BILINGUAL PUBLISHING GROUP

Forum for Linguistic Studies

https://journals.bilpubgroup.com/index.php/fls

ARTICLE

Detection of Alzheimer's Disease Using Fine-Tuned Large Language Models

Baha Ihnaini [1*] , ***Yongxin Deng*** [1] , ***Yujie He*** [1] , ***Le Geng*** [1] , ***Jiyai Xu*** [2]

[1] *The College of Science, Mathematics and Technology, Wenzhou-Kean University, Zhejiang 325027, China*

[2] *The College of Liberal Arts, Wenzhou-Kean University, Zhejiang 325027, China*

ABSTRACT

Since there is no known cure for Alzheimer's disease (AD), early detection is essential to controlling its progression. Because of the high cost and invasiveness of traditional diagnostic techniques like MRIs and pathological testing, researchers are looking into less expensive alternatives that use machine learning (ML) and natural language processing (NLP). By evaluating their performance against traditional ML and deep learning (DL) techniques, this study explores the possibility of using fine-tuned open-source large language models (LLMs) to identify AD through linguistic analysis. To optimize models like Qwen1.5–7B and OLMo1.7–7B, we used supervised fine-tuning (SFT) with parameter-efficient techniques like LoRA and QLoRA on the Pitt Corpus dataset, which consists of speech transcripts from the "Cookie Theft" picture description task. The findings showed that LLMs performed noticeably better than conventional techniques; Qwen1.5–7B had an F1-score of 0.8824, which was higher than CNN (0.7987), LSTM (0.7689), and logistic regression (0.83). The study demonstrates how LLMs can detect subtle linguistic impairments in AD that are difficult for traditional models to identify, like syntactic errors and repetitions. The comparatively small dataset size and exclusive reliance on textual data are limitations, though, and it is recommended that future studies include multimodal inputs and more varied datasets. Despite limitations, the results highlight the potential of optimized LLMs as scalable, non-invasive methods for early AD

*CORRESPONDING AUTHOR:

Baha Ihnaini, The College of Science, Mathematics and Technology, Wenzhou-Kean University, University Road, Wenzhou, Zhejiang 325027, China; Email: bihnaini@kean.edu

ARTICLE INFO

Received:6 May 2025 | Revised: 26 May 2025 | Accepted: 10 June 2025 | Published Online: 1 August 2025
DOI: https://doi.org/10.30564/fls.v7i8.9899

CITATION

Ihnaini, B., Deng, Y., He, Y., et al., 2025. Detection of Alzheimer's Disease Using Fine-Tuned Large Language Models. Forum for Linguistic Studies. 7(8): 373–384. DOI: https://doi.org/10.30564/fls.v7i8.9899

detection, providing a way to enhance patient care and diagnostic precision. Through this study, a novel, accurate, and reliable method for early diagnosis of Alzheimer's disease patients can be provided.

Keywords: Alzheimer's Disease; Large Language Models; Natural Language Processing; Supervised Fine-Tuning

1. Introduction

Alzheimer's disease (AD) is a progressive neurodegenerative disorder without effective treatment and irreversibly affects memory and thinking skills, ultimately interfering with the ability to perform simple tasks[1]. It mainly manifests itself through blurred or lost memory and impairment of cognitive, linguistic, and executive functions, which seriously affect the independence of daily life and the quality of life of patients. However, early diagnosis and prevention can only help to delay the development of Alzheimer's disease. Although AD is more common among older adults, it is not considered a normal part of aging. However, as the global population ages, the prevalence of it has increased significantly.

As reported in the 2019 World Alzheimer's Disease Report, the Alzheimer's Disease International (ADI) projected that over 50 million individuals worldwide were living with dementia. The estimate for 2050 was expected to reach more than 150 million. In the same year, the annual cost of dementia in the United States alone was expected to exceed $600 billion[2], which imposes a huge economic burden on individuals, families, and society. Therefore, in recent years, in addition to the common pathological test, magnetic resonance imaging (MRI)[3], researchers have tried to develop inexpensive non-invasive methods to diagnose Alzheimer's patients early. Recent studies have demonstrated the feasibility of using natural language processing (NLP) and machine learning techniques to identify early Alzheimer's disease[4]. These methods are low-cost and non-invasive, making them accessible to patients and their families. They are also effective for facilitating the early detection of the disease.

Current methods for diagnosing Alzheimer's disease still suffer from limited accuracy and reliability, especially in early detection. Recent advances in large language models (LLMs) and their success in text-based tasks have opened new possibilities for improving diagnostic performance[5]. For example, Yuan et al. suggested using fine-tuned BERT models to recognize language interruptions (e.g., pauses and repetitions) in patients with Alzheimer's disease[6]. This approach achieved good classification results at Interspeech 2020, indicating that pre-trained language models can capture features of language disorders. This work laid a foundation for further exploration of large language models in AD detection. This study aims to evaluate whether optimized open-source large language models (LLMs) can outperform traditional machine learning (ML) and deep learning (DL) methods in Alzheimer's disease detection tasks[4,7]. This study will adopt open-source large-scale language models as transfer learning method models, which are relatively advanced and better than the basic large-scale language models of other language models[5]. By fine-tuning LLMs, we expect them to show better performance for AD diagnostics than traditional ML and DL models. The purpose of this study was to evaluate the effectiveness of the fine-tuned LLMs, including examining their performance in terms of diagnostic accuracy in terms of post-test f1 scores and reliability and finally comparing it to other existing ML models and DL models. Our goal is to contribute a scalable and accurate approach to assist early-stage Alzheimer's diagnosis based on language features.

