assigned an original sentiment score ranging from 7 to 10. We combine the negative and positive reviews into a single dataset. This resulted in a dataset with 50,000 reviews, each paired with a sentiment score ranging from 1 to 4 for negative reviews and 7 to 10 for positive reviews. To ensure a balanced and fair evaluation, we randomly split the combined dataset into training and testing sets with a $50\% - 50\%$ split, with 25000 reviews and scores in both training set and test set.

4.1.2 Data Splitting

To evaluate the impact of having the same user in both the training and test sets on model performance, we propose an additional method involving the division of the myPersonality dataset based on user IDs. This will allow us to observe whether the presence of the same user in both sets affects the training results. Start with the original myPersonality dataset consisting of 9742 samples, split the dataset into training and test sets based on user IDs. Randomly select 80% users and choose the training dataset according to their user ID. The last 20% users' text and scores would become the test dataset. This ensures there is no overlap of user IDs between the training and test datasets.

4.2 Data-Model-Relationship Framework

In this study, we evaluate the interplay between the dataset, model, and the relationships within the data. We design our method based on this, and try to investigate each relation. Our approach is threefold, as shown in Figure 3:

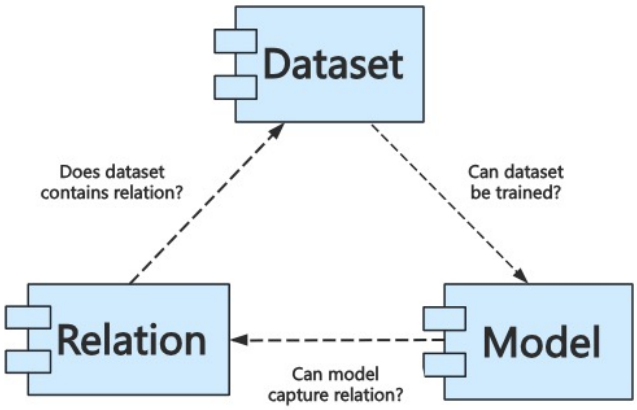

Figure 3: Triangle of dataset, model and relation.

4.2.1 Dataset and Model

The objective of this part is to assess whether our model can be effectively trained on the dataset. We train three different BERT models (bert-base-uncased, bert-base-uncased with

fix weight and one linear layer, and bert-base-uncased with fix weight and two linear layers) on the preprocessed myPersonality and IMDB datasets.

4.2.2 Model and Relation

The goal of this section is to assess the model's ability to recognize and identify patterns or relationships within the data. For the model to effectively learn these patterns, the dataset must be sufficiently large to provide diverse examples. Small datasets, such as the myPersonality dataset with only 250 users and valid scores, are prone to overfitting, making it challenging for the model to generalize well. To address this limitation, we turn to a well-established and robust dataset—the IMDB dataset—which is widely recognized for its ability to demonstrate the effectiveness of models like BERT. By training the BERT model on the IMDB dataset, we aim to confirm the model's capability to capture the necessary relationships, thereby validating its potential to perform similarly on other datasets.

4.2.3 Dataset and Relation

The objective of this part is to confirm whether the dataset contains the relevant relationships or patterns necessary for the model to learn effectively. This involves conducting an initial examination of the dataset to understand its structure and the distribution of personality scores. We use feature correlation analysis to identify inherent relationships between predicted scores and real scores. The BERT model is then trained and tested on the dataset to evaluate its performance using metrics such as Mean Squared Error (MSE), R-squared (R^2), and correlation matrices, which help assess the model's learning capability and the adequacy of the dataset. Additionally, techniques for data augmentation and preprocessing, such as adding noise, gluing data by user ID, and splitting data by user ID, are explored to enhance the dataset's quality and observe their impact on the model's performance. By thoroughly analyzing the dataset and validating the model's learning patterns, this part aims to establish whether the dataset provides the necessary information for the model to make accurate and meaningful predictions.

By examining these three aspects—Dataset and Model, Model and Relation, and Dataset and Relation—we investigate if the dataset is suitable for training the model, and the model is capable of learning from the dataset, and the dataset contains the necessary relationships for the model to identify.

4.3 BERT Models

The original BERT model is mainly used for natural language processing tasks such as language modeling and text classification. In our work, we adapt the BERT model for both classification and regression tasks by modifying its output layer to suit our specific needs. The primary outputs of the BERT model are sequence output and pooled output. We use this pooled [CLS] output as the input to our classification or regression layer, where the learned representation of all tokens is pooled into the [CLS] token, and this pooled representation is then fed into the final linear layer for prediction.

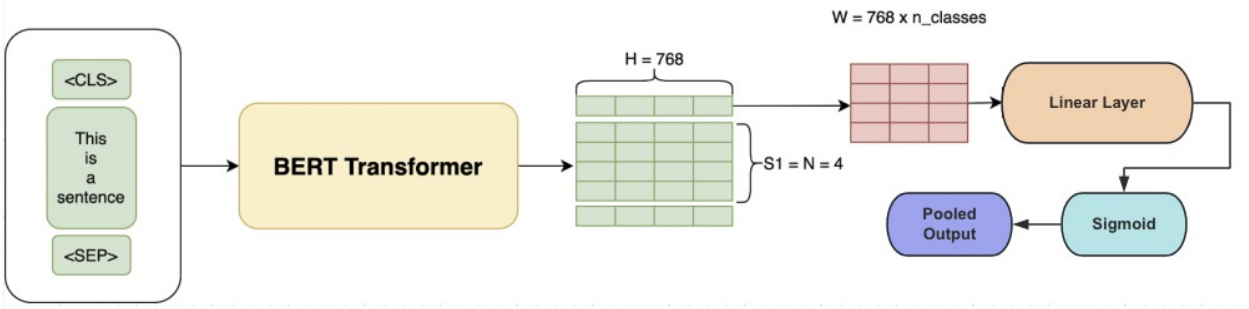

Figure 4: Adapting BERT for Classification and Regression Task with a Linear Layer.

An additional linear layer is added on top of the pooled output to produce the regression and classification logits. In our work, we do the binary classification task on IMDB dataset because it has the label of a binary negative and positive. We use a linear layer to map BERT's output dimension $W = 768$ to the num_labels dimension 2. We also do the regression task on myPersonality dataset because it has five different personality traits score. We also use a linear layer to map BERT's output dimension $W = 768$ to the num_labels dimension 5. Additionally, in order to make sure the output is in the range between 0 and 5, we first use sigmoid activation function to limit the output range between 0 and 1, then multiply the output by 5. We want to further investigate the impact of the complexity of the BERT model. So we try to freeze the pre-trained BERT model weights, and just let the linear layer train. In general a more complicated structure of a neural network should learn faster. We also add two additional linear layers on top of the BERT model's output to further process the pooled [CLS] representation. The first additional linear layer is followed by a Leaky ReLU activation function, which introduces non-linearity and helps in handling the issue of dying ReLUs by allowing a small, non-zero gradient when the unit is not active. The second and final linear layer is followed by a sigmoid activation function. This sigmoid function scales the output to the range of 0 to 5, which is suitable for our specific regression task using BERT, ensuring that the model's predictions fall within the desired score range. Then we can compare the performance of three different BERT configurations on the myPersonality regression task and IMDB sentiment classification task. We would discuss the configuration in Section 5.1.

4.4 Fine-tuning BERT Models

In this section, we describe the process of setting up and customizing three BERT model for regression tasks on three myPersonality dataset and both classification and regression tasks on the IMDB dataset. The following steps outline the procedures and code used to achieve this.

We initialize a pre-trained BERT model for sequence classification and created a custom BERT model to handle continuous outputs for regression tasks. The custom model includes a regression layer and a sigmoid activation function to predict continuous scores in the range of 0 to 5, corresponding to the Big Five personality traits.

Also, we initialize the BERT tokenizer using the bert-base-uncased pre-trained model. The texts are tokenized to convert them into a format that BERT can process. This involves truncation, padding, and setting a maximum sequence length of 512 tokens.

The next procedure involves initializing the model, training it over multiple epochs, and evaluating its performance on a test set at the end of each epoch. Before starting the training, we calculate the initial test loss using the model's predictions on the test set. This helps in establishing a baseline for comparison. In each epoch, the model processes batches of data, calculates the loss, performs back-propagation, and updates the model weights. After each epoch, the model's performance is evaluated on the test set. The predictions and actual scores are compared using Mean Squared Error (MSE) [15] to calculate the test loss for the epoch. We also calculate R^2 and the correlation matrix [7] of predicted scores and real scores. By tracking the training and test losses, we can monitor the model's learning progress and ensure it is effectively capturing the relationships in the data. Also we calculate R^2 and correlation matrix to see the ability of generalization of the model.