2. Literature Review

Ning Liu and his team demonstrated that the use of transfer learning techniques could improve model performance in the field of Alzheimer's disease (AD) detection[4]. Their team focused on extracting linguistic features from a speech dataset collected by the team on "A picture of a Boston Cookie-Theft description task" and then applied transfer learning to the data containing AD features. Their team found that after applying this training method to the model, ERNIE+Pause was the champion model, performing the best in a column of measures, with an Accuracy of 0.896, an F1-score of 0.889, and a Precision of 0.952. So, using the transfer learning technique allows the model to focus on detecting AD and perform significantly better than traditional AD detection techniques. The performance in terms of diag-

nosing AD is significantly better than traditional methods of diagnosing AD.

Matosevic and Jovic's research used BERT model as a baseline model and compared the results from BERT and RoBERTa. The dataset is the Pitt Corpus, which asked people to describe the Cookie Theft picture, showing that NLP method performed better in AD detection [7]. In this work, they used linguistic transcripts, which are in the CHAT format. The initial preprocessing phase involved extracting the speech content from the participants. Consequently, they removed the speech from examiners and eliminated any details pertaining to morphological and grammatical relationships in the transcripts. Following that, a decision was made regarding which participant speech information should be retained and which should be discarded. Then the researchers trained the setup. They both used RoBERTa and BERT with 256 and 512 tokenizers. In the result, they found that their models are good at larger text spans. They also showed that pre-trained transformer models like RoBERTa can deliver substantial outcomes when it comes to identifying dementia through speech transcripts. Their top-performing model, trained on transcripts containing repetitive speech segments, achieved an accuracy rate of 90.16%.

Liu et al. proposed a transfer learning method for detecting Alzheimer's disease based on speech and natural language processing. Their research uses transfer learning and natural language processing technology to diagnose Alzheimer's disease (AD) [8]. Its result improves the AD prediction with high accuracy and solves the problem of a lack of datasets. This project mainly consists of the distilBert and the logistic regression. Although BERT is popular, the model it chose could be faster and smaller architecture, and it could "retain 97% language understanding capability of the BERT model". In this research, the distilBert model is employed to capture profound semantic characteristics, and these extracted features are subsequently fed into a logistic regression model for the purpose of sentence classification. The primary procedures can be outlined as follows: Initially, words are segmented into tokens utilizing the distilBert tokenizer, and specific terms (namely, "(CLS)" inserted at the sentence's start and "(SEP)" at the conclusion) are incorporated into the text. Subsequently, the pretrained model's vocabulary is consulted to exchange these tokens with their corresponding numeric representations, which are then utilized in the Distil Bert model to generate a 768-dimensional output vector. Finally, this vector is fed into a logistic regression classifier, yielding the ultimate binary classification outcome. The study used the ADReSS Datasets, which include 78 AD and 78 Normal controls, and the participants described the Cookie Theft picture in detail. This research achieved 0.896 accuracy by Enhanced Language Representation with the Informative Entities (ERNIE) model, and to check the influence of different classifiers, it found that logistic regression performed best.

Liu and Yuan conducted a study using NLP method to detect AD and explore lexical performance, using the comparison with latest deep learning, with small Chinese datasets [9]. In this study, researchers utilized datasets sourced from the Predictive Challenge of Alzheimer's Disease organized by iFlytek in 2019 for training and validating their model. Their experimental results show that their model outperforms the top-performing model in the 2019 binary classification competition, achieving an F1 score of 98% compared to the competition's score of 75.4%. Additionally, they calculated the proportion of nouns and verbs in the linguistic descriptions, aligning with international research on linguistic characteristics in AD patients. The study employed the k-nearest neighbor (KNN) algorithm to differentiate between individuals with Alzheimer's disease (AD) and those without (CTRL) based on their transcripts. This method involves assigning the unknown data to the category that shares the closest proximity. Initially, all transcripts within the same category are aggregated. Subsequently, researchers compute the distance between the unknown data and the two categories under consideration. Ultimately, they determine the unknown category based on this distance calculation. The training process consists of text preprocessing, word segmentation, keyword extraction, text vectorization, and similarity calculation. The dataset in this study is Corpus, which is famous for describing the Cookie Theft picture. There are 68 patients with AD, 144 MCI patients, and 111 controls in this experiment. The result shows that their model could reach the best 97.77% accuracy and is stable when the number of features is over 835. It also compared the results of using deep learning models. However, the BERT model in deep learning performed best, which was still not as well as their method.

3. Materials and Methods

3.1. Dataset

In this study, the data set is taken from the Pitt Corpus[8], which is accessed through the DementiaBank website[8]. This dataset consists of descriptive texts related to the "cookie theft" picture, designed by Goodglass and Kaplan[9]. Participants were placed in a quiet room where a clinician asked them to provide as much detail as possible in describing the "cookie theft" image. The recordings and texts of these descriptions were collected by the University of Pittsburgh. Entries with other dementia diagnoses and unclear labels were removed, ultimately obtaining data from 242 controls and 256 individuals with possible or probable Alzheimer's Disease. Following is a description of the demographic and clinical characteristics of controls and AD group, illustrating differences in age, education, sex distribution, and cognitive performance (reflected by the Mini-Mental State Exam (MMSE)): The average age of the control group is 65.2 years, and its standard deviation is 7.8 years, whereas the possible/probable AD group is older. The average age of the AD group is 71.8 years, and the standard deviation is 8.5 years. In terms of gender, the control group was composed of 86 males and 156 females, while the probable suspected Alzheimer's disease (AD) group was composed of 90 males and 166 females. The control group had an average of 14.1 years of education with a standard deviation of 2.4 years, while the AD group had an average of 12.5 years with a standard deviation of 2.9 years. Regarding cognitive functioning, the control group had a higher mean MMSE score of 29.1, indicating better cognitive functioning, while the AD group had a mean MMSE score of 18.5, indicating more impaired cognitive functioning.

The dataset is split into a training subset and a test subset in a ratio of 80:20. Specifically, 80% of the data is randomly selected to form the training set for model training and development, and the remaining 20% data is selected from the test set to evaluate model accuracy. This approach follows the well-established methods in the field, and Liu's study[4] adopts a similar approach, which also divides the training set and test set in the ratio of 80:20.

3.2. Traditional Machine Learning

This study selected two widely applied classification models in machine learning: support vector machine (SVM) and logistic regression (LR)[9,10]. Among them, SVM is a supervised learning model mainly for classification and regression tasks[11]. The core idea of SVM is to find the optimal hyperplane by hyperplane, which separates two types of data points and maximizes the difference between them[12]. Support vector machine for text classification is implemented by using the scikit-learn library in Python. Firstly, text and labels are extracted from the dataset. Then, the text data is transformed into feature vectors in the form of sparse matrices using the Count Vectorizer's fit_transform method in the feature extraction stage. The SVM classifier is initialized once the data is ready, and the model is trained on the training set. For linearly differentiable data, SVM looks for a hyperplane that maximizes the interval. For non-linearly differentiable data, SVM allows data to be linearly differentiable in high-dimensional space by using kernel functions that map the data into high-dimensional space, such as the polynomial kernel and the radial basis kernel. After the model is trained, classification prediction is performed on the test set, and prediction labels are generated. The accuracy, F1 score, recall, and precision of the model are calculated using a weighted average method, and the results of these model performance metrics are output.

On the other hand, LR is also a supervised learning model, which is commonly used in binary classification problems[13]. LR converts the output of a linear model into probability values by using a logical function, which is the Sigmoid function, thereby separating the data linearly and predicting the probability that the sample belongs to a certain class, thus achieving classification. Especially in binary classification problems, LR can effectively classify data[14]. In this project, during data loading and preprocessing, text and labels are first extracted from the dataset, and then the text data is converted into feature vectors in the form of sparse matrices, which is a key step in converting the textual data into a numerical form that the model can handle. In the model training phase, one instance of the Logistic regression model is initialized, followed by training the model using the training set. LR estimates the parameters by maximizing the likelihood function and maps the output of the linear model to the probability values for classification using the Sigmoid function (logistic function). For each sample, LR

calculates the probability that it belongs to a certain category and classifies it according to a probability threshold of 0.5. Once the training is complete, the trained model is used to predict the test set. The prediction process involves feeding the features of the test samples into the model and outputting the predicted categories. Finally, the accuracy, F1 score, recall, and precision of the model are calculated.

3.3. Deep Learning

Deep learning is a branch of ML[15]. It learns based on how the human brain works and finds patterns to understand difficult data. It can process images[16], voices, and do the NLP work[17]. Some different layers are added to the deep learning model compared to the machine learning model. Some early layers recognize the characteristics and representations and feed them to the model. Later layers are in charge of synthesizing the traits to recognize objects and forecast the results[15].

LSTM (long short-term memory) is a particular category of RNN that solves the problem of exploding and vanishing gradients when processing long sequences[18]. It can store information for prolonged durations with developed memory cells.

The LSTM model used in this project is mainly used to process time series data in order to extract linguistic features related to dementia and to accomplish the binary classification task. The model structure is as follows:

(1) Embedding Layer
Vocabulary size: 5000
Output dimension: 12
Input sequence length: 100

(2) LSTM Layer
Hidden units: 128
Dopout: 0.2
Recurrent dropout: 0.2

(3) Dense Layer
Units: 1
Activation: Sigmoid

CNN (convolutional neural network) utilizes several convolutional kernels to extract text features, then passes through pooling layers and connected layers to the classifier. CNN has good performance in intention classification[19].

For model performance comparison, this study also constructs a text classification model based on CNN for extracting local linguistic features. The structure is as follows:

(1) Embedding Layer
Vocabulary size: 5000
Output dimension: 128
Input sequence length: 100

(2) Conv1D Layer
Filters: 128
Kernel size: 5
Activation: ReLU

(3) MaxPooling1D Layer
Pool size: 2

(4) Dropout Layer
Dropout rate: 0.2

(5) Dense Layer
Units: 128
Activation: ReLU

(6) Dropout Layer (reused)
Dropout rate: 0.2

(7) Final Dense Layer
Units: 1
Activation: Sigmoid

3.4. Fine-tuning LLMs

3.4.1. Supervised Fine-Tuning

In this project, a fine-tuning of the open-source LLMs is used to train these pre-trained LLMs further using a text dataset dedicated to the task of describing pictures for detecting AD in order to adapt the model to the task of early detection. Supervised fine-tuning (SFT) was chosen for this project, which is the use of training datasets with the labels "dementia" and "control" to provide target outputs during fine-tuning to address the low-resource challenges of Alzheimer's disease, as it reduces the reliance on large amounts of labeled data by employing self-supervised learning, thus reducing the number of pre-trained LLMs[20].