4.5 Classification task on BERT Model Using IMDB Dataset

The objective of this method is to demonstrate that a text-score formatted dataset can be effectively trained using a BERT model, and the patterns within the data can be recognized, at least in the context of a classification task. This involves using the IMDB dataset to train a BERT model for binary sentiment classification and evaluating its performance. The model uses the pre-trained BERT model with an additional linear layer for classification, which is similar to previous fine-tuning BERT model. Then do the fine-tuning using IMDB dataset. The difference is here we measure the accuracy of the model on test set. We calculate initial test loss and test accuracy to measure the training process. After demonstrating success in binary classification, the focus shifts to the primary objective: proving that BERT can learn to predict continuous scores (regression task).

One notable implementation of sentiment analysis is the work[2] on the IMDB dataset, which involves building a model to accurately predict whether a movie review is positive or negative. This project leveraged the pre-trained BERT model provided by Hugging Face.

The dataset used for this project consisted of approximately 50,000 movie reviews from IMDB. After removing duplicates, the final dataset contained 49,582 samples with a balanced number of positive and negative reviews. The BERT model (bert-base-uncased) was fine-tuned with the following hyperparameters:

- Learning Rate: $2e-5$, optimized using the AdamW optimizer.
- Scheduler: Linear scheduler with $num\_warmup\_steps = 0$.
- Maximum Sequence Length: 128 tokens.
- Batch Size: 32.
- Number of Training Epochs: 5.

[2] https://github.com/MLphile/BERT_on_Movie_Reviews?tab=readme-ov-file

This fine-tuning process involved adjusting the pre-trained BERT model to the specific task of sentiment classification on movie reviews. The model was then uploaded to the Hugging Face hub for easy access and deployment.Evaluation of the model showed an accuracy of 89.35% on the validation set.

Also, to determine the minimal dataset size required to maintain a good performance without overfitting on the IMDB dataset, the training set size was gradually reduced from 25,000 to 10,000, and then to 5,000 while keeping the test set constant at 25,000. The learning curves and evaluation metrics for each configuration are analyzed to understand the model's behavior.

4.6 Data Augmentation

4.6.1 Data Aggregation

To address the mismatch between the number of text samples $x = 9742$ and the number of personality scores $y = 250$ in the myPersonality dataset as we show in the data section, we implement two data augmentation strategies. This ensures that our model can learn more effectively from the available data by creating balanced datasets where the number of text samples matches the number of personality scores.

The myPersonality dataset contains 250 users, resulting in 9742 text samples but only 250 unique sets of personality scores. This discrepancy arises because multiple text samples can belong to the same user, all sharing the same personality scores. To mitigate this, we applied a data augmentation method where we concatenated (glued) all text samples from the same user into a single text entry. We collect all text samples for each of the 250 users. We concatenate these samples to form a single, comprehensive text entry for each user. As a result, we created a new dataset where the number of text samples x matched the number of personality scores y, resulting in 250 text entries corresponding to 250 sets of personality scores.

However, since BERT can only process inputs up to 512 tokens, we faced the challenge of text entries exceeding this limit after concatenation. To mitigate this, we employed a truncation strategy, where the concatenated text was truncated to fit within the 512-token limit.

4.6.2 Adding Noise

A different technique we use to augment the data is adding Gaussian noise to the personality scores, creating a larger dataset where each text sample has a unique, slightly varied set of scores. This approach allows us to balance the dataset by increasing the number of unique personality scores to match the number of text samples:

$$y' = y + \mathcal{N}(0, \sigma^2) \tag{1}$$

where:

- y' is the new personality score after adding noise.
- y is the original personality score.

- $\mathcal{N}(0, \sigma^2)$ represents Gaussian noise with a mean of 0 and variance σ^2.
- σ is the standard deviation of the Gaussian noise.

For each text sample, we add Gaussian noise with $\sigma = 0.05$ to the original personality scores. This noise is generated with a mean of zero and a small standard deviation, ensuring that the augmented scores remained close to the original values but were distinct enough to be considered new samples. This augmentation produced 9742 unique sets of personality scores, each slightly different from the others, corresponding to the 9742 text samples. Through these augmentation techniques, we create three datasets:

- Original myPersonality Dataset: The initial dataset with 9742 text samples and 250 unique sets of personality scores.
- Glued myPersonality Dataset: A new dataset where all text samples from each user are concatenated, resulting in 250 text samples and 250 unique sets of personality scores.
- Noise-Added myPersonality Dataset: An augmented dataset where Gaussian noise was added to the personality scores, resulting in 9742 text samples and 9742 unique sets of personality scores.

These datasets allow us to explore the effectiveness of different data augmentation strategies in improving the model's ability to learn from and generalize to the myPersonality dataset.

4.6.3 Data Augmentation Using LLM

There is a third method for augmenting the myPersonality dataset. We implement a method using OpenAI's ChatGPT-4 model to generate additional texts and adjusted personality scores.

Initially, we chose to augment the aggregated data from the myPersonality dataset. This decision is motivated by the observation that the original dataset often contained entries with very limited textual information, sometimes just one or two words, which are insufficient for accurately predicting an individual's Big Five personality scores. Aggregated data, where multiple entries for a single user are combined, provides a richer context and more comprehensive information about the user's personality traits. This approach enhances the dataset's quality and ensures that the generated texts have enough substance to be useful for training the model. We load the original dataset and remove the user ID column to focus on the texts and scores. For each user within a specified range, we create a prompt designed to instruct ChatGPT-4 to generate multiple similar texts, each accompanied by personality scores resembling the original ones.

The prompt (See details in Appendix A) instructs ChatGPT-4 to produce 50 similar texts for each original text, including personality scores formatted as $E\_score, N\_score, A\_score, C\_score$, and O_score. We request these texts and scores using ChatGPT-4's API, ensuring the response contain enough tokens for the generated content. Upon receiving the response, we parse the generated texts and scores, splitting them appropriately. Given that instructions in the prompt, the ChatGPT-4 model has a robust understanding of these dimensions based on its training data.

To ensure variability while maintaining realism, we adjust the personality scores by adding random noise within the range of -0.1 to 0.1. These adjusted scores are then clip to ensure they remain within the valid range of 0 to 5. This process includes error handling to manage potential parsing issues and rate limits, with appropriate delays and retries.

Because experiments with the IMDB dataset established that a suitable dataset size for training was 10K, we aim to achieve this target size for the new augmented dataset. The newly generated datasets are then used to fine-tune the BERT model. The test set is selected from the original aggregated data, consisting of 50 users' data that were not used in the augmentation either, to evaluate the model's performance. Additionally, we will do the experiment with using the entire aggregated data as the test set, which includes all the texts and scores from the original myPersonality dataset.

4.7 Evaluation Methods

4.7.1 Mean Squared Error

The Mean Squared Error (MSE) we used is a common loss function used for regression tasks. It measures the average of the squares of the errors, which are the differences between the predicted values $\hat{m}$ and the actual values m:

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(m_i - \hat{m}_i)^2 \tag{2}$$

where:

- n is the number of observations.
- m_i represents the actual value for the i-th observation.
- $\hat{m}_i$ represents the predicted value for the i-th observation.

4.7.2 R-squared

We utilize R-squared [4] to represent the proportion of the variance for a dependent variable that's explained by an independent variable or variables in a regression model. It provides an indication of the goodness of fit of the model. The R-squared (R^2) is defined as:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{3}$$

where:

- y_i is the actual value for the i-th observation.
- $\hat{y}_i$ is the predicted value for the i-th observation.
- $\bar{y}$ is the mean of the actual values.
- n is the number of observations.

The R^2 value of 1 indicates perfect prediction, whereas an R^2 value of 0 indicates that the model does not explain any of the variability of the response data around its mean. Additionally, if the model is too simple and fails to capture the underlying trend in the data, it can result in poor predictions and a negative R^2.

4.7.3 Correlation-Matrix

The correlation matrix we used is a table showing correlation coefficients between variables. Each cell in the table shows the correlation between two variables. This matrix is useful to understand the linear relationship between predicted and actual OCEAN scores.

Arrange these coefficients into a matrix form. For variables A, B, C, and D, the correlation matrix $\mathbf{R}$ is:

$$\mathbf{R} = \begin{pmatrix} 1 & r_{AB} & r_{AC} & r_{AD} \\ r_{BA} & 1 & r_{BC} & r_{BD} \\ r_{CA} & r_{CB} & 1 & r_{CD} \\ r_{DA} & r_{DB} & r_{DC} & 1 \end{pmatrix}$$

.

5 Experiments and Results

In this section, we show our implementation details and experiment results.

5.1 Experimental setup

5.1.1 Fine-tuning BERT Models

To comprehensively evaluate the performance of different BERT model configurations on various versions of the myPersonality dataset, we design a 3x3 experimental matrix. This matrix allows us to systematically explore the interactions between different model architectures and dataset augmentation strategies. The detailed description of the approach. is **BERT** Base Uncased and **One Linear Layer** (bert-base-uncased), referred to as **BERT**. Also we add one or two linear layers to see if the model could be overfitting.

myPersonality datasets:

- Original myPersonality Dataset (referred to as Origin): The initial dataset with 9742 text samples and 250 unique sets of personality scores.
- Glued myPersonality Dataset (referred to as Glued): A dataset where all text samples from the same user are concatenated, resulting in 250 text samples and 250 unique sets of personality scores.
- Noise-Added myPersonality Dataset (referred to as Noise): An augmented dataset where Gaussian noise with a standard deviation of 0.05 is added to the personality scores, resulting in 9742 text samples and 9742 unique sets of personality scores.
- Augmented datasets by ChatGPT4 (referred to as Augmented5k and Augmented10k): Two augmented dataset by ChatGPT4 with the size 5k and 10k.