Supervised fine-tuning of the model requires setting specific parameters, and in this project, the initial learning rate was set to 5e−4 to allow for larger adjustments to the model parameters in the early stages of training. To prevent the gradient explosion phenomenon and improve the training stability, the gradient trimming was set to 1.0[21]. In addition, each dataset contained a maximum of 398 samples to control

the size and time of training. And bf16 mixed precision training is selected to improve the computational efficiency[21], and the truncation length, which is the maximum length of the input sequence, is set to 512 bytes. Besides, two samples are processed per GPU to adapt to the GPU memory capacity. The gradient accumulation step is set to 8 steps to achieve a larger effective batch size with memory constraints. The cosine scheduler was chosen in order to decrease the learning rate gradually during the training process to improve the stability and final model performance[22]. LoRA dropout was set to 0.2 to improve the model's generalization ability and reduce the risk of overfitting. During the training, model performance was also evaluated using 10% of the data as a validation set, so that the epoch could be selected from the validation loss values derived from the training, and the appropriate epoch could ensure that the model had enough training time to avoid overfitting[23]. In this study, the number of training rounds was finally chosen to be set to 2.0 after combining all the tested models, as this gives the best accuracy. Furthermore, the fine-tuning training in 4-bit, 8-bit, and no-quantization experiments was conducted on all models in order to observe the changes in model performance and efficiency under different quantization strategies.

3.4.2. Low-Rank Adaptation (LoRa)

The two main techniques used for fine-tuning in this project are Low-Rank Adaptation (LoRA) and QLORA.

LoRA was proposed by Hu et al. in 2021[24]. It can help to efficiently fine-tune large language models. Unlike traditional methods, it does not completely retrain all parameters, but simply freezes the weights with the training model and applies trainable low-rank matrices to every layer of the transformer. This enables LoRA to reduce the number of parameters to be trained for downstream tasks[25]. This not only maintains the quality of the model but also improves the throughput of training without leading to inference delays. The main formula for LoRA is shown below. In the formula, W_0 represents the original weight matrix, and A and B refer to the two low-rank matrices, respectively.

$$h = W_0 x + BAx \tag{1}$$

When using LoRA in this project, first take a large model that has been trained. The weights of this model and the features it has learned will not change in subsequent training. While keeping the original weights constant, two smaller matrices are added to each layer of the model, which are multiplied together to produce a new weight update. This update is specifically designed to fit the training task of this project.

There are three advantages of using the LoRA technique to fine-tune the model in this paper. First, LoRA reduces the number of parameters that need to be trained. Since it only adds small matrices and most of the original model's weights remain the same, the number of parameters that need to be trained is greatly reduced. Second, it improves the throughput in training. Because the training parameters are reduced, fewer computational resources are required to train the model. As a result, more data can be processed during training, and the training is faster. Third, LoRA also does not increase inference latency. Using LoRA to fine-tune a large model does not change the structure of the entire model, so it does not add computational steps and therefore does not increase latency.

QLoRA was proposed by Dettmers et al[26]. It combines quantization techniques on top of LoRA to reduce memory usage. This approach makes it possible to fine-tune the 65B parameter of the model using only a single 48GB GPU, greatly reducing the need for expensive hardware resources. QLoRA takes a large, trained model and compresses it down to 4-bit (or 8-bit), which itself is not altered during training. It then passes the training signals into LoRA to tune these modules. This approach maintains the efficiency of the original model while allowing fine-tuning for specific tasks[27].

In this paper, the use of QLORA has the following benefits: first, it reduces memory usage and supports us in fine-tuning the LLMs on a single GPU. Second, with reduced hardware requirements, the QLORA-fine-tuned model achieves comparable performance to the unquantized fine-tuning, as will be shown by specific run results in the subsequent sections.

3.5. Evaluation

To assess the performance of the proposed model, we employed four standard classification metrics: accuracy, precision, recall, and F1 score. These metrics were computed based on the counts of True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN). The corresponding formulas are defined as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (2)$$

$$Precision = \frac{TP}{TP + FP} \quad (3)$$

$$Recall = \frac{TP}{TP + FN} \quad (4)$$

$$F1score = 2 \times \frac{Precision * Recall}{Precision + Recall} \quad (5)$$

The F1 score was chosen as the primary performance metric because it is highly effective in addressing class imbalance and strikes a balance between false positives and false negatives — a critical consideration in clinical diagnostic tasks such as the detection of Alzheimer's disease.

Accuracy reflects the overall correctness of the classification; recall indicates the model's ability to correctly identify cases of Alzheimer's disease (sensitivity); and precision reflects the model's reliability in predicting a positive label. Together, these metrics provide a comprehensive assessment of the model's diagnostic ability.

These metrics are reported for a variety of large-scale language models and quantitative settings, as discussed in the next section. In this study, supervised fine-tuning was used on several open-source large language models (LLMs) to detect Alzheimer's disease using linguistic transcripts. Models such as Qwen1.5–7B and OLMo1.7–7B achieved F1 scores of over 0.88, outperforming traditional machine learning and deep learning baselines on all metrics. These results suggest that optimised LLMs can effectively capture the linguistic impairments. This capacity may also generalise to other neurocognitive conditions with language-related symptoms, such as mild cognitive impairment. Unlike earlier studies [28], which focused on fundamental model adaptation, our research involved the precise customisation of each model. This entailed adjusting learning rates, training epochs, batch sizes, gradient accumulation steps, and LoRA dropout rates to strike a balance between computational efficiency and diagnostic accuracy. These tuning strategies were designed to enable the models to better capture language-level indicators of AD, which often include subtle syntactic errors and repetition patterns. These require long-context understanding.

This paper employs parameter-efficient tuning (PET), which uses newly added parameters to study the scaling. It further researches its scaling relationship with some factors, such as model size, data size, or PET parameter size [29]. The multiplicative joint scaling law for LLMs finetuning is as follows:

$$\widehat{L}(X, D_f) = A * X^{-\alpha} * D_f^{-\beta} + E \quad (6)$$

where $[A, E, \alpha, \beta]$ are parameters to be used, D_f is the data size, and X denotes other parameters [27].

4. Results

In a uniform manner, this study sets the same training hyperparameters during supervised fine-tuning. The main parameters include: the learning rate is set to 5e−4, the training period is 2, the maximum number of samples is 398, the input truncation length is limited to 512, the number of warm-up steps is set to 2, the Lora dropout rate is 0.2, and the rest of the hyper-parameters follow the default settings of LLaMA.

The F1 scores of the models with different quantization accuracies (4-bit, 8-bit, and unquantized) are shown in **Table 1**.

Table 1. F1 Scores of Different Models.

Model	F1 Score (Q = none)	F1 Score (Q = 4)	F1 Score (Q = 8)
Qwen1.8B	0.5926	0.7872	0.7356
Qwen1.5–0.5B	0.7174	0.7789	0.8043
Qwen1.5–1.8B	0.7872	0.7778	0.5926
Qwen1.5–4B	0.6250	0.7609	0.6829
Qwen1.5–7B	0.7640	0.8824	0.8041
Qwen1.5–14B	0.5542	0.7209	0.6966
Qwen2–0.5B	0.6897	0.5116	0.5421
Qwen2–1.5B	0.8119	0.6408	0.8431
Qwen2–7B	0.6170	0.7312	0.5417

Table 1. *Cont.*

Model	F1 Score (Q = none)	F1 Score (Q = 4)	F1 Score (Q = 8)
Gemma–2B	0.5682	0.6512	0.7191
Gemma–7B	0.7129	0.6604	0.6591
Gemma2–9B	0.5200	0.5200	0.5200
LLaMA–7B	0.4054	0.4872	0.4324
LLaMA2–7B	0.6265	0.5679	0.4865
Llama3–8B	0.6667	0.8785	0.6392
OLMo1.7–7B	0.8381	0.8627	0.8846
Falcon–7B	0.7640	0.7447	0.7527
Mistral–7B–v0.1	0.8515	0.6813	0.3944
Mistral–7B–v0.3	0.6667	0.7708	0.8319

Table 2 summarizes some of the better performing supervised fine-tuning models, as well as the test results of traditional deep learning and machine learning models. It can be observed that the F1 scores of the traditional methods are generally low, so this study focuses on the performance of the new models. After supervised fine-tuning, the new large language models generally outperform the traditional models in terms of F1 scores, and not only in terms of F1 scores, but also in terms of other evaluation metrics.

Figure 1 illustrates the F-1 scores obtained from fine-tuning various sizes of the QWEN–1.5 model, ranging from 0.5 billion to 14 billion parameters. The results are reported for different quantization options, including 4-bit, 8-bit, and no quantization. The x-axis represents the quantization options, the y-axis represents F-1 scores, different colors represent different sizes of Qwen1.5 models, and the size of the circle represents the size of the model. **Figure 1** shows that the 7-bit model has the best performance; a larger size doesn't mean better performance. It has the trend that, when the size is smaller than 7 bits, the F1 score is higher with the larger size; when the size is larger than 7 bits, the result decreases.

Table 2. The Result of Different Models.

	Accuracy	Precision	Recall	F1-score
SFT-llms				
Llama3–8B (Q = 4)	0.8687	0.8704	0.8785	0.8868
Qwen1.5–7B (Q = 4)	0.88	0.9184	0.8491	0.8824
Qwen2–1.5B (Q = 8)	0.8384	0.8776	0.8113	0.8431
OLMo–7B (Q = none)	0.85	0.9130	0.7925	0.8485
OLMo–7B (Q = 8)	0.84	0.8364	0.8679	0.8519
OLMo1.7–7B (Q = 4)	0.8586	0.898	0.838	0.8627
OLMo1.7–7B (Q = 8)	0.8788	0.902	0.8846	0.8679
Deep Learning				
CNN	0.8	0.8	0.8	0.7987
LSTM	0.77	0.7693	0.77	0.7689
Machine Learning				
LR	0.84	0.78	0.89	0.83
SVM	0.82	0.82	0.82	0.82