As we mentioned that we split the original myPersonality dataset as well as the noise myPersonality dataset into the train dataset and test dataset according to the user ID (referred to as User-Split). We put the new split data into BERT model to run the fine-tuning experiments. To analyze the results of using a single model to predict five personality trait scores versus using five separate models to predict each trait, we do an additional experiment to set five models to predict five scores.

5.1.2 Training Details

Then we put IMDB dataset also in BERT fine-tuning, so there will become a experiment matrix, with 2 user-split experiments and 2 augmented experiments as illustrated in Table 5:

Dataset / Model	BERT	BERT1L	BERT2L
Origin Dataset	BERT_Ori	BERT1L_Ori	BERT2L_Ori
Glued Dataset	BERT_Glued	BERT1L_Glued	BERT2L_Glued
Noise Dataset	BERT_Noise	BERT1L_Noise	BERT2L_Noise
IMDB Dataset	IMDB	IMDB_1L	IMDB_2L
User-Split Origin Dataset	BERT_Ori_User		
User-Split Noise Dataset	BERT_Noise_User		
Augmented5K	BERT_Aug5K		
Augmented10K	BERT_Aug10K		

Table 5: Experimental Matrix: Combinations of BERT Models and myPersonality Datasets and IMDB dataset.

All the training experiments are conducted on Google Colab, using an NVIDIA Tesla T4 GPU with a 16GB GDDR6 memory.

BERT_Ori, BERT1L_Ori, BERT2L_Ori, BERT_Noise, BERT1L_Noise, BERT2L_Noise are trained with a batch size of 32, 32 epochs and learning rate $1e{-}5$. BERT_Glued, BERT1L_Glued, BERT2L_Glued are trained with a batch size of 1, 48 epochs and learning rate $1e-5$.

For replicating IMDB classification task we use a batch size of 16, 5 epochs and learning rate $2e-5$. For IMDB regression task with BERT model(IMDB), we use a batch size of 32, 8 epochs and learning rate $1e-5$ in general, and for the optimum reduce epochs to 4. For IMDB_1L and IMDB_2L, we use a batch size of 32, 8 epochs and learning rate $1e-5$.

5.2 Evaluation Measures

In the experiments we calculate the correlation matrix of untrained BERT model with the original myPersonality dataset as the baseline. Also the initial test error is the baseline for every training to determine whether it learns or not. In general, the lower MSE indicates a better performance. In our case, R^2 falls within the range of $(0, 1)$. The closer the R^2 is to 1, the better performance we get. For correlation matrix, the correlation coefficient ranges $(-1, 1)$: higher positive correlation coefficients between the predicted scores and the real scores are desirable. This indicates that the model's predictions are closely aligned with the actual values. Specifically, correlation coefficients close to 1 indicate that the model has high predictive accuracy. Correlation coefficients closer to 0 or negative values suggest poor predictive performance.

5.3 Results of myPersonality Dataset on Three BERT Models

5.3.1 Results of myPersonality Dataset on the BERT Model

The results of myPersonality dataset on the BERT model are shown in Figure 5 and 6.

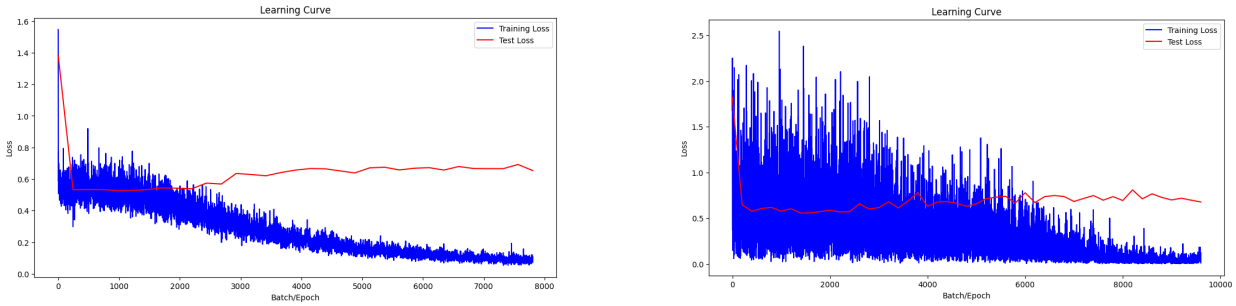

Figure 5: Comparison of Learning Curves for Different Datasets: Learning Curve for Original myPersonality Dataset(left), Learning Curve for Glued myPersonality Dataset(right), Both Using BERT Model.

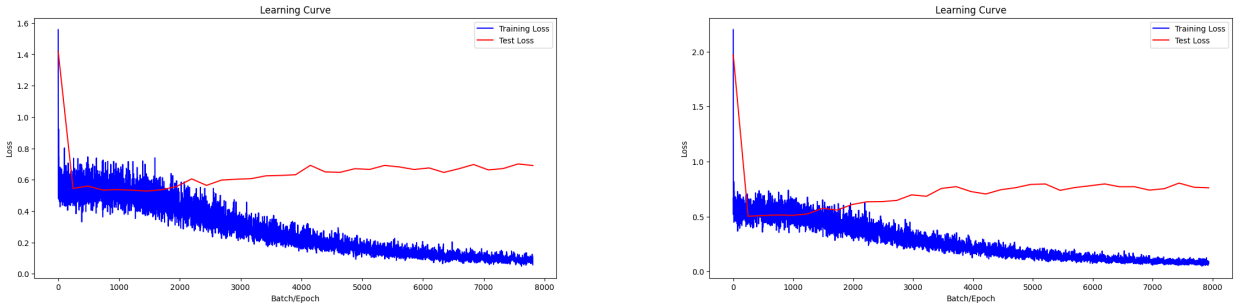

Figure 6: Comparison of Learning Curves for Different Datasets: Learning Curve for Noise myPersonality Dataset(left), Learning Curve for User-split myPersonality Dataset(right), Both Using BERT Model.

Model	**Mean Squared Error**	**R^2**	**Initial Test Loss**
BERT/Original	**0.65**	**-0.22**	1.38
BERT/Glued	0.68	-0.26	1.82
BERT/Original noise	0.69	-0.29	1.42
BERT/Original Userid split	0.76	-0.57	1.97

Table 6: Results of BERT fine-tuning on myPersonality dataset with different settings

For the Original myPersonality dataset with the BERT model, The training loss curve demonstrates a steady decline, but the gap between training and test loss becomes noticeable towards the end, hinting at mild overfitting. The results indicate that the subset that only contains 250 scores might be too small for the complexity of the BERT model, as seen from the relatively high initial test loss and the eventual plateauing of the test loss. For the glued myPersonality dataset with the BERT model, the training loss shows a gradual decrease, similar to the original setting, but with higher fluctuations. A larger gap between

training and test loss is observed compared to the original setting, indicating increased overfitting. Combining texts into a glued dataset may have introduced redundancy and noise, exacerbating overfitting and reducing the model's effectiveness in learning. This indicates that the small dataset size further hindered the model's ability to discern meaningful patterns. For the noise myPersonality dataset, the training loss declines steadily, but the test loss remains relatively high, showing poor convergence on the test set. The gap between training and test loss is more pronounced than in the original setting, indicating significant overfitting. Adding noise to the original dataset introduced more variance, resulting in worse performance and overfitting, further indicating that the small dataset size is insufficient for the model to learn robustly. For the userid split myPeronality dataset with the BERT model, the training loss decreases initially but shows a more significant gap with the test loss, which remains relatively high. We will discuss this in details later.

The competitive results among these four dataset settings also shown in Table 6. On the other hand, the results are indicating that the model performs better with the original dataset, but still shows signs of overfitting.

These analyses illustrate the importance of dataset handling and preprocessing methods in fine-tuning models for specific tasks. The results also indicate that the myPersonality dataset is likely too small to effectively train a complex model like BERT, leading to poor generalization and overfitting issues. Effective data augmentation and larger datasets could potentially improve model performance and mitigate these problems.

5.3.2 Results of myPersonality Dataset on the BERT Model with Linear Layers

The results of myPersonality dataset on the BERT model with different linear layers are shown in Appendix (Figure 14, Figure 15 and Figure 16).

Model/Configuration	**Initial Test Loss**	**Mean Squared Error**	**R^2**
BERT+1linear/Original	1.85	0.55	-0.04
BERT+2linear/Original	1.45	0.54	-0.02
BERT+1linear/Glued	1.81	0.55	-0.03
BERT+2linear/Glued	1.65	0.56	-0.04
BERT+1linear/Original noise	1.81	0.56	-0.04
BERT+2linear/Original noise	1.49	0.54	-0.02

Table 7: Performance Metrics for Different BERT Configurations

The table 7 presents the performance metrics for various BERT model configurations on the test set.

In summary, while different configurations of the BERT model show variations in initial test loss, the overall performance metrics (MSE and R^2) indicate that all models struggle to fit the test data well. Adding additional linear layers and noise shows slight improvements in some cases, but the overall impact on performance remains minimal.

5.3.3 Results of myPersonality Dataset Split Training Dataset by User ID

The results of comparison of leaning curves for different dataset splitting methods are shown in Figure 7 and Figure 8. The goal is to understand the impact of having the same user in both the training and test sets on the model's performance.

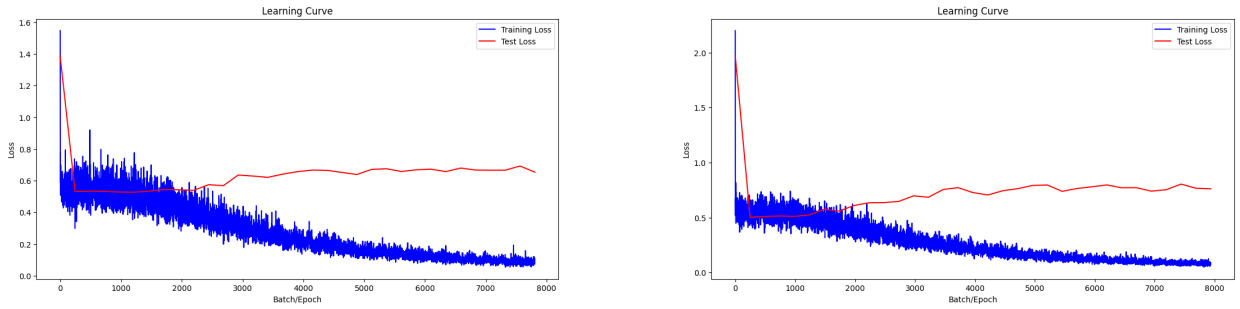

Figure 7: Comparison of Learning Curves for Different Splitting Methods: Learning Curve for Random Splits(left), Learning Curve for User-Based Splits(right), Both Using BERT Model.

Since we do two user ID split datasets: Original myPersonality dataset and Noise myPersonality dataset, we compare two with the random split datasets.

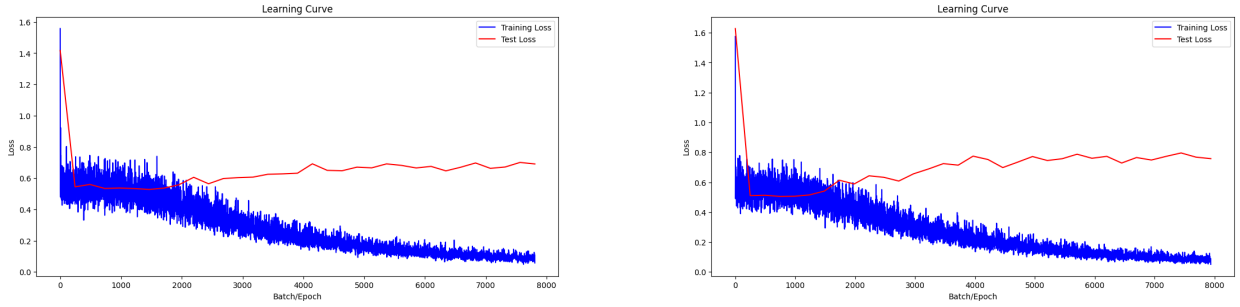

Figure 8: Comparison of Learning Curves for Different Splitting Methods: Learning Curve for Random Splits with Noise myPersonality Dataset(left), Learning Curve for User-Based Splits with Noise myPersonality Dataset(right), Both Using BERT Model.

Split Method	**Initial Test Loss**	**Mean Squared Error**	**R^2 on Test Set**
Random Splits	1.38	0.65	-0.22
User-Based Splits	1.97	0.76	-0.57
Random Splits with Noise	1.42	0.69	-0.29
User-Based Splits with Noise	1.63	0.76	-0.56

Table 8: Performance Metrics for Different Splitting Methods and Data Augmentation

The test loss and MSE indicate that the model struggles to generalize well, as reflected by the negative R^2 value. The correlation matrix shows a moderate correlation between predicted and actual scores, but the model's overall performance is sub-optimal. When the data was split based on user IDs, the model's performance worsened, as indicated by higher test loss and MSE, and a more negative R^2 value. The correlation matrix still shows some correlation between predicted and actual scores, but the model struggles more compared to random splits. Adding noise to the scores did not significantly improve the model's performance. The test loss and MSE are slightly worse than without noise, and the R^2 remains negative. The correlation matrix in Table 16 and Table 18 in Appendix indicate that the model's predictions are not highly correlated with the actual scores.

Though the learning curves show the same trend and overfitting, splitting the dataset by user IDs negatively impacts the model's performance, leading to higher test loss and MSE, and more negative R^2 values.

5.4 Results of IMDB Dataset

5.4.1 Results of IMDB on BERT Model, Classification Task

The test set evaluation results from the original work[3] indicated robust performance, with precision, recall, and F1-scores for both positive and negative reviews as follows:

Class	Precision	Recall	F1-Score	Support
0	0.91	0.89	0.90	3705
1	0.90	0.92	0.91	3733

Table 9: Evaluation Results on the IMDB Test Set

These results demonstrate the effectiveness of fine-tuning BERT for sentiment analysis tasks. By replicating this approach, it is possible to develop a reliable sentiment classification model that can be integrated into various applications, providing valuable insights into user opinions.

The result of training we replicate for classification task of IMDB dataset on BERT is shown in Figure 9.

[3] https://github.com/MLphile/BERT_on_Movie_Reviews?tab=readme-ov-file

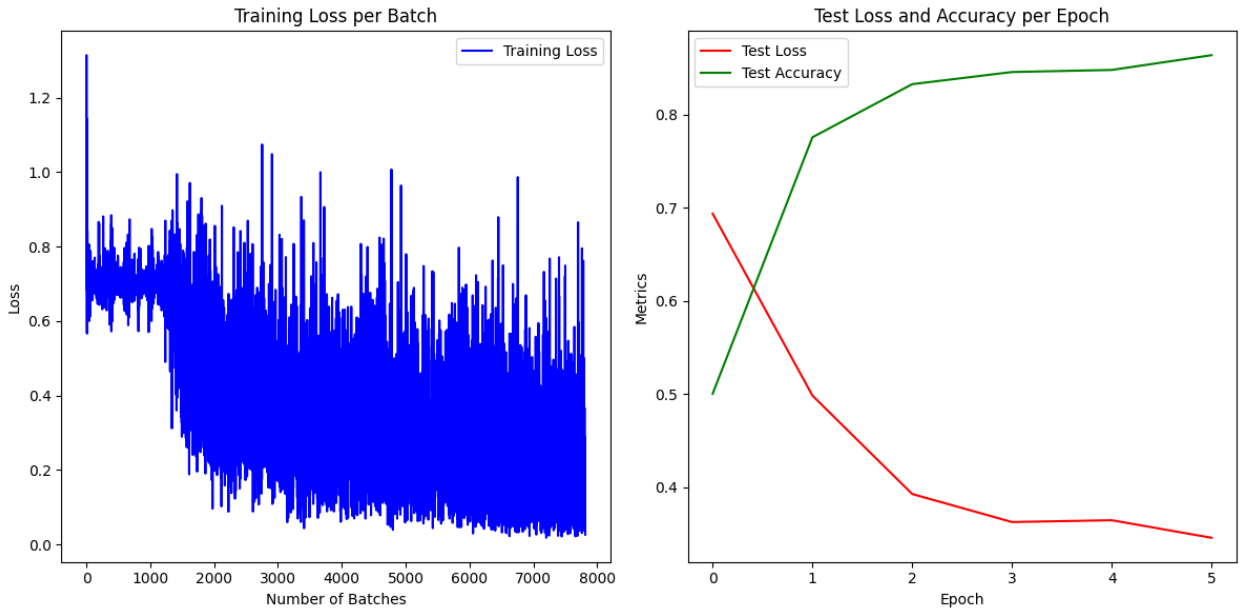

Figure 9: Training Loss per Batch (left) and Test Loss and Recall per Epoch (right) for IMDB Classification

The left plot in Figure 9 shows the training loss per batch over the entire training process. Key observations include:

- **Initial High Loss**: The loss is relatively high at the beginning of the training, indicating that the model's predictions are far from the actual labels.
- **Loss Decrease**: As training progresses, the loss decreases significantly, showing that the model is learning and improving its predictions.
- **Stabilization**: Towards the end of the training, the loss stabilizes, indicating that the model is converging.