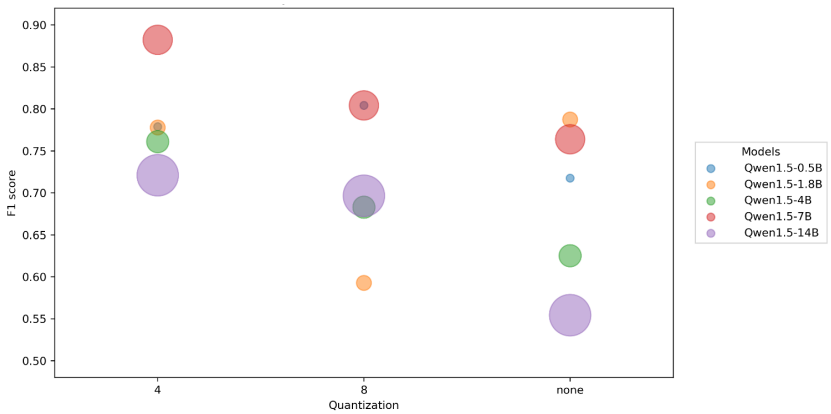

Figure 1. F1 Score by Quantization for Different Models.

5. Discussion

In this study, we employed supervised fine-tuning on several open-source large language models (LLMs) to detect Alzheimer's disease through linguistic analysis of transcripts. Models such as Qwen1.5–7B and OLMo1.7–7B achieved F1 scores of over 0.88, outperforming traditional machine learning and deep learning baselines across all metrics. These results suggest that optimised LLMs can accurately identify linguistic impairments associated with AD. The same capacity may also be applicable to other neurocognitive conditions that present with language-related symptoms, for example, mild cognitive impairment. Unlike earlier studies, which focused on fundamental model adaptation [30], our research involved the precise customisation of each model. This entailed adjusting learning rates, training epochs, batch sizes, gradient accumulation steps, and LoRA dropout rates to strike a balance between computational efficiency and diagnostic accuracy. These tuning strategies were designed to enable the models to better capture language-level indicators of AD, which often include subtle syntactic errors and repetition patterns. These require an understanding of the full context, a strength of LLMs.

To compare different models, this project employs the same parameters when training models, such as epoch = 2. However, larger models need more epochs to reach their best result, while smaller models may need fewer epochs. This research chooses several models that have better performance to find the favorite epoch, like Qwen1.5–7B (quantization four and quantization 8), OLMo1.7–7B (quantization four and quantization 8), and OLMo–7B (quantization eight and quantization none). Researchers used a validation size of 0.1 when training to determine validation loss [28].

Tables 3–5 show the validation loss of several models with different epochs. The model is overfitting when the validation loss is increasing [31]. In the figures, the overfitting point is highlighted. After the overfitting point, the validation loss increases, so that epoch is the most suitable epoch parameter. OLMo1.7–7B (quantization four and quantization 8) uses epochs 4 and 5, OLMo–7B (quantization none and quantization 8) better employs epochs 3, and model Qwen1.5–7B (quantization four and quantization 8) is suitable for epochs 3 and 4.

Table 3. The Validation loss OLMo1.7–7B.

OLMo1.7–7B (Q = 4)		**OLMo1.7–7B (Q = 8)**	
Epoch	Validation Loss	Epoch	Validation loss
1	0.2782	1	0.2627
2	0.155	2	0.2367
3	0.1197	3	0.1079
4	**0.1083**	4	0.0982
5	0.1333	**5**	**0.0848**
6	1.2244	6	0.1501

Table 4. The Validation Loss of OLMo–7B.

OLMo–7B (Q = none)		**OLMo–7B (Q = 8)**	
Epoch	Validation Loss	Epoch	Validation loss
1	0.3525	1	0.2971
2	0.2693	2	0.1497
3	**0.085**	**3**	**0.0822**
4	0.1069	4	0.097

Table 5. The Validation Loss of Qwen1.5–7B.

Qwen1.5–7B (Q = 4)		**Qwen1.5–7B (Q = 8)**	
Epoch	Validation Loss	Epoch	Validation Loss
1	0.2745	1	0.2765
2	0.2745	2	0.3644
3	**0.1134**	3	0.1369
4	0.15	**4**	**0.1307**
5	0.136	5	0.1309
6	0.1155	6	0.2627

Table 3 presents the validation loss across different epochs for the model OLMo1.7–7B under two quantization levels (Q = 4 and Q = 8). The highlighted rows indicate the epochs with the lowest validation loss for each quantization level.

Table 4 shows the validation loss across different epochs for the model OLMo–7B under two quantization levels (Q = none and Q = 8). The highlighted rows indicate the epochs with the lowest validation loss for each quantization level.

Table 5 shows the validation loss across different epochs for the model Qwen1.5–7B under two quantization levels (Q = 4 and Q = 8). The highlighted rows indicate the epochs with the lowest validation loss for each quantization level.

Despite the encouraging results, we are acutely aware of the inherent limitations in our scientific approach. Firstly, the Pitt corpus is widely used. However, its relatively small sample size may limit the generalisability of the results. Secondly, although adapter-based fine-tuning is quick, full fine-tuning or the use of multimodal inputs may achieve a higher level of correct diagnosis. Thirdly, our assessment focused solely on linguistic features, which may not fully reflect the cognitive decline associated with AD.