The right plot in Figure 9 illustrates the test loss and recall per epoch. Key observations include:

- **Decreasing Test Loss**: The test loss decreases steadily over the epochs, from an initial test loss of 0.69 to 0.34 at epoch 5, indicating that the model is generalizing well to the test data.
- **Increasing Test Accuracy**: The test accuracy increases correspondingly, from an initial accuracy of 50.02% to 86.40% at epoch 5, showing that the model's performance on the test set is improving.
- **Convergence**: Both the test loss and test accuracy curves suggest that the model is converging, as the test loss plateaus and test accuracy stabilizes towards the end of the training.

Description	Test Loss	Test Recall
Initial Test	0.69	50.02%
Epoch 1	0.50	77.58%
Epoch 2	0.39	83.28%
Epoch 3	0.36	84.58%
Epoch 4	0.36	84.81%
Epoch 5	0.34	86.40%
Ref		89.35%

Table 10: Test Loss and Accuracy per Epoch

The detailed test loss and accuracy for each epoch are shown in Table 10. Note that there is a slight difference in results compared to the results from the referenced GitHub repository, which reported a test accuracy of 89.35% at epoch 5. This discrepancy can be attributed to the different training and testing splits used. The referenced repository used a training set of 34,707 samples, a validation set of 7,437 samples, and a test set of 7,438 samples, whereas our splits consisted of 25,000 samples each for training and testing.

5.4.2 Results of IMDB Dataset on the BERT Model, Regression Task

The result of training IMDB dataset on BERT model are shown in Figure 10, Table 11 and Table 12. The learning curves for both experiments show a clear decreasing trend in both training and test loss, indicating that the model is learning and improving over time. In both experiments, there is a noticeable reduction in test loss initially, which then stabilizes towards the later epochs.

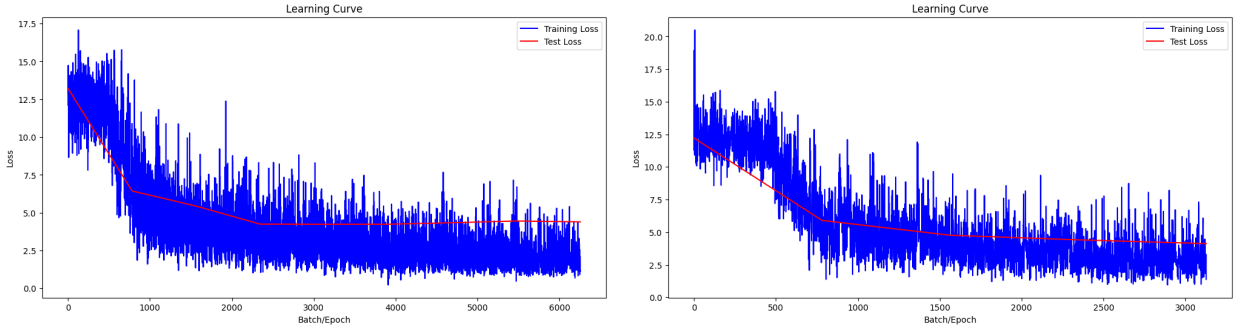

Figure 10: Training Loss and Test Loss for Two Experiments: IMDB on BERT with 8 epochs(left, Experiment 1) and 4 epochs(right, Experiment 2).

Key observations include:

- **Initial Test Loss**: The initial test loss is 13.21 for Experiment 1 and 12.22 for Experiment 2.

- **Mean Squared Error**: The MSE on the test set is 4.39 for Experiment 1 and 4.13 for Experiment 2, indicating slightly better performance in the second experiment.
- R^2 **Score**: The R^2 score on the test set is 0.64 for Experiment 1 and 0.66 for Experiment 2, suggesting better goodness-of-fit in the second experiment.

Metric	**Experiment 1**	**Experiment 2**
Initial Test Loss	13.21	12.22
Mean Squared Error (Test Set)	4.39	4.13
R^2 (Test Set)	0.64	0.66

Table 11: Comparison of Evaluation Metrics for Two Experiments

The table 11 summarizes the evaluation metrics for both experiments. The metrics show that the second experiment slightly outperforms the first in terms of MSE and R^2 on the test set.

	Test Set		**Train Set**	
	Predicted	**Real**	**Predicted**	**Real**
Experiment 1				
Predicted Score	1	0.80	1	0.87
Real Score	0.80	1	0.87	1
Experiment 2				
Predicted Score	1	0.82	1	0.85
Real Score	0.82	1	0.85	1

Table 12: Correlation Matrices for Predicted and Real Scores

The correlation matrices for the predicted and real scores in both the test and training sets indicate strong positive correlations, demonstrating the model's ability to predict the scores accurately.

5.4.3 Results of IMDB Dataset on the BERT Model with Fixed Weights and Linear Layers

The result of models with one linear layer versus two linear layers on the IMDB dataset is shown in Figure 17 in Appendix. Additionally, we compare these results to the learning curves from the myPersonality dataset, as shown in Figure 18 in Appendix.

The comparison highlights that models with one or two linear layers do not learn effectively on the IMDB dataset, as indicated by the lack of convergence and flat test loss. Both one and two linear layers perform bad on the myPersonality dataset, without showing clear learning trends.

5.4.4 Results of IMDB Datset Reasonable Size

The training results of the minimal dataset size required to maintain a good performance without overfitting on the IMDB dataset are shown in Figure 11.

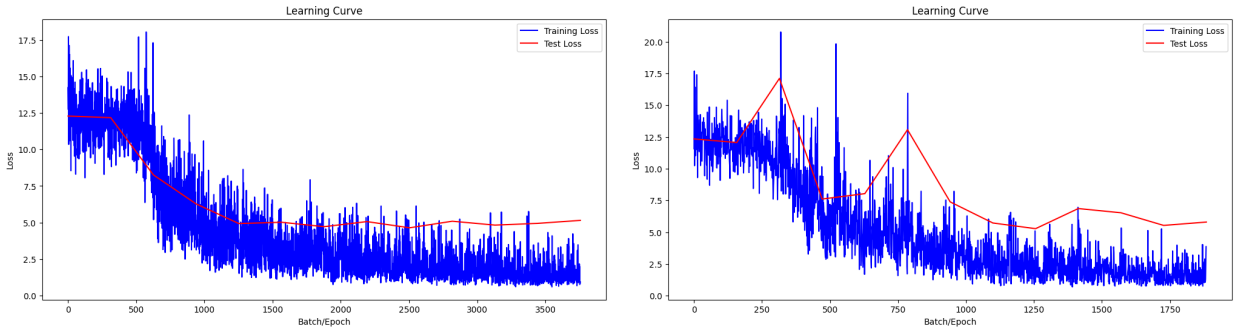

Figure 11: Training Loss and Test Loss for Reduced Dataset Sizes: 10,000 Samples Learning Curve (left), 5,000 Samples Learning Curve (Right)

The learning curves in Figure 11 show the training and test loss over epochs for different training set sizes (10,000 and 5,000 samples).The learning curve of 10,000 samples shows a decrease in both training and test loss over time. However, there is a noticeable gap between the training loss and test loss from 1500 batches, indicating potential overfitting at the end of training.

The learning curve of 5,000 samples shows more fluctuations in the test loss, indicating instability in the model's performance. The gap between training and test loss is more pronounced, suggesting increased overfitting.

Metric	**10,000 Samples**	**5,000 Samples**
Initial Test Loss	12.29	12.34
Mean Squared Error (Test Set)	5.14	5.7980
R^2 (Test Set)	0.58	0.52

Table 13: Evaluation Metrics for Reduced Dataset Sizes

The 10,000 sample model shows a reasonable performance with an MSE of 5.14 and an R^2 of 0.58 on the test set. The correlation between predicted and real scores is strong, both for the training set (0.82) and the test set (0.77).

	Test Set		Train Set	
	Predicted	**Real**	**Predicted**	**Real**
10,000 Samples				
Predicted Score	1	0.77	1	0.82
Real Score	0.77	1	0.82	1
5,000 Samples				
Predicted Score	1	0.75	1	0.79
Real Score	0.75	1	0.79	1

Table 14: Correlation Matrices for Predicted and Real Scores

The combined correlation matrices in Table 14 show the strong positive correlations between the predicted and real scores for both the test and training sets across two experiments. The values demonstrate the model's accuracy in predicting scores closely aligned with the real scores.

Reducing the training set size to 10,000 samples still allows the model to maintain a reasonable performance, although there is some overfitting. However, reducing the training set further to 5,000 samples leads to a significant increase in overfitting and instability in the model's performance. Thus, a training set size of around 10,000 samples appears to be a minimal threshold for maintaining acceptable performance without excessive overfitting.

5.5 Results of Augmented Dataset

For the first experiment, 200 user entries are selected. Each entry is used to generate approximately 20 augmented texts using GPT-4, resulting in a total of 4283 entries. The dataset was further expanded by generating 50 texts per user for the same set of 200 users, resulting in a dataset size of 10,263 entries. Then fine-tune the BERT model(with a linear layer) with the following parameters: 16 epochs, learning rate of 1e-5, and batch size of 8. The test set we use are a 50 users test set and a 250 users test set. The training results of dataset with 4283 size are shown in Figure 12. The training results of dataset with 10263 size are shown in Figure 13.