6. Conclusions

In this study, we demonstrate that a large-scale language model optimised using hyperparameters can effectively detect AD from speech transcripts, outperforming traditional ML and DL methods. Using adapter-based fine-tuning techniques such as LoRA and QLoRA, we achieved high diagnostic accuracy while reducing the computational cost. Future studies should explore multimodal inputs and larger, more diverse datasets to further improve early detection. Although our results are promising, there are a number of scientific limitations to consider. Firstly, while the Pitt corpus is widely used, its relatively small sample size may limit the generalisability of the results. Secondly, although adapter-based fine-tuning is computationally efficient, full fine-tuning or the use of multimodal inputs may yield a higher level of diagnostic accuracy. Thirdly, our assessment focused only on linguistic features, which may not fully reflect the cognitive decline associated with AD.

Nonetheless, there are some limitations to this study. First, although the Pitt corpus provides a rich dataset, its sample size is very limited. The number and diversity of samples affect the fine-tuning strategy as well as the performance of the trained model in real-world testing, especially in the diagnosis of Alzheimer's disease. Second, there is a limited number of LLMs used in this project; also, the number of times experiments have been conducted was not sufficient. In the future, more models and their effects in combination with other models can be explored, and more ways to regulate the parameters can be tried.

Author Contributions

Conceptualization, B.I. and Y.D.; methodology, Y.D., and Y.H.; software, Y.H.; validation, Y.D., and Y.H.; formal analysis, L.G.; investigation, Y.H.; resources, Y.D.; data curation, L.G.; writing—original draft preparation, Y.D.; writing—review and editing, Y.D., and J.X.; visualization, Y.H.; supervision, B.I.; project administration, B.I.; funding acquisition, B.I. All authors have read and agreed to the published version of the manuscript.

Funding

This work was supported by Wenzhou-Kean University grant number [IRSPK2023005].

Institutional Review Board Statement

Not applicable.

Informed Consent Statement

Not applicable.

Data Availability Statement

The data utilized in this study are sourced from the publicly accessible Pitt Corpus, available via the DementiaBank database (https://dementia.talkbank.org). Access to the dataset requires approval from DementiaBank to ensure compliance with data usage and ethical guidelines. The models developed and the code used for the analysis are available from the corresponding author upon reasonable request.

Acknowledgments

Our sincere appreciation extends to the university for its financial support, which was instrumental in advancing our research. We also gratefully acknowledge the support provided by the WKU Institute of Advanced Natural Language Processing (IANLP), whose resources and collaborative environment significantly contributed to the successful execution of this work.

Conflicts of Interest

The authors declare no conflict of interest.

References

[1] Lane, C.A., Hardy, J., Schott, J.M., 2018. Alzheimer's disease. European Journal of Neurology. 25(1), 59–70. DOI: https://doi.org/10.1111/ene.13439

[2] Lynch, C., 2020. World Alzheimer Report 2019: Attitudes to dementia, a global survey: Public health: Engaging people in ADRD research. Alzheimer's & Dementia. 16(S10), e038255. DOI: https://doi.org/10.1002/alz.038255

[3] Jack, C.R., et al., 2015. Magnetic resonance imaging in Alzheimer's Disease Neuroimaging Initiative 2. Alzheimer's & Dementia. 11(7), 740–756. DOI: https://doi.org/10.1016/j.jalz.2015.05.002

[4] Liu, N., Yuan, Z., Tang, Q., 2022. Improving Alzheimer's Disease Detection for Speech Based on Feature Purification Network. Frontiers in Public Health. 9, 835960. DOI: https://doi.org/10.3389/fpubh.2021.835960

[5] Wu, J., Yang, S., Zhan, R., et al., 2025. A Survey on LLM-Generated Text Detection: Necessity, Methods, and Future Directions. Computational Linguistics. 51(1), 275–338. DOI: https://doi.org/10.1162/coli_a_00549

[6] Yuan, J., Bian, Y., Cai, X., et al., 2020. Disfluencies and fine-tuning pre-trained language models for detection of Alzheimer's disease. In Proceedings of the 21st Annual Conference of the International Speech Communication Association (Interspeech 2020), Shanghai, China, (25–29 October 2020); pp. 2162–2166. DOI: https://doi.org/10.21437/Interspeech.2020-2516

[7] Matosevic, L., Jovic, A., 2022. Accurate Detection of Dementia from Speech Transcripts Using RoBERTa Model. In Proceedings of the 2022 45th Jubilee International Convention on Information, Communication and Electronic Technology (MIPRO), Opatija, Croatia, 23–27 May 2022; pp. 1478–1484. DOI: https://doi.org/10.23919/MIPRO55190.2022.9803462

[8] Liu, N., Luo, K., Yuan, Z., et al., 2022. A Transfer Learning Method for Detecting Alzheimer's Disease Based on Speech and Natural Language Processing. Frontiers in Public Health. 10, 772592. DOI: https://doi.org/10.3389/fpubh.2022.772592

[9] Liu, N., Yuan, Z., 2022. Spontaneous Language Analysis in Alzheimer's Disease: Evaluation of Natural Language Processing Technique for Analyzing Lexical Performance. Journal of Shanghai Jiaotong University Science. 27(2), 160–167. DOI: https://doi.org/10.1007/s12204-021-2384-3

[10] Liu, N., Wang, L., 2023. An approach for assisting diagnosis of Alzheimer's disease based on natural language processing. Frontiers in Aging Neuroscience. 15, 1281726. DOI: https://doi.org/10.3389/fnagi.2023.1281726

[11] Goodglass, H., Kaplan, E., 1996. The assessment of aphasia and related disorders, 2nd ed. Lea & Febiger: Philadelphia, PA, USA.