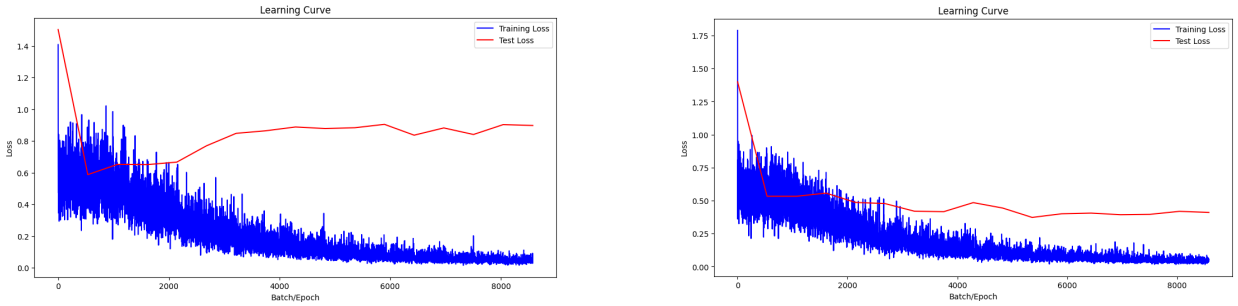

Figure 12: Training and Test Loss Curve for First Experiment with 4283 size dataset. Test set: 50 Users Left Out of Augmentation(Left Figure), test set: 250 Users, 200 in Augmentation(Right Figure).

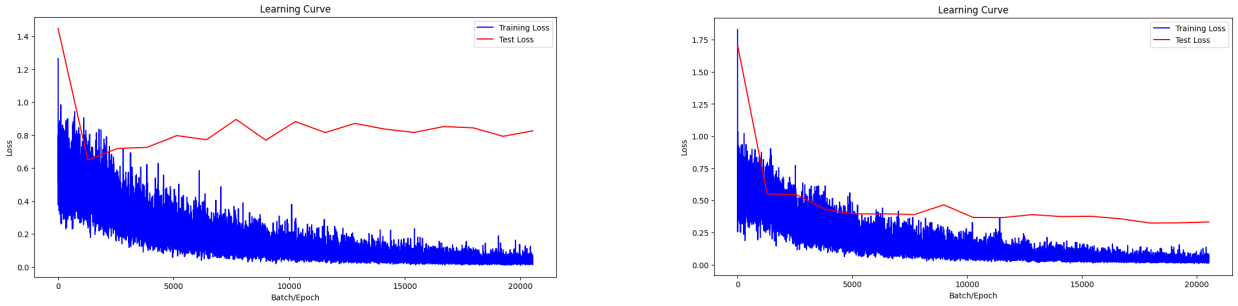

Figure 13: Training and Test Loss Curve for Second Experiment with 10263 size dataset. Test set:50 Users(left), test set: 250 Users(right).

Test Set	**Initial Test Loss**	**Mean Squared Error (MSE)**	**R^2**
4283 entries			
50 Users	1.50	0.90	-0.71
250 Users	1.40	0.41	0.21
10,263 entries			
50 Users	1.46	0.83	-0.58
250 Users	1.71	0.33	0.39

Table 15: Evaluation Metrics for BERT Model on Augmented myPersonality Dataset

In the first experiment with 4283 augmented entries, the model shows signs of overfitting on the smaller test set (50 users) as evidenced by the high MSE and negative R^2. However, when evaluate on a larger test set (250 users), the model performs remarkably better. This improvement is reflected in the lower MSE, higher R^2, and better correlation matrices.

The second experiment further validates the effectiveness of data augmentation. With 10,263 augmented entries, the BERT model shows substantial improvement across all evaluation

metrics compared to the first experiment. The results on both the smaller and larger test sets demonstrate better generalization, with reduced overfitting and enhanced predictive performance.

Despite the initial overfitting observe in the smaller test set, the data augmentation approach is effective in creating a more robust dataset. The 50-user test set, which does not include any users from the augmented training data, highlights the model's struggle with generalization due to the limited and potentially unrepresentative nature of the test set. However, the improved performance on the 250-user test set, which encompasses all the original users from the myPersonality dataset, suggests that the BERT model can capture and generalize the relevant patterns when provided with sufficient and representative data.

6 Discussion

In this section, we discuss the research questions according to the experiment results.

1. How valid and reliable is BERT in predicting personality traits based on self-written texts in CVs and LinkedIn profiles?

Our experiments aimed to evaluate the effectiveness of BERT models in predicting the Big Five personality traits from text data. The results indicate that while BERT models are capable of capturing relationships within the data, the size and quality of the dataset significantly impact the model's performance. The experiments on the original myPersonality dataset revealed substantial overfitting, suggesting that the dataset was insufficient for training a robust model. This highlights the need for larger and more diverse datasets to improve the reliability of personality predictions using NLP methodologies.

2. To what extent can BERT capture the relationship between text and OCEAN scores within the myPersonality dataset?

Initial experiments with the myPersonality dataset showed that the BERT model struggled with overfitting, as indicated by poor MSE and R^2 scores. This suggests that while the model can learn from the data, the limited size in the myPersonality dataset hinder its ability to generalize.

3. To what extent can BERT capture the relationship between text and labels in a larger and more comprehensive text-to-score dataset? What is the reasonable dataset size required for BERT to avoid overfitting, as estimated using the same model on differently sized samples of a dataset (IMDB) known to perform well with BERT?

Experiments with the IMDB dataset demonstrated that BERT models could effectively capture relationships when provided with a sufficiently large and diverse dataset. We examined by classification task and regression task. This indicates that the BERT model has the potential to perform well on personality prediction tasks, provided the dataset is large enough to support robust learning, and the relation between status update text and personality is actually there.

Through our experiments, we determined that a dataset size of around 10,000 entries is reasonable for training a BERT model to avoid overfitting. The IMDB dataset experiments provided a benchmark, showing that BERT models perform well when trained on datasets of this magnitude.

4. How does adding noise, gluing the data according to user ID, or splitting the data according to user ID affect the BERT model's performance? What is the effect of data augmentation on the BERT model's performance?

Data augmentation techniques such as adding noise, gluing data by user ID, and splitting data by user ID were tested to improve the model's performance. Augmenting the dataset with synthetic data generated by GPT-4 resulted in significant improvements in model performance, reducing overfitting and enhancing generalization. However, overfitting persisted

in smaller test sets, highlighting the challenges in ensuring data diversity.

Augmenting the myPersonality dataset to a size of approximately 10,000 entries resulted in better convergence and improved performance metrics, including lower MSE and higher R^2 scores. The enhanced dataset allowed the BERT model to learn more effectively, demonstrating that data augmentation is a viable strategy to improve model training and generalization.

However, the improvement was only observed when the entire original dataset was used as basis for the augmentation, and then tested on the originals. If we used a 50 user leave out test set, the model overfitted again. The results support previous findings that emphasize the importance of dataset size and diversity in improving model performance.

The study's limitations include the initial overfitting on the smaller dataset and the challenges associated with generalizing across different users. And the choice of augmentation techniques and model architectures could be further refined to better capture the complexities of personality traits.

7 Conclusion and Future Work

This study thoroughly examined the impact of data augmentation on the performance of BERT models in predicting personality traits using small datasets, along with insights drawn from previous experiments on the IMDB dataset. The BERT model exhibits significant overfitting when trained on the original, relatively small myPersonality dataset. This is evident from poor performance metrics on the test set. Previous experiments on the IMDB dataset, where BERT performs well with approximately 10k entries, guides the target size for the augmented myPersonality dataset. These experiments underscore the importance of an adequately sized dataset for achieving optimal model performance. By augmenting the myPersonality dataset using LLM promoting generating novel simulated users with their scores to 4283 and 10263 entries, the model's performance improves notably on a larger test set of 250 users. This demonstrates that data augmentation mitigated overfitting and helped the model generalize better. However, excluding the 50-user test set revealed significant overfitting in the initial smaller dataset. Despite this, the model's performance on the larger test set suggests that BERT could not reliably predict personality traits from the given data, pointing to the need for further refinement and possibly more sophisticated augmentation techniques.

In conclusion, this study demonstrates that while BERT models can effectively predict personality traits with sufficient data, significant challenges remain, particularly related to dataset size and user variability. Data augmentation proved to be a valuable technique in improving model performance, but only if the full dataset was used as basis for the augmentation. The future work should focus on refining these techniques and exploring more diverse datasets to enhance the robustness of personality prediction models.

For future research and practical applications, ensuring a sufficiently large and representative training dataset is crucial for achieving reliable predictive performance in personality trait prediction using BERT models. Also we should focus on exploring additional augmentation techniques, validating these findings on diverse datasets, and refining model architectures to further improve predictive capabilities. During augmented process, ethical considerations and potential societal impacts of deploying automated personality prediction systems must be carefully considered to ensure responsible use of this technology.

References

[1] Laith Alzubaidi et al. "Review of deep learning: concepts, CNN architectures, challenges, applications, future directions". In: *Journal of big Data* 8 (2021), pp. 1–74.

[2] Joshua Evan Arijanto et al. "Personality prediction based on text analytics using bidirectional encoder representations from transformers from english twitter dataset". In: *International Journal of Fuzzy Logic and Intelligent Systems* 21.3 (2021), pp. 310–316.

[3] Murray R Barrick and Michael K Mount. "The big five personality dimensions and job performance: a meta-analysis". In: *Personnel psychology* 44.1 (1991), pp. 1–26.

[4] A Colin Cameron and Frank AG Windmeijer. "An R-squared measure of goodness of fit for some common nonlinear regression models". In: *Journal of econometrics* 77.2 (1997), pp. 329–342.

[5] Haixing Dai et al. "Auggpt: Leveraging chatgpt for text data augmentation". In: *arXiv preprint arXiv:2302.13007* (2023).

[6] Jacob Devlin et al. "Bert: Pre-training of deep bidirectional transformers for language understanding". In: *arXiv preprint arXiv:1810.04805* (2018).

[7] Charles D Dziuban and Edwin C Shirkey. "When is a correlation matrix appropriate for factor analysis? Some decision rules." In: *Psychological bulletin* 81.6 (1974), p. 358.

[8] William L Gardner et al. "Matching personality and organizational culture: Effects of recruitment strategy and the Five-Factor Model on subjective person–organization fit". In: *Management Communication Quarterly* 26.4 (2012), pp. 585–622.

[9] Airlie Hilliard et al. "Eliciting Big Five Personality Traits in Large Language Models: A Textual Analysis with Classifier-Driven Approach". In: *arXiv preprint arXiv:2402.08341* (2024).

[10] Amirmohammad Kazameini et al. "Personality trait detection using bagged svm over bert word embedding ensembles". In: *arXiv preprint arXiv:2010.01309* (2020).

[11] Diederik P Kingma and Jimmy Ba. "Adam: A method for stochastic optimization". In: *arXiv preprint arXiv:1412.6980* (2014).

[12] Meera Komarraju et al. "The Big Five personality traits, learning styles, and academic achievement". In: *Personality and individual differences* 51.4 (2011), pp. 472–477.

[13] Hao Lin and Xiaolei Li. "Big five personality prediction based on pre-training language model and sentiment knowledge base". In: *Sixth International Conference on Computer Information Science and Application Technology (CISAT 2023)*. Vol. 12800. SPIE. 2023, pp. 1122–1127.

[14] Kiran Maharana, Surajit Mondal, and Bhushankumar Nemade. "A review: Data preprocessing and data augmentation techniques". In: *Global Transitions Proceedings* 3.1 (2022), pp. 91–99.

[15] Hans Marmolin. "Subjective MSE measures". In: *IEEE transactions on systems, man, and cybernetics* 16.3 (1986), pp. 486–489.

[16] Andrew Neal et al. "Predicting the form and direction of work role performance from the Big 5 model of personality traits". In: *Journal of Organizational Behavior* 33.2 (2012), pp. 175–192.

[17] Sonia Roccas et al. "The big five personality factors and personal values". In: *Personality and social psychology bulletin* 28.6 (2002), pp. 789–801.

[18] Mustafa Safdari et al. “Personality traits in large language models”. In: *arXiv preprint arXiv:2307.00184* (2023).

[19] Alex Sherstinsky. “Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network”. In: *Physica D: Nonlinear Phenomena* 404 (2020), p. 132306.

[20] Connor Shorten, Taghi M Khoshgoftaar, and Borko Furht. “Text data augmentation for deep learning”. In: *Journal of big Data* 8.1 (2021), p. 101.

[21] Stephen Soldz and George E Vaillant. “The Big Five personality traits and the life course: A 45-year longitudinal study”. In: *Journal of research in personality* 33.2 (1999), pp. 208–232.

[22] Heydar Soudani et al. “A Survey on Recent Advances in Conversational Data Generation”. In: *arXiv preprint arXiv:2405.13003* (2024).

[23] Michael M Tadesse et al. “Personality predictions based on user behavior on the facebook social media platform”. In: *IEEE Access* 6 (2018), pp. 61959–61969.

[24] Tommy Tandera et al. “Personality prediction system from facebook users”. In: *Procedia computer science* 116 (2017), pp. 604–611.

[25] Yla R Tausczik and James W Pennebaker. “The psychological meaning of words: LIWC and computerized text analysis methods”. In: *Journal of language and social psychology* 29.1 (2010), pp. 24–54.

[26] Alec Yenter and Abhishek Verma. “Deep CNN-LSTM with combined kernels from multiple branches for IMDb review sentiment analysis”. In: *2017 IEEE 8th annual ubiquitous computing, electronics and mobile communication conference (UEMCON)*. IEEE. 2017, pp. 540–546.

A Prompt in ChatGPT

Listing 1: Pseudo-Code for Text Generation

```
prompt = (f"Generate {num_texts_per_user} texts similar to '{original_text}' with similar "
          f"Big Five personality scores: {original_scores.tolist()}. "
          "Each text should be followed by the scores in the format: E_score: x.xx, N_score: x.xx, A_score: x.xx, C_score: x.xx, O_score: x.xx "
          "Each text should be about 80-120 words long. Separate each set of text and scores with '###'.")
messages=[
    {"role": "system", "content": "You are an assistant that generates text based on given personality scores."},
    {"role": "user", "content": prompt}
],
max_tokens=1000 * num_texts_per_user  # Ensure enough tokens
```

B Correlation Matrix

	Test Set					Train Set				
	E	N	A	C	O	E	N	A	C	O
Real_Extraversion	0.25	-0.12	0.07	0.08	-0.00	0.85	-0.36	0.23	0.19	0.11
Real_Neuroticism	-0.11	0.19	-0.08	-0.09	-0.01	-0.38	0.85	-0.36	-0.26	-0.17
Real_Agreeableness	0.09	-0.06	0.23	0.07	0.03	0.22	-0.32	0.85	0.04	0.22
Real_Conscientiousness	0.07	-0.09	0.06	0.17	-0.00	0.17	-0.24	0.08	0.84	0.04
Real_Openness	0.01	0.00	0.07	-0.04	0.19	0.13	-0.17	0.24	0.01	0.83
Real_Extraversion	0.06	-0.07	0.08	-0.01	-0.02	0.84	-0.40	0.22	0.16	0.14
Real_Neuroticism	0.01	0.00	0.02	0.04	-0.02	-0.41	0.85	-0.38	-0.28	-0.21
Real_Agreeableness	-0.01	0.03	0.02	-0.02	-0.02	0.26	-0.41	0.86	0.13	0.27
Real_Conscientiousness	-0.02	0.02	-0.01	-0.05	0.01	0.21	-0.24	0.13	0.82	0.00
Real_Openness	-0.07	0.10	-0.03	-0.05	0.04	0.15	-0.18	0.23	-0.01	0.82

Table 16: Correlation Matrix for Predicted and Real Scores on Test and Train Sets. Upper: Random Splits with Original myPersonality Dataset. Down: User-based Splits with Original myPersonality Dataset.

	Test Set					Train Set				
	E	N	A	C	O	E	N	A	C	O
Real_Extraversion	0.22	-0.12	0.08	0.07	-0.03	0.84	-0.35	0.20	0.23	0.12
Real_Neuroticism	-0.12	0.22	-0.08	-0.13	-0.00	-0.34	0.85	-0.31	-0.31	-0.20
Real_Agreeableness	0.11	-0.10	0.22	0.08	0.07	0.21	-0.33	0.84	0.07	0.23
Real_Conscientiousness	0.05	-0.10	0.05	0.19	0.17	-0.29	0.07	0.83	0.04	
Real_Openness	0.02	-0.02	0.01	0.06	0.13	0.13	-0.17	0.19	0.06	0.82
Real_Extraversion	0.10	-0.05	0.08	-0.02	0.00	0.85	-0.39	0.30	0.16	0.18
Real_Neuroticism	-0.01	0.02	-0.01	-0.02	-0.01	-0.40	0.85	-0.41	-0.26	-0.17
Real_Agreeableness	-0.01	0.04	-0.02	-0.02	-0.02	0.25	-0.37	0.85	0.09	0.27
Real_Conscientiousness	0.02	0.01	0.00	-0.05	0.03	0.19	-0.25	0.15	0.82	0.02
Real_Openness	-0.06	0.10	-0.04	-0.06	0.04	0.14	-0.18	0.25	-0.01	0.81

Table 17: Correlation Matrix for Predicted and Real Scores on Test and Train Sets. Upper: Random Splits with Noise myPersonality Dataset. Down: User-based Splits with Noise myPersonality Dataset.