[12] Christodoulou, E., Ma, J., Collins, G.S., et al., 2019. A systematic review shows no performance benefit of machine learning over logistic regression for clinical prediction models. Journal of Clinical Epidemiology. 110, 12–22. DOI: https://doi.org/10.1016/j.jclinepi.2019.02.004

[13] Dudzik, W., Nalepa, J., Kawulok, M., 2021. Evolving data-adaptive support vector machines for binary classification. Knowledge-Based Systems. 227, 107221. DOI: https://doi.org/10.1016/j.knosys.2021.107221

[14] Nusinovici, S., Tham, Y.C., Yan, M.Y.C., et al., 2020. Logistic regression was as good as machine learning for predicting major chronic diseases. Journal of Clinical Epidemiology. 122, 56–69. DOI: https://doi.org/10.1016/j.jclinepi.2020.03.002

[15] Hsu, B.M., 2020. Comparison of Supervised Classification Models on Textual Data. Mathematics. 8(5), 851. DOI: https://doi.org/10.3390/math8050851

[16] Abdullah, D.M., Abdulazeez, A.M., 2021. Machine Learning Applications based on SVM Classification A Review. Qubahan Academic Journal. 1(2), 81–90. DOI: https://doi.org/10.48161/qaj.v1n2a50

[17] Salehi, W., Baglat, P., Gupta, G., et al., 2023. An Approach to Binary Classification of Alzheimer's Disease Using LSTM. Bioengineering. 10(8), 950. DOI: https://doi.org/10.3390/bioengineering10080950

[18] Wu, M., Chen, L., 2015. Image recognition based on deep learning. In Proceedings of the 2015 Chinese Automation Congress (CAC), Wuhan, China, (27–29 November 2015); pp. 542–546. DOI: https://doi.org/10.1109/CAC.2015.7382560

[19] Torfi, A., Shirvani, R.A., Keneshloo, Y., et al., 2021. Natural Language Processing Advancements by Deep Learning: A Survey. arXiv:2003.01200v4. DOI: https://doi.org/10.48550/arXiv.2003.01200

[20] Wang, F., Wang, H., Zhou, X., et al., 2022. A

Driving Fatigue Feature Detection Method Based on Multifractal Theory. IEEE Sensors Journal. 22(19), 19046–19059. DOI: https://doi.org/10.1109/JSEN.2022.3201015

[21] Li, C., Zhang, C., Fu, Q., 2020. Research on CNN + LSTM user intention classification based on multi-granularity features of texts. The Journal of Engineering. 2020(13), 486–490. DOI: https://doi.org/10.1049/joe.2019.1175

[22] Howard, J., Ruder, S., 2018. Universal Language Model Fine-tuning for Text Classification. arXiv:1801.06146v5. DOI: https://doi.org/10.48550/arXiv.1801.06146

[23] Zhu, C., Ni, R., Xu, Z., et al., 2021. GradInit: Learning to Initialize Neural Networks for Stable and Efficient Training. Advances in Neural Information Processing Systems, 34, 16410–16422.

[24] Rios, J.O., Armejach, A., Petit, E., et al., 2021. Dynamically Adapting Floating-Point Precision to Accelerate Deep Neural Network Training. In Proceedings of the 2021 20th IEEE International Conference on Machine Learning and Applications (ICMLA), Pasadena, CA, USA, 13–16 December 2021; pp. 980–987. DOI: https://doi.org/10.1109/ICMLA52953.2021.00161

[25] Wu, Y., Liu, J., Bae, J., et al., 2019. Demystifying Learning Rate Policies for High Accuracy Training of Deep Neural Networks. In Proceedings of the 2019 IEEE International Conference on Big Data (Big Data), Los Angeles, CA, USA, (9–12 December 2019); pp. 1971–1980. DOI: https://doi.org/10.1109/BigData47090.2019.9006105

[26] Hu, E.J., Shen, Y., Wallis, P., et al., 2021. LoRA: Low-Rank Adaptation of Large Language Models. arXiv:2106.09685v2. DOI: https://doi.org/10.48550/arXiv.2106.09685

[27] Zeng, Y., Lee, K., 2024. The Expressive Power of Low-Rank Adaptation. arXiv:2310.17513v3. DOI: https://doi.org/10.48550/arXiv.2310.17513

[28] Zhang, B., Liu, Z., Cherry, C., et al., 2024. When Scaling Meets LLM Finetuning: The Effect of Data, Model and Finetuning Method. arXiv:2402.17193v1. 27 Feburary 2024. DOI: https://doi.org/10.48550/arXiv.2402.17193

[29] Dettmers, T., Pagnoni, A., Holtzman, A., et al., 2023. QLoRA: Efficient Finetuning of Quantized LLMs. Advances in neural information processing systems, 36, 10088–10115.

[30] Zhang, X., Rajabi, N., Duh, K., Koehn, P., 2023. Machine Translation with Large Language Models: Prompting, Few-shot Learning, and Fine-tuning with QLoRA. In Proceedings of the Eighth Conference on Machine Translation, Singapore, 06–07 December, 2023; pp. 468–481. DOI: https://doi.org/10.18653/v1/2023.wmt-1.43

[31] Thirunavukarasu, A.J., Ting, D.S.J., Elangovan, K., et al., 2023. Large language models in medicine. Nature Medicine. 29(8), 1930–1940. DOI: https://doi.org/10.1038/s41591-023-02448-8