	Test Set					Train Set				
	E	N	A	C	O	E	N	A	C	O
Real_Extraversion	-0.08	0.09	-0.24	-0.14	-0.03	0.98	-0.41	0.24	0.29	0.30
Real_Neuroticism	0.09	0.07	0.14	-0.00	0.12	-0.43	0.98	-0.43	-0.27	-0.08
Real_Agreeableness	-0.03	0.23	-0.15	-0.13	0.12	0.22	-0.42	0.98	0.12	0.26
Real_Conscientiousness	-0.04	0.12	-0.10	-0.27	0.01	0.25	-0.27	0.08	0.97	0.10
Real_Openness	0.05	-0.09	0.12	0.02	-0.02	0.27	-0.08	0.24	0.13	0.97
Real_Extraversion	0.65	-0.23	0.13	0.13	0.11	0.98	-0.39	0.24	0.21	0.19
Real_Neuroticism	-0.27	0.68	-0.34	-0.12	-0.09	-0.42	0.98	-0.45	-0.24	-0.07
Real_Agreeableness	0.20	-0.32	0.68	0.11	0.20	0.20	-0.37	0.97	0.06	0.26
Real_Conscientiousness	0.11	-0.08	0.09	0.69	0.06	0.24	-0.22	0.08	0.98	0.01
Real_Openness	0.15	-0.05	0.21	0.04	0.74	0.26	-0.06	0.26	0.07	0.96

Table 18: Correlation Matrix for Predicted and Real Scores on Test and Train Sets. Upper: Augmented 5K myPersonality Dataset. Down: Augmented 10K myPersonality Dataset.

C Figures

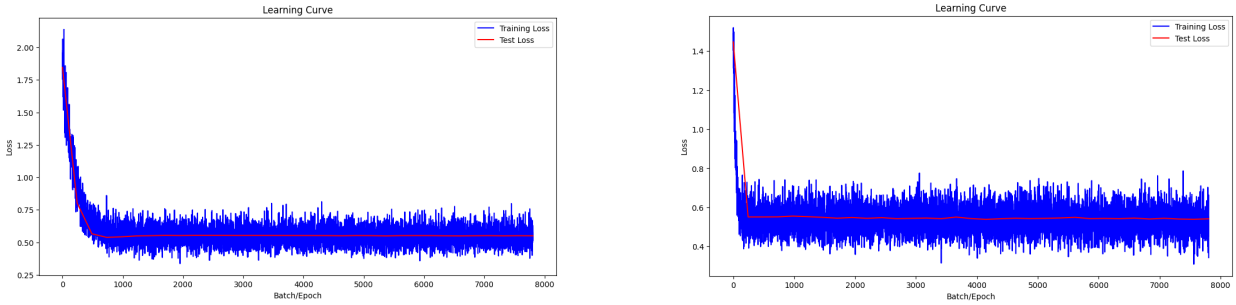

Figure 14: Comparison of Learning Curves for Different Layers: Learning Curve for One Linear Layer(left), Learning Curve for Two Linear Layers(right), Both Using BERT Model with Original myPersonality Dataset.

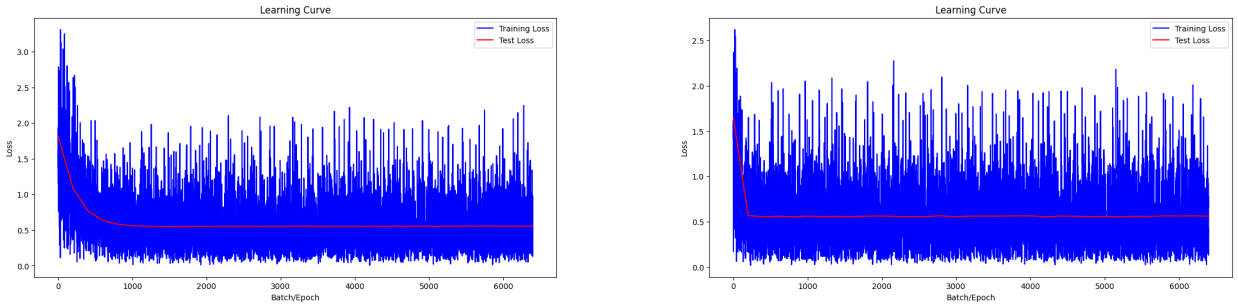

Figure 15: Comparison of Learning Curves for Different Layers: Learning Curve for One Linear Layer(left), Learning Curve for Two Linear Layers(right), Both Using BERT Model with Glued myPersonality Dataset.

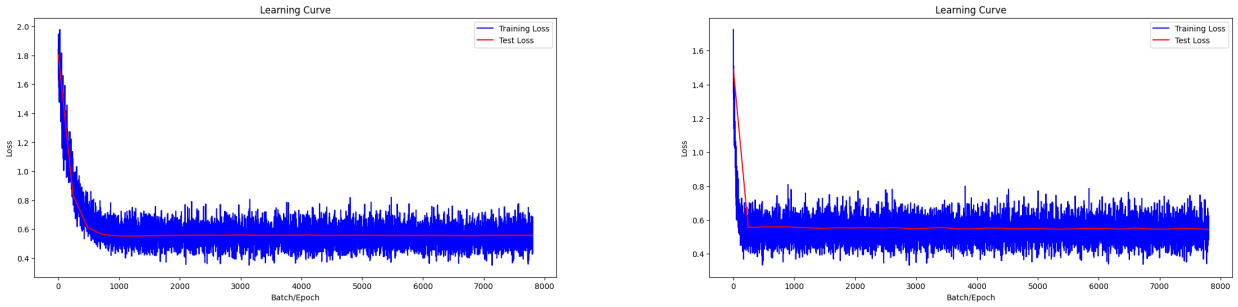

Figure 16: Comparison of Learning Curves for Different Layers: Learning Curve for One Linear Layer(left), Learning Curve for Two Linear Layers(right), Both Using BERT Model with Noisy myPersonality Dataset.

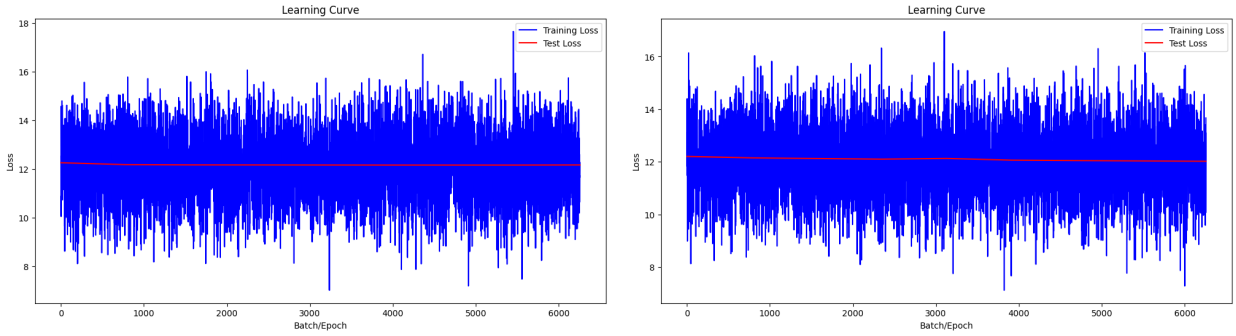

Figure 17: Learning Curve for IMDB Dataset with One Linear Layer(left), Learning Curve for IMDB Dataset with Two Linear Layers(right).

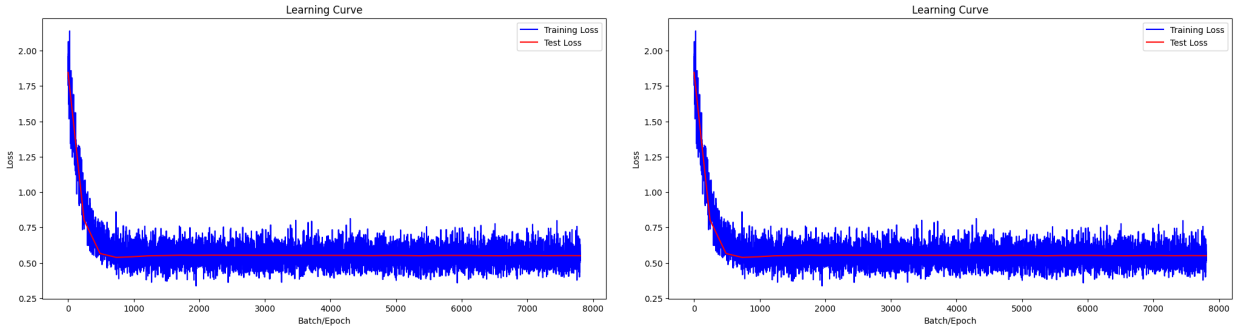

Figure 18: Learning Curve for myPersonality Dataset with One Linear Layer(left), Learning Curve for myPersonality Dataset with Two Linear Layers(right), Both on Original myPersonality Dataset